

OPIS PROGRAMU STUDIÓW

Program studiów obowiązuje dla cykli kształcenia rozpoczynających się od:
rok akademicki **2025/2026**, semestr **zimowy**

Jednostka/i organizacyjna/e: **Wydział Informatyki i Telekomunikacji**

od 1 października 2025 r. Wydział Informatyki i Matematyki

Kierunek studiów: **Informatyka**

Specjalność/i: **brak**

Profil studiów: **ogólnoakademicki**

Poziom studiów: **I stopień**

Poziom Polskiej Ramy Kwalifikacji: **6 PRK**

Forma studiów: **studia niestacjonarne**

Język wykładowy: **język polski**

Dziedzina lub dziedziny nauki/sztuki: **Nauki inżynieryjno-techniczne**

Dyscyplina naukowa/artystyczna wiodąca: **Informatyka techniczna i telekomunikacja (100%)**

- liczba semestrów studiów: **8**
- łączna liczba godzin zajęć organizowanych przez uczelnię: **1632**
- liczba punktów ECTS konieczna do ukończenia studiów: **210**
- liczba punktów ECTS, jaką student musi uzyskać w ramach zajęć prowadzonych z bezpośrednim udziałem nauczycieli akademickich lub innych osób prowadzących zajęcia: **100 (47,61%)**
- liczba punktów ECTS, jaką student musi uzyskać w ramach zajęć z dziedziny nauk humanistycznych lub nauk społecznych: **6**
- liczba punktów ECTS przyporządkowana zajęciom do wyboru: **67(31,9%)**
- liczba punktów ECTS przypisana do zajęć związanych z prowadzoną działalnością naukową w dyscyplinie lub dyscyplinach, do których przyporządkowany jest kierunek studiów – *dotyczy studiów o profilu ogólnoakademickim*: **175**
- liczba punktów ECTS przypisana do zajęć kształtujących umiejętności praktyczne – *dotyczy studiów o profilu praktycznym*: **nie dotyczy**
- liczba godzin zajęć z wychowania fizycznego: **nie dotyczy**
- wymiar odbywania studenckich praktyk zawodowych oraz liczba punktów ECTS, jaką student musi uzyskać w ramach praktyk:

zasady i forma praktyk: **praktyka odbywana w firmach branży IT**

wymiar: **160 godzin (1 miesiąc)**

liczba ECTS: **6**

Zasady odbywania praktyk zawodowych określa Procedura organizacji studenckich praktyk zawodowych na studiach I lub II stopnia o profilu ogólnoakademickim na Politechnice Krakowskiej oraz karta przedmiotu Praktyka

- liczba punktów ECTS uzyskiwanych w ramach kształcenia z wykorzystaniem metod i technik kształcenia na odległość: **do 105 (50%)**
- sposoby weryfikacji i oceny efektów uczenia się osiągniętych przez studenta w trakcie całego cyklu kształcenia:
 - egzamin pisemny,**
 - egzamin ustny,**
 - test,**
 - praca pisemna,**
 - prezentacja tradycyjna lub multimedialna,**
 - projekt,**
 - formy interaktywne (np. wystąpienia, dyskusje)**
- tytuł zawodowy nadawany absolwentowi: **inżynier**

**DODATKOWE INFORMACJE DOTYCZĄCE WSZYSTKICH KART PRZEDMIOTÓW
NA KIERUNKU INFORMATYKA
DLA STUDIÓW STACJONARNYCH I NIESTACJONARNYCH I i II STOPNIA**

1. Szczegółowe informacje o sposobie wyliczenia oceny z przedmiotów znajdujących się w programie studiów dla kierunku Informatyka (ocena formująca i podsumowująca) zostaną przedstawione na pierwszych zajęciach z przedmiotu lub zapisane w regulaminie przedmiotu i udostępnione wszystkim uczestnikom zajęć.

Pkt.5 § 12 Regulaminu studiów na Politechnice Krakowskiej im. Tadeusza Kościuszki

Nauczyciel akademicki prowadzący zajęcia podaje na pierwszych zajęciach szczegółowy opis przedmiotu, z uwzględnieniem zapisów w karcie przedmiotu, zawierający: informacje dotyczące efektów uczenia się, program zajęć, wykaz literatury, warunki uczestnictwa w zajęciach i usprawiedliwiania nieobecności na zajęciach, w tym liczbę dopuszczalnych nieobecności, **warunki uzyskiwania zaliczenia i składania egzaminu**, sposób informowania studentów o uzyskanych wynikach zaliczenia i egzaminu, terminy konsultacji oraz tryb wglądu do ocenionych prac zaliczeniowych i egzaminacyjnych, z zastrzeżeniem, iż studentowi przysługuje prawo wglądu do pracy będącej podstawą zaliczenia lub egzaminu w terminie czternastu dni od daty podania do wiadomości studentów informacji o wynikach zaliczenia lub egzaminu, jednak nie później, niż przed kolejnym terminem egzaminu lub zaliczenia.

2. Wszystkie zajęcia na kierunku Informatyka są możliwe do prowadzenia z wykorzystaniem metod i technik kształcenia na odległość.
3. Do godzin bezpośredniego kontaktu zaliczane są godziny konsultacji i godziny przewidziane na zaliczenia i egzaminy, prezentacje projektów i obrony prac dyplomowych.

Politechnika Krakowska im. Tadeusza Kościuszki w Krakowie

Nazwa wydziału lub wydziałów: Wydział Informatyki i Telekomunikacji Politechniki Krakowskiej

Nazwa kierunku studiów: Informatyka

Poziom studiów: I stopień

Profil studiów: ogólnoakademicki

Dziedzina lub dziedziny nauki:¹ nauki inżyniersko - techniczne

Dyscyplina lub dyscypliny naukowe z określeniem procentowego udziału efektów uczenia się dla każdej dyscypliny:¹ informatyka techniczna i telekomunikacja (100 %)

Poziom Polskiej Ramy Kwalifikacji:² 6 PRK

Symbole efektów uczenia się	Przyporządkowanie do dyscypliny naukowej	KIERUNKOWE EFEKTY UCZENIA SIĘ Obowiązują dla cykli kształcenia rozpoczynających się w roku akademickim 2025/26 i w latach następnych	Odniesienie do		
			uniwersalnych charakterystyk pierwszego stopnia PRK ³	charakterystyk drugiego stopnia efektów uczenia się PRK ⁴	charakterystyk drugiego stopnia efektów uczenia się PRK umożliwiających uzyskanie kompetencji inżynierskich ⁵
1	2	3	4	5	6
		WIEDZA: ABSOLWENT ZNA I ROZUMIE	Kod składnika opisu	Kod składnika opisu	Kod składnika opisu
I1_W01	Informatyka techniczna i telekomunikacja	pojęcia z matematyki przydatne do formułowania i rozwiązywania złożonych zadań informatycznych z wykorzystaniem zagadnień analizy matematycznej, algebry, matematyki dyskretnej, logiki i teorii mnogości, metod probabilistycznych, statystyki i metod numerycznych.	P6U_W	P6S_WG	P6S_WG
I1_W02	Informatyka techniczna i telekomunikacja	pojęcia z zakresu fizyki niezbędne do rozwiązywania problemów informatycznych związanych przede wszystkim z modelowaniem procesów w czasie rzeczywistym.	P6U_W	P6S_WG	P6S_WG
I1_W03	Informatyka techniczna i telekomunikacja	elementarne pojęcia w zakresie elektrotechniki, miernictwa i elektroniki potrzebne do zrozumienia technik cyfrowych i zasad funkcjonowania współczesnych komputerów i sieci komputerowych (szkieletowych i bezprzewodowych).	P6U_W	P6S_WG	P6S_WG

I1_W04	Informatyka techniczna i telekomunikacja	Zagadnienia w zakresie algorytmiki i złożoności obliczeniowej algorytmów. Zna metody, technologie i narzędzia projektowania i implementacji algorytmów.	P6U_W	P6S_WG	P6S_WG
I1_W05	Informatyka techniczna i telekomunikacja	Zagadnienia w zakresie architektury systemów komputerowych, systemów operacyjnych, technologii sieciowych i systemów wbudowanych oraz ma wiedzę o trendach rozwojowych.	P6U_W	P6S_WG	P6S_WG
I1_W06	Informatyka techniczna i telekomunikacja	Zagadnienia w zakresie języków i paradygmatów programowania, programowania obiektowego oraz inżynierii oprogramowania.	P6U_W	P6S_WG	P6S_WG
I1_W07	Informatyka techniczna i telekomunikacja	w sposób uporządkowany ogólne pojęcia z zakresu grafiki komputerowej, komunikacji człowiek-komputer, budowania prostych interfejsów graficznych.	P6U_W	P6S_WG	P6S_WG
I1_W08	Informatyka techniczna i telekomunikacja	teoretyczne szczegóły nt. algorytmiki, projektowania i programowania obiektowego, baz danych i sztucznej inteligencji.	P6U_W	P6S_WG	P6S_WG
I1_W09	Informatyka techniczna i telekomunikacja	modele i narzędzia umożliwiające przesyłanie, przetwarzanie i analizę danych w mikro i makro skali.	P6U_W	P6S_WG	P6S_WG
I1_W10	Informatyka techniczna i telekomunikacja	podstawowe zagadnienia dotyczące cyklu życia systemów informatycznych.	P6U_W	P6S_WG	P6S_WG
I1_W11	Informatyka techniczna i telekomunikacja	podstawowe metody, techniki i narzędzia stosowane przy rozwiązywaniu prostych zadań informatycznych z zakresu analizy złożoności obliczeniowej algorytmów oraz implementacji języków programowania.	P6U_W	P6S_WG	P6S_WG
I1_W12	Informatyka techniczna i telekomunikacja	podstawowe metody, techniki i narzędzia stosowane przy rozwiązywaniu prostych zadań informatycznych z zakresu budowy systemów komputerowych, sieci komputerowych i technologii sieciowych oraz systemów wbudowanych.	P6U_W	P6S_WG	P6S_WG

I1_W13	Informatyka techniczna i telekomunikacja	podstawowe zagadnienia z zakresu kodeksów etycznych dotyczących informatyki, rozumie zagrożenia związane z przestępczością elektroniczną, nt. patentów, ustawy prawo autorskie i prawa pokrewne oraz ustawy o ochronie danych osobowych.	P6U_W	P6S_WK	P6S_WK
I1_W14	Informatyka techniczna i telekomunikacja	podstawowe pojęcia z zakresu ekonomii, zarządzania i prowadzenia działalności gospodarczej w sektorze informatycznym.	P6U_W	P6S_WK	P6S_WK
		UMIEJĘTNOŚCI: ABSOLWENT POTRAFI	Kod składnika opisu	Kod składnika opisu	Kod składnika opisu
I1_U01b	Informatyka techniczna i telekomunikacja	pozyskiwać informacje z literatury, baz danych oraz innych źródeł, integrować je, dokonywać ich interpretacji oraz wyciągać wnioski i formułować opinie.	P6U_U	P6S_UW	P6SU_UW
I1_U02	Informatyka techniczna i telekomunikacja	dostrzegać potrzebę i zna możliwości dalszego doskonalenia się (studia II i III stopnia, studia podyplomowe, kursy i egzaminy przeprowadzane przez uczelnie, firmy i organizacje zawodowe).	P6U_U	P6S_UW	P6SU_UW
I1_U03	Informatyka techniczna i telekomunikacja	porozumiewać się przy użyciu różnych technik w środowisku zawodowym.	P6U_U	P6S_UK	-
I1_U04	Informatyka techniczna i telekomunikacja	posługiwać się językiem angielskim na poziomie B2 w stopniu pozwalającym na porozumienie się, przeczytanie ze zrozumieniem tekstów i opisów programowych oraz przedstawienie prezentacji problemu z zakresu własnego kierunku studiów.	P6U_U	P6S_UK	-
I1_U05	Informatyka techniczna i telekomunikacja	pracować i kierować w zespołach projektowych przy realizacji projektów informatycznych.	P6U_U	P6S_UO	-
I1_U06b	Informatyka techniczna i telekomunikacja	wykorzystać nabytą wiedzę matematyczną do opisu procesów, tworzenia i analizy modeli matematycznych, analizy statystycznej wyników symulacji i pomiarów.	P6U_U	P6S_UW	P6S_UW

I1_U07b	Informatyka techniczna i telekomunikacja	planować i przeprowadzać proste eksperymenty obliczeniowe i wykorzystać je do rozwiązywania zadań informatycznych oraz wyciągać wnioski.	P6U_U	P6S_UW	P6S_UW
I1_U08	Informatyka techniczna i telekomunikacja	dostrzegać aspekty społeczne i prawne przy formułowaniu i rozwiązywaniu zadań informatycznych.	P6U_U	P6S_UW	P6S_UW
I1_U09	Informatyka techniczna i telekomunikacja	stworzyć model obiektowy prostego systemu (np. w języku UML).	P6U_U	P6S_UW	P6S_UW
I1_U10	Informatyka techniczna i telekomunikacja	ocenić złożoność obliczeniową algorytmów i problemów	P6U_U	P6S_UW	P6S_UW
I1_U11	Informatyka techniczna i telekomunikacja	posługiwać się systemami operacyjnymi.	P6U_U	P6S_UW	P6S_UW
I1_U12	Informatyka techniczna i telekomunikacja	zaprojektować odpowiedni interfejs użytkownika dla różnych klas systemów informatycznych	P6U_U	P6S_UW	P6S_UW
I1_U13	Informatyka techniczna i telekomunikacja	zabezpieczyć dane przed nieuprawnionym dostępem	P6U_U	P6S_UW	P6S_UW
I1_U14	Informatyka techniczna i telekomunikacja	poprawnie użyć przynajmniej jednej metody szacowania pracochołności wytwarzania oprogramowania.	P6U_U	P6S_UW	P6S_UW
I1_U15	Informatyka techniczna i telekomunikacja	zaprojektować prostą sieć komputerową oraz pełnić funkcję administratora sieci komputerowej.	P6U_U	P6S_UW	P6S_UW

I1_U16	Informatyka techniczna i telekomunikacja	budować proste systemy bazodanowe, wykorzystując przynajmniej jeden z najbardziej popularnych systemów zarządzania bazą danych.	P6U_U	P6S_UW	P6S_UW
I1_U17	Informatyka techniczna i telekomunikacja	systematycznie przeprowadzać testy oprogramowania.	P6U_U	P6S_UW	P6S_UW
I1_U18	Informatyka techniczna i telekomunikacja	rozwiązywać proste zagadnienia z zakresu sztucznej inteligencji, formułowania algorytmów i projektowania prostych systemów informatycznych.	P6U_U	P6S_UW	P6S_UW
I1_U19	Informatyka techniczna i telekomunikacja	budować proste systemy wbudowane.	P6U_U	P6S_UW	P6S_UW
I1_U20	Informatyka techniczna i telekomunikacja	wykonać analizę sposobu funkcjonowania systemu informatycznego i ocenić istniejące rozwiązania informatyczne, przynajmniej w odniesieniu do ich cech funkcjonalnych.	P6U_U	P6S_UW	P6S_UW
I1_U21	Informatyka techniczna i telekomunikacja	sformułować specyfikację prostych systemów informatycznych w odniesieniu do sprzętu, oprogramowania systemowego i cech funkcjonalnych aplikacji.	P6U_U	P6S_UW	P6S_UW
I1_U22	Informatyka techniczna i telekomunikacja	ocenić, na podstawowym poziomie, przydatność rutynowych metod i narzędzi informatycznych oraz wybrać i zastosować właściwą metodę i narzędzia do typowych zadań informatycznych.	P6U_U	P6S_UW	P6S_UW
I1_U23	Informatyka techniczna i telekomunikacja	zaprojektować oraz zrealizować (zgodnie z zadaną specyfikacją) system informatyczny, używając właściwych metod, technik i narzędzi.	P6U_U	P6S_UW	P6S_UW
		KOMPETENCJE SPOŁECZNE: ABSOLWENT JEST GOTÓW DO	Kod składnika opisu	Kod składnika opisu	-

I1_K01	Informatyka techniczna i telekomunikacja	zrozumienia przyczyny wadliwie działających systemów informatycznych, które doprowadziły do poważnych strat finansowych oraz społecznych.	P6U_K	P6S_KK	-
I1_K02	Informatyka techniczna i telekomunikacja	zrozumienia potrzeby zachowań profesjonalnych i przestrzegania zasad etyki, w tym uczciwości.	P6U_K	P6S_KR	-
I1_K03	Informatyka techniczna i telekomunikacja	pracy indywidualnie i w zespole informatyków, w tym także potrafi zarządzać swoim czasem oraz podejmować zobowiązania i dotrzymywać terminów.	P6U_K	P6S_KO	-
I1_K04	Informatyka techniczna i telekomunikacja	wykazywania się skutecznością w realizacji projektów o charakterze społecznym, naukowo-badawczym lub programistyczno-wdrożeniowym, wchodzących w program studiów lub realizowanych poza studiami.	P6U_K	P6S_KO	-
I1_K05	Informatyka techniczna i telekomunikacja	przekazywania informacji o osiągnięciach informatyki i wielu aspektach zawodu informatyka w sposób powszechnie zrozumiały.	P6U_K	P6S_KR	-
I1_K06	Informatyka techniczna i telekomunikacja	podjęcia roli społecznej absolwenta uczelni technicznej i zrozumienia potrzeby przekazywania opinii społecznej informacji dotyczących aktualnego stanu wiedzy w reprezentowanej dyscyplinie na ogólny rozwój techniki i komunikacji społecznej.	P6U_K	P6S_KO	-

Objaśnienia używanych symboli:

1. Uniwersalne charakterystyki poziomów PRK (pierwszego stopnia):

P = poziom PRK (6, 7)

U = charakterystyka uniwersalna

W = wiedza

U = umiejętności

K = kompetencje społeczne

Przykłady:

P6U_W = poziom 6 PRK, charakterystyka uniwersalna, wiedza

„Absolwent zna i rozumie w zaawansowanym stopniu – fakty, teorie, metody oraz złożone zależności między nimi. Absolwent zna i rozumie różnorodne, złożone uwarunkowania prowadzonej działalności.”

P7U_W = poziom 7 PRK, charakterystyka uniwersalna, wiedza

„Absolwent zna i rozumie w pogłębiony sposób wybrane fakty, teorie, metody oraz złożone zależności między nimi, także w powiązaniu z innymi dziedzinami. Absolwent zna i rozumie różnorodne, złożone uwarunkowania i aksjologiczny kontekst prowadzonej działalności.”

2. Charakterystyki poziomów PRK typowe dla kwalifikacji uzyskiwanych w ramach szkolnictwa wyższego (drugiego stopnia):

P = poziom PRK (6, 7)

S = charakterystyka typowa dla kwalifikacji uzyskiwanych w ramach szkolnictwa wyższego

W = wiedza

G = głębia i zakres

K = kontekst

U = umiejętności

W = wykorzystanie wiedzy

K = komunikowanie się

O = organizacja pracy

U = uczenie się

K = kompetencje społeczne

K = krytyczna ocena

O = odpowiedzialność

R = rola zawodowa

Przykłady:

P6S_WG = poziom 6 PRK, charakterystyka typowa dla kwalifikacji uzyskiwanych w ramach szkolnictwa wyższego, wiedza- głębia i zakres

„Absolwent zna i rozumie w zaawansowanym stopniu – wybrane fakty, obiekty i zjawiska oraz dotyczące ich metody i teorie wyjaśniające złożone zależności między nimi, stanowiące podstawową wiedzę ogólną z zakresu dyscyplin naukowych lub artystycznych tworzących podstawy teoretyczne oraz wybrane zagadnienia z zakresu wiedzy szczegółowej – właściwe dla programu studiów, a w przypadku studiów o profilu praktycznym – również zastosowania praktyczne tej wiedzy w działalności zawodowej związanej z ich kierunkiem ”

P7S_WG = poziom 7 PRK, charakterystyka typowa dla kwalifikacji uzyskiwanych w ramach szkolnictwa wyższego, wiedza - głębia i zakres

„Absolwent zna i rozumie w pogłębionym stopniu – wybrane fakty, obiekty i zjawiska oraz dotyczące ich metody i teorie wyjaśniające złożone zależności między nimi, stanowiące zaawansowaną wiedzę ogólną z zakresu dyscyplin naukowych lub artystycznych tworzących podstawy teoretyczne, uporządkowaną i podbudowaną teoretycznie wiedzę obejmującą kluczowe zagadnienia oraz wybrane zagadnienia z zakresu zaawansowanej wiedzy szczegółowej – właściwe dla programu studiów, a w przypadku studiów o profilu praktycznym – również zastosowania praktyczne tej wiedzy w działalności zawodowej związanej z ich kierunkiem”. Absolwent zna i rozumie główne tendencje rozwojowe dyscyplin naukowych lub artystycznych do których jest przyporządkowany kierunek studiów – w przypadku studiów o profilu ogólnoakademickim.”

3. W przypadku braku Kodu składnika opisu należy wprowadzić poziomą kreskę.

¹ W przypadku więcej niż jednej dziedziny nauki/sztuki lub dyscypliny naukowej/artystycznej należy wpisać wszystkie, zgodnie z rozporządzeniem Ministra Nauki i Szkolnictwa Wyższego z dnia 20 września 2018 r. w sprawie dziedzin nauki i dyscyplin naukowych oraz dyscyplin artystycznych (Dz.U. z 2018 r. poz.1818).

² Należy podać właściwy poziom Polskiej Ramy Kwalifikacji, zgodnie z ustawą z dnia 22 grudnia 2015 r. o Zintegrowanym Systemie Kwalifikacji (Dz.U. z 2018 r. poz.2153).

³ Opis zakładanych efektów uczenia się dla kierunku studiów, poziomu i profilu uwzględnia uniwersalne charakterystyki pierwszego stopnia określone w ustawie z dnia 22 grudnia 2015 r. o Zintegrowanym Systemie Kwalifikacji, właściwe dla danego poziomu Polskiej Ramy Kwalifikacji.

⁴ Wszystkie charakterystyki drugiego stopnia efektów uczenia się określone w rozporządzeniu Ministra Nauki i Szkolnictwa Wyższego z dnia 14 listopada 2018 r. w sprawie charakterystyk drugiego stopnia efektów uczenia się dla kwalifikacji na poziomach 6-8 Polskiej Ramy Kwalifikacji (Dz.U. 2018 r. poz. 2218) - część I.

⁵ Część III - charakterystyki drugiego stopnia efektów uczenia się dla kwalifikacji na poziomach 6 i 7 Polskiej Ramy Kwalifikacji umożliwiających uzyskanie kompetencji inżynierskich (rozwiniecie opisów zawartych w części I) opisane w rozporządzeniu Ministra Nauki i Szkolnictwa Wyższego z dnia 14 listopada 2018 r. w sprawie charakterystyk drugiego stopnia efektów uczenia się dla kwalifikacji na poziomach 6-8 Polskiej Ramy Kwalifikacji.

Legenda: W - wykłady, C - ćwiczenia, L - laboratoria, LK - laboratoria komputerowe, P - projekty, S - seminaria, PZ - praktyka zawodowa

Karta przedmiotu

Administrowanie systemami komputerowymi nazwa przedmiotu
Computer systems administration nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Systemy operacyjne
2. Sieci komputerowe
3. Architektura komputerów
4. Znajomość języka angielskiego w stopniu umożliwiającym studiowanie literatury

Cele przedmiotu:

1. Zapoznanie studentów z zagadnieniami administrowania systemami klasy Unix. Szczególnie uwzględnione zostaną zagadnienia związane z instalacją systemu, instalacją oprogramowania, zarządzaniem użytkownikami, kontrolą procesów sieciowych oraz instalacją i konfiguracją serwerowych usług sieciowych i intranetowych, serwerów takich jak DNS, FTP/SCP, HTTP/HTTPS, SMTP/SMTPs, POP/POPs, NFS czy Samba. Zapoznanie studentów z zagadnieniami administrowania systemami klasy MS Windows w zakresie konfiguracji dostępu do usług serwera Linux.

2. Celem ćwiczeń laboratoryjnych będzie zdobycie praktycznych umiejętności odnośnie zagadnień poruszanych na wykładach, a związanych z administrowaniem systemami linuksowymi.
3. Dodatkowym celem ćwiczeń laboratoryjnych będzie zdobycie praktycznych umiejętności związanych z konfiguracją sieciowych serwerowych systemów linuksowych. Niektóre z ważniejszych ćwiczeń to umiejętność konfiguracji sieciowych serwerowych systemów, takich jak serwer DNS, FTP/SCP, HTTP/HTTPs, SMTP/SMTPs, POP/POPs, NFS, RDP, VNC, Samba oraz DHCP.
4. Nabycie umiejętności pracy w małych zespołach.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	Student posiada wiedzę z zakresu zagadnień administrowania systemami klasy Unix, a zwłaszcza z zagadnień związanymi z instalacją systemu, instalacją oprogramowania, zarządzaniem użytkownikami, kontrolą procesów sieciowych oraz instalacją i konfiguracją serwerowych usług sieciowych i intranetowych, serwerów takich jak DNS, FTP/SCP, HTTP/HTTPs, SMTP/SMTPs, POP/POPs, NFS czy Samba. Student będzie posiadał wiedzę z zakresu zagadnień systemami klasy MS Windows w zakresie konfiguracji dostępu do usług serwera Linux.	I1_W05
Umiejętności Absolwent potrafi:		
EU1	Student posiada umiejętność posługiwania się najważniejszymi poleceniami i narzędziami administracyjnymi w systemach linuksowych.	I1_U11
EU2	Student potrafi konfigurować linuksowe serwerowe usługi sieciowe, takie jak usługi serwera DNS, FTP/SCP, HTTP/HTTPs, SMTP/SMTPs, POP/POPs, NFS, RDP, VNC, Samba oraz DHCP.	I1_U13
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Student potrafi współpracować w grupie.	I1_K03

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
8	6	E	18	0	0	18	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	Omówienie zagadnień ogólnych związanych z administracją systemami komputerowymi. Przybliżenie sylwetki administratora. Zapoznanie z zagadnieniami z zakresu zarządzania użytkownikami, dokumentacji i etyki pracy administratora.	2
2	W	Zapoznanie z procesem uruchamiania komputera oraz wyboru systemu operacyjnego. Omówienie zawartości pierwszych sektorów dysku twardego. Analiza tablicy partycji. Omówienie bootloadera GRUB dla systemu Linuks. Zapoznanie z procesem uruchamiania systemu operacyjnego wraz z niezbędnymi usługami.	2
3	W	Zapoznanie ze sposobem przechowywania plików na dysku. Analiza i porównanie wybranych systemów plików. Omówienie typów macierzy dyskowych RAID. Zapoznanie z LVM jako elastycznego narzędzia do zarządzania przestrzenią dyskową.	2
4	W	Omówienie zagadnień związanych z kopią zapasową, takich jak tworzenie, przywracanie, rodzaje, nośniki danych. Omówienie mechanizmów CDP i deduplikacji. Szczegółowe omówienie dwóch systemów tworzenia kopii zapasowej: Bacula i Amanda.	2
5	W	Zapoznanie z procesem monitorowania działania systemu operacyjnego i pracy użytkowników. Omówienie popularnych programów i komend, zapoznanie z wybranymi plikami systemu operacyjnego. Poruszenie problemu etyki monitorowania użytkowników.	2
6	W	Zapoznanie z zagadnieniami związanymi z tzw. cienkim klientem. Omówienie oprogramowania Linux Diskless Workstation, Linux Terminal Server Project, PC over IP a także inne narzędzia zdalnego administrowania. Instalacja i konfiguracja LAMP (Linux, Apache, MySQL, PHP).	1
7	W	Zapoznanie z zagadnieniami z zakresu prawa w pracy administratora. Rodzaj umowy dla administratora. Administrator Bezpieczeństwa Informacji oraz ochrona danych osobowych. Aspekty prawne Internetu: Internet a prawo ustrojowe, Internet a prawo procesowe, Internet a prawo materialne.	2
8	W	Zapoznanie się z oprogramowaniem do translacji nazw domenowych oraz systemem przesyłania poczty. Podstawowa i zaawansowana konfiguracja serwera nazw bind. Konfiguracja wraz z testowaniem systemu pocztowego Postfix. Konfiguracją bezpiecznego przesyłania plików z wykorzystaniem protokołu FTP. Podstawowa konfiguracja serwera Samba. Konfiguracja serwera NFS.	2

9	W	Omówienie problemów i zagadnień dotyczących pomieszczenia serwerowni. Poruszane zagadnienia to: wyposażenie, systemy bezpieczeństwa, monitorowanie, zarządzanie, zasilanie, zdalna konfiguracja.	1
10	W	Omówienie zagadnień związanych z niechcianą pocztą. Analiza wiadomości pod kątem niechcianej poczty, sposoby wysyłania i zapobiegania wysyłaniu, narzędzia do ochrony przed niechcianą pocztą.	2
11	LK	Laboratorium polega na zapoznaniu się z wybranymi poleceniami oraz ważniejszymi plikami konfiguracyjnymi systemu Linuks, np. polecenia pozwalające na uzyskanie pomocy takie jak help, ?help, type, man, apropos, finger, info, file oraz polecenia wyszukujące pliki.	2
12	LK	Praktycznie ćwiczenie pozwalające na zapoznanie się z zarządzaniem systemami kontroli wersji plików (VCS) na przykładzie Git. Zapoznanie z najbardziej podstawowymi czynnościami wykonywanymi z użyciem VCS, a także zapoznanie ze sposobami konfiguracji systemów typu VCS.	2
13	LK	Laboratorium obejmuje zagadnienia z zakresu przechowywania plików i budowy pamięci masowych. Praktycznie ćwiczenia pozwalające na dokładniejsze poznanie budowy różnych tablicy partycji. Zapoznanie z dynamicznym sposobem zarządzania miejscem na dysku twardym z wykorzystaniem narzędzia LVM.	2
14	LK	Zapoznanie z zaawansowanym systemem tworzenia kopii zapasowej Bacula. Ćwiczenia praktyczne pozwalające na analizę działania mechanizmu tworzenia kopii zapasowej. Skonfigurowanie systemu do wykonywania kopii zapasowej a także przywrócenie plików z wykonanej kopii. Zapoznanie z narzędziem do synchronizacji plików rsync. Poznanie narzędzia BackInTime jako systemu kopii zapasowej z podstawową deduplikacją.	2
15	LK	Laboratorium pozwala na zaznajomienie się z narzędziami do monitorowania systemu operacyjnego. Instalacja i konfiguracja kilku wybranych narzędzi do monitorowania. narzędzia do efektywnego przeszukiwania logów systemowych.	1
16	LK	Laboratorium pozwala na zapoznanie z oprogramowaniem do zdalnego zarządzania innym komputerem. Ćwiczenia praktyczne z konfiguracji i korzystania z protokołów VNC i RDP. Zapoznanie z metodami zabezpieczania połączenia VNC. Zapoznanie z zaawansowanym korzystaniem z oprogramowania SSH.	2
17	LK	Laboratorium pozwala na zapoznanie się z oprogramowaniem do serwowania stron internetowych Apache oraz systemem zarządzania relacyjną bazą danych MySQL. Ćwiczenia praktyczne dotyczące programu Apache: podstawowa konfiguracja, zarządzania modułami, autoryzacja, obsługa przekierowań, sterowanie dostępem. Ćwiczenia praktyczne z systemu MySQL: tworzenie i przywracanie kopii zapasowej, migracja bazy danych, klastrowanie/replikacja.	2
18	LK	Laboratorium pozwala na zapoznanie się z oprogramowaniem do translacji nazw domenowych oraz systemem przesyłania poczty. Podstawowa i zaawansowana konfiguracja serwera nazw bind. Konfiguracja wraz z testowaniem systemu pocztowego Postfix.	1
19	LK	Zapoznanie z protokołami udostępniania i przesyłania plików. Zapoznanie z konfiguracją bezpiecznego przesyłania plików z wykorzystaniem protokołu FTP. Podstawowa konfiguracja serwera Samba. Konfiguracja serwera NFS.	2

20	LK	Laboratorium realizuje praktyczne ćwiczenia z zakresu systemów autoryzacji. Zarządzanie i konfiguracja modułów systemu PAM. Wykorzystanie systemu OPIE jako systemu do logowania hasłami jednorazowymi. Konfiguracja serwera NIS do zdalnego uwierzytelniania.	2
----	----	--	---

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Egzaminy i zaliczenia (w bezpośrednim kontakcie z prowadzącym)	2
2	Konsultacje przedmiotowe (w bezpośrednim kontakcie z prowadzącym)	18
3	Przygotowanie do egzaminu	8
4	Przygotowanie do wykonania ćwiczeń laboratoryjnych i opracowanie wyników	86

Metody dydaktyczne:

konsultacje, wykład, laboratorium komputerowe, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, egzamin pisemny, odpowiedź ustna, sprawozdanie

Kryteria oceny:

1. Wymaganie na ocenę 5,0: student potrafi sporządzić poprawną dokumentację konfiguracji systemu, która pozwoli innemu pracownikowi szybko i sprawnie zorientować się w tej konfiguracji. Student potrafi także sporządzić poprawną instrukcję dla użytkownika, dzięki której użytkownik bez problemu będzie wiedział jak korzystać z systemu w sposób prawidłowy, a także będzie wiedział w jaki sposób zgłaszać problemy administratorowi aby mógł on szybko odtworzyć błąd u siebie gdy u administratora wszystko działa. Wymaganie na ocenę 4,5: student opanował mniej niż 91% ww. zagadnień, ale więcej niż 80%. Wymaganie na ocenę 4,0: student opanował mniej niż 81% ww. zagadnień, ale więcej niż 70%. Wymaganie na ocenę 3,5: student opanował mniej niż 71% ww. zagadnień, ale więcej niż 60%. Wymaganie na ocenę 3,0: student opanował mniej niż 61% ww. zagadnień, ale więcej niż 50%.
2. Wymaganie na ocenę 5,0: student potrafi sprawnie poruszać się po systemie operacyjnym celem konfiguracji systemu, zlokalizowania usterki, dokonania zmian w konfiguracji i wykonania kopii zapasowej. Wymaganie na ocenę 4,5: student opanował mniej niż 91% ww. zagadnień, ale więcej niż 80%. Wymaganie na ocenę 4,0: student opanował mniej niż 81% ww. zagadnień, ale więcej niż 70%. Wymaganie na ocenę 3,5: student opanował mniej niż 71% ww. zagadnień, ale więcej niż 60%. Wymaganie na ocenę 3,0: student opanował mniej niż 61% ww. zagadnień, ale więcej niż 50%.
3. Wymaganie na ocenę 5,0: student potrafi poprawnie zainstalować system Linux i skonfigurować przerabiane na zajęciach systemy takie jak serwer LAMP, repozytorium projektu, macierz RAID, zarządzanie użytkownikami, udostępnianie plików w sieci komputerowej oraz zabezpieczenie ich przed niepożądanym dostępem, wykonanie kopii zapasowej danych oraz przywrócenie danych z wykonanej kopii. Wymaganie na ocenę 4,5: student opanował mniej niż 91% ww. zagadnień, ale więcej niż 80%. Wymaganie na ocenę 4,0: student opanował mniej niż 81% ww. zagadnień, ale więcej niż 70%. Wymaganie na ocenę 3,5: student opanował mniej niż 71% ww. zagadnień, ale więcej niż 60%. Wymaganie na ocenę 3,0: student opanował mniej niż 61% ww. zagadnień, ale więcej niż 50%.

4. Wymaganie na ocenę 5,0: student ma wiedzę na temat sposobu uruchomienia systemu operacyjnego na serwerze, wiedzę na temat różnych konfiguracji macierzy RAID i systemów plików oraz wiedzę na temat różnych usług działających na serwerze takich jak: serwer pocztowy, serwer WWW, serwer aplikacji webowych, serwer plików, serwer bazy danych oraz wykonywania kopii zapasowych. Jest świadom odpowiedzialności jaka ciąży na pracowniku administratora systemów komputerowych odnośnie utrzymania systemu w działaniu i bezpieczeństwa danych. Wymaganie na ocenę 4,5: student opanował mniej niż 91% ww. zagadnień, ale więcej niż 80%. Wymaganie na ocenę 4,0: student opanował mniej niż 81% ww. zagadnień, ale więcej niż 70%. Wymaganie na ocenę 3,5: student opanował mniej niż 71% ww. zagadnień, ale więcej niż 60%. Wymaganie na ocenę 3,0: student opanował mniej niż 61% ww. zagadnień, ale więcej niż 50%.

Literatura:

obowiązkowa:

1. Matotek, et. al, "Linux: profesjonalne administrowanie systemem", Gliwice, Helion, 2023
2. Nemeth Evi, et.al. "Unix and Linux system administration handbook (5th Edition)", Penguin Putnam Inc, 2017

zalecana/fakultatywna:

1. William E. Shotts Jr. "Linux. Wprowadzenie do wiersza poleceń", Gliwice, 2021, Helion
2. Evi Nemeth, Garth Snyder, Trent R. Hein, Ben Whaley "Unix i Linux. Przewodnik administratora systemów", Wydanie IV, Gliwice, 2011, Helion
3. Liczne dostępne na stronach Microsoftu manuale dotyczące instalacji, konfiguracji i głównych zadań wykonywanych przy administrowaniu systemu Microsoft Windows Server.
4. Własne materiały, które będą dostarczone studentom w wersji elektronicznej.

Algebra z geometrią nazwa przedmiotu
Algebra and geometry nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Wiedza z Matematyki na poziomie określonym podstawą programową dla liceum ogólnokształcącego.

Cele przedmiotu:

1. Zapoznanie studentów z podstawowymi pojęciami teorii zbiorów, podstawowymi strukturami algebraicznymi (grupa, ciało) zbiorami liczbowymi, teorii podzielności w pierścieniu liczb całkowitych.
2. Zapoznanie studentów z podstawowymi pojęciami i metodami algebry liniowej. W tym rozwiązywanie układów równań, faktoryzacja macierzy.
3. Zapoznanie studentów z podstawowymi pojęciami i metodami trójwymiarowej geometrii analitycznej.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	podstawowe definicje i twierdzenia z zakresu podstaw teorii zbiorów, relacji i funkcji, struktur algebraicznych, liczb zespolonych, podzielności, kongruencji, równań diofantycznych.	I1_W01
EW2	podstawowe definicje i twierdzenia z zakresu algebry liniowej.	I1_W01
EW3	podstawowe definicje i twierdzenia z zakresu geometrii analitycznej.	I1_W01
Umiejętności Absolwent potrafi:		
EU1	rozwiązywać zadania dotyczące podstaw teorii zbiorów, relacji i funkcji, teorii struktur algebraicznych, liczb zespolonych, podzielności, kongruencji, równań diofantycznych.	I1_U06b
EU2	rozwiązywać zadania dotyczące algebry liniowej w tym układy równań liniowych metoda Gausa i metoda Cramera. Dokonywać rozkładów LU, QR oraz SVD macierzy. Sprowadzać macierz odwzorowania do postaci Jordana.	I1_U06b
EU3	rozwiązywać zadania dotyczące podstaw geometrii analitycznej.	I1_U06b

Kompetencje społeczne		
Absolwent jest gotów do:		
EK1	aktywnego udziału w zajęciach i konsultacjach, systematycznej pracy, w tym samodzielnego poszerzania wiedzy. Student jest gotów do postępowania etycznego, oraz nie korzystania z niedozwolonej pomocy w czasie sprawdzianów i egzaminów.	I1_K02

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
1	5	E	27	27	0	0	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	Podstawy teorii zbiorów (suma, różnica, iloczyn i iloczyn kartezjański zbiorów), relacje (relacje równoważności, klasy równoważności, zbiór ilorazowy), funkcje (iniekcja, suriekcja, bijekcja, funkcja odwrotna). Działania (własności działań), grupa, ciało, ciało liczb zespolonych (część rzeczywista i urojona, moduł, argument, postać trygonometryczna, mnożenie, dzielenie, potęgowanie, pierwiastkowanie liczb zespolonych w postaci trygonometrycznej).	5
2	W	Podzielność w pierścieniu liczb całkowitych (cechy i kryteria podzielności, liczby pierwsze i złożone, rozkład na czynniki, NWD, NWW, algorytm Euklidesa), kongruencje (własności kongruencji zbioru reszt), równania diofantyczne (uogólniony algorytm Euklidesa).	4
3	W	Macierze (wymiar, działania, typy macierzy kwadratowych), wyznaczniki (permutacja znak permutacji, twierdzenie Laplace'a o rozwijaniu wyznacznika, macierze nieosobliwe, macierz odwrotna), układy równań liniowych (rzad macierzy, twierdzenie Kroneckera, twierdzenie Cramera). Metoda Gaussa i rozkład L_U.	5
4	W	Przestrzenie wektorowe (liniowa zależność i niezależność wektorów, baza i wymiar przestrzeni liniowej), odwzorowania liniowe, macierz odwzorowania liniowego (macierz przejścia, zmiana macierzy odwzorowania liniowego przy zmianie bazy). Wektory i wartości własne (postać macierzy odwzorowania liniowego w bazie wektorów własnych, diagonalizowalność macierzy, postać Jordana). Iloczyn skalarny, bazy ortonormalne i macierze ortogonalne. Rozkłady QR i SVD.	8

5	W	Rachunek wektorowy (dodawanie wektorów, mnożenie wektora przez liczbę, iloczyn skalarny, wektorowy, mieszany), prostokątny kartezjański układ współrzędnych dodatnio zorientowany, równanie płaszczyzny (równanie ogólne, interpretacja współczynników, równoległość płaszczyzn, odległość punktu od płaszczyzny, kąt między płaszczyznami) i prostej (równanie krawędziowe, równanie parametryczne, interpretacja współczynników w równaniu parametrycznym, równoległość i skośność prostych, odległość prostych skośnych, odległość punktu od prostej, kąt między prostymi, prosta na płaszczyźnie). Krzywe stopnia drugiego. Niekartezjańskie układy na płaszczyźnie. Translacje i obroty na płaszczyźnie.	5
6	C	Rozwiązywanie zadań dotyczących podstaw teorii zbiorów, relacji, funkcji, struktur algebraicznych, ciała liczb zespolonych.	5
7	C	Rozwiązywanie zadań dotyczących podzielności w pierścieniu liczb całkowitych, kongruencji oraz równań diofantycznych.	4
8	C	Rozwiązywanie zadań dotyczących macierzy, wyznaczników, układów równań liniowych. Rozkład LU.	5
9	C	Rozwiązywanie zadań dotyczących przestrzeni wektorowych, odwzorowań liniowych, macierzy odwzorowania liniowego, wektorów i wartości własnych. Znajdowanie postaci Jordana. Wyznaczanie rozkładów QR i SVD.	9
10	C	Rozwiązywanie zadań dotyczących rachunku wektorowego w prostokątnym kartezjańskim układzie współrzędnych dodatnio zorientowanym, równania płaszczyzny i prostej.	4

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury	30
2	Rozwiązanie zadań udostępnianych za pośrednictwem Teams/Delta	30
3	Konsultacje przedmiotowe w bezpośrednim kontakcie z prowadzącym	6
4	Egzaminy i zaliczenia w sesji w bezpośrednim kontakcie z prowadzącym	5

Metody dydaktyczne:

ćwiczenia, konsultacje, wykład, ćwiczenia tablicowe

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, egzamin pisemny, kolokwium, test, kolokwium z zadań

Kryteria oceny:

1. Student na ocenę 5.0 zna i potrafi przedstawić podstawowe definicje i twierdzenia z zakresu podstaw teorii zbiorów, relacji i funkcji, struktur algebraicznych, liczb zespolonych, podzielności, kongruencji, równań diofantycznych. Wybrane twierdzenia potrafi udowodnić. Zakres obejmuje treści programowe wykładów W1, W2.
2. Student na ocenę 5.0 zna i potrafi przedstawić podstawowe definicje i twierdzenia z zakresu algebry liniowej, wykazane w treściach programowych przedmiotu, wykłady W3, W4.
3. Student na ocenę 5.0 zna i potrafi przedstawić podstawowe definicje i wzory z zakresu geometrii analitycznej wykazane w treściach programowych przedmiotu, wykład W5.

4. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji umiejętności
5. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji umiejętności
6. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji umiejętności
7. Student na ocenę 5.0 aktywnie uczestniczy we wszystkich formach zajęć, systematycznie pracuje, samodzielnie poszerza swoją wiedzę. Student postępuje etycznie, nie korzystania z niedozwolonej pomocy w czasie sprawdzianów i egzaminów.

Literatura:

obowiązkowa:

1. B. Gdowski, R. Pluciński, Zadania z rachunku wektorowego i geometrii analitycznej, Warszawa, 1982, PWN
2. W. Stankiewicz, Zadania z matematyki dla wyższych uczelni technicznych, Warszawa, 1998, PWN
3. S. Przybyło, A. Szlachetowski, Algebra i wielowymiarowa geometria analityczna w zadaniach, Warszawa, 2015, WNT

zalecana/fakultatywna:

1. A. Białynicki-Birula, Algebra liniowa z geometria, Warszawa, 1979, PWN
2. A. Piękosz, Algebra liniowa, Kraków, 2010, PK

Algorytmy i struktury danych nazwa przedmiotu
Algorithms and data structures nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia
Forma studiów: niestacjonarne

Wymagania wstępne:

1. Zaliczenie przedmiotów: "Wstęp do programowania", "Analiza matematyczna".

Cele przedmiotu:

1. Zapoznanie studentów z podstawowymi strukturami algorytmicznymi
2. Zapoznanie studentów z podstawowymi specyfikacjami algorytmów oraz metodami projektowania
3. Zapoznanie studentów z metodami i technikami analizy algorytmów
4. Zapoznanie studentów z dynamicznymi strukturami danych i sposobami reprezentacji

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	Student zna podstawowe metody budowania algorytmów i dokonuje poprawnego porównania i wyboru.	I1_W04
Umiejętności Absolwent potrafi:		
EU1	student umie pozyskiwać informacje z literatury, integrować je, dokonywać ich interpretacji oraz wyciągać wnioski i zastosować je do realizacji swojego projektu	I1_U01b
EU2	student umie dokonać analizy algorytmów	I1_U10
Kompetencje społeczne Absolwent jest gotów do:		
EK1	student umie pracować indywidualnie i potrafi zarządzać swoim czasem oraz podejmować zobowiązania i dotrzymywać terminów	I1_K03

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
2	5	E	18	0	0	18	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	Przegląd metod i technik projektowania algorytmów	2
2	W	Elementy analizy algorytmów. Złożoność obliczeniowa algorytmów. Notacje asymptotyczne. Metody wyznaczania złożoności obliczeniowej.	3
3	W	Strategia "dziel i zwyciężaj", metoda programowania dynamicznego. Algorytmy iteracyjne i rekurencyjne implementujące wyszukiwanie binarne, ciąg Fibonacciego, dwumian Newtona, wieże Hanoi.	2
4	W	Algorytmy sortowania	3
5	W	Abstrakcyjne typy danych. Dynamiczne struktury danych	6
6	W	Haszowanie	2
7	LK	Algorytmy całkowania	3
8	LK	Algorytmy sortowania	3
9	LK	Algorytmy rekurencyjne, abstrakcyjne typy danych	1
10	LK	Struktury dyskretne	2
11	LK	Realizacja projektów: 1 - algorytmy całkowania 2 - algorytmy sortowania 3 - projekt indywidualny	9

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Opracowanie wyników	20
2	Przygotowanie się do zajęć	40
3	Przygotowanie projektu	30
4	Egzaminy i zaliczenia w sesji (w bezpośrednim kontakcie z prowadzącym)	6
5	Konsultacje (w bezpośrednim kontakcie z prowadzącym)	18

Metody dydaktyczne:

ćwiczenia, konsultacje, projekt, wykład, ćwiczenia tablicowe, laboratorium komputerowe, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, dyskusja, egzamin pisemny, kolokwium, odpowiedź ustna, projekt, sprawozdanie z pracy indywidualnej, test, zaliczenie pisemne, sprawdzian wiadomości z wykładów, sprawozdanie

Kryteria oceny:

1. Na ocenę 5,0 student zna metody konstruowania algorytmów i ich analizy
2. Na ocenę 5,0 student potrafi wybrać właściwy algorytm do rozwiązania problemu, przeanalizować go, ocenić jego złożoność.

3. Na ocenę 5,0 student potrafi rozwiązać problem algorytmiczny bazując na wiedzy pozyskanej z literatury
4. NA ocenę 5,0 student dotrzymuje terminów realizacji projektów

Literatura:

obowiązkowa:

1. Cormen T. et. al., Wprowadzenie do algorytmów, Warszawa, WNT.
2. Heineman G. et. al., Algorytmy. Almanach, Gliwice, 2010, Helion.

zalecana/fakultatywna:

1. R. Sedgewick, Algorytmy w C++, Warszawa, 1999, Wydawnictwo RM
2. N. Wirth, Algorytmy + struktury danych = programy, Warszawa, WNT

Analiza matematyczna nazwa przedmiotu
Mathematical analysis nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Drugi semestr mogą zaliczać tylko osoby mające zaliczony pierwszy.

Cele przedmiotu:

1. Zaznajomienie studentów z rachunkiem różniczkowym i całkowym.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	pojęcia z matematyki przydatne do formułowania i rozwiązywania złożonych zadań informatycznych z wykorzystaniem zagadnień analizy matematycznej, algebry, matematyki dyskretnej, logiki i teorii mnogości, metod probabilistycznych, statystyki i metod numerycznych.	I1_W01
Umiejętności Absolwent potrafi:		
EU1	wykorzystać nabytą wiedzę matematyczną do opisu procesów, tworzenia i analizy modeli matematycznych, analizy statystycznej wyników symulacji i pomiarów.	I1_U06b
Kompetencje społeczne Absolwent jest gotów do:		
EK1	zrozumienia potrzeby zachowań profesjonalnych i przestrzegania zasad etyki, w tym uczciwości.	I1_K02

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
1	5	Z	18	18	0	0	0	0	BRAK DANYCH
2	5	E	18	18	0	0	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	Funkcje elementarne, Ogólne własności funkcji.	6
2	C	Funkcje elementarne. Ogólne własności funkcji.	5

3	C	Funkcje elementarne i ogólne własności funkcji-sprawdzian	1
4	W	Granice ciągów i zbieżność szeregów.	6
5	C	Obliczanie granic ciągów i badanie zbieżności szeregów.	5
6	C	Obliczanie granic ciągów i badanie zbieżności szeregów-sprawdzian.	1
7	W	Rachunek różniczkowy funkcji jednej zmiennej.	6
8	C	Obliczanie granic funkcji, pochodnych. Badanie funkcji.	5
9	C	Rachunek różniczkowy funkcji jednej zmiennej - sprawdzian.	1
10	W	Całki nieoznaczone i oznaczone.	6
11	C	Obliczanie całek nieoznaczonych i oznaczonych.	5
12	C	Całki nieoznaczone i oznaczone - sprawdzian.	1
13	W	Rachunek różniczkowy funkcji wielu zmiennych.	6
14	C	Obliczanie pochodnych cząstkowych, pochodnych kierunkowych, pochodnych funkcji dwóch zmiennych, znajdowanie ekstremów funkcji dwóch zmiennych.	5
15	C	Rachunek różniczkowy funkcji dwóch zmiennych - sprawdzian.	1
16	W	Całki wielokrotne, krzywoliniowe, powierzchniowe.	6
17	C	Obliczanie całek wielokrotnych, krzywoliniowych, powierzchniowych.	5
18	C	Całki wielokrotne, krzywoliniowe, powierzchniowe. - sprawdzian	1

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Studiowanie notatek z wykładów.	45
2	Konsultacje przedmiotowe w bezpośrednim kontakcie z prowadzącym	38
3	Egzamin i zaliczenie w bezpośrednim kontakcie z prowadzącym	15
4	Rozwiązywanie zadań	45
5	Przygotowanie się do zajęć	35

Metody dydaktyczne:

konsultacje, wykład, ćwiczenia tablicowe

Sposoby weryfikacji i oceny efektów uczenia się:

egzamin pisemny, test, zaliczenie pisemne

Kryteria oceny:

1. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji kompetencji społecznych.
2. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji umiejętności.
3. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji wiedzy.

Literatura:

obowiązkowa:

1. Analiza matematyczna w zadaniach. Cz. 1 Lech Włodarski, Włodzimierz Krysiński
2. Analiza matematyczna w zadaniach. Cz. 2 Lech Włodarski, Włodzimierz Krysiński

zalecana/fakultatywna:

1. materiały przygotowane przez prowadzącego przedmiot

Architektura systemów komputerowych nazwa przedmiotu
Computer system architecture nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Znajomość podstaw elektroniki i techniki cyfrowej

Cele przedmiotu:

1. Zapoznanie z podstawowymi elementami cyfrowymi, stanowiącymi podstawowe składniki budowy komputera i metodami ich realizacji.
2. Zaznajomienie z układami liczącymi, rejestrami przesuwными, układami arytmetyczno-logicznymi oraz układami peryferyjnymi architektur systemów komputerowych.
3. Zaznajomienie z budową procesora i komputera bazującego na architekturze von Neumanna.
4. Zapoznanie ze współczesnymi systemami komputerowymi, nowoczesnymi architekturami, mocą obliczeniową i perspektywami rozwoju.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	Wiedza na temat podstawowych elementów wchodzących w skład systemu komputerowego i znajomość sposobu ich działania. Wiedza na temat zasady działania procesora oraz systemu komputerowego opartego na architekturze von Neumanna. Wiedza dotycząca współczesnych architektur systemów komputerowych.	I1_W03
Umiejętności Absolwent potrafi:		
EU1	Umiejętność zaprojektowania poszczególnych układów systemu komputerowego takich jak: układy czasowo-licznikowe, rejestry przesuwne, układy arytmetyczno-logiczne.	I1_U23
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Projektowanie elementów systemu komputerowego indywidualnie i w zespole wg. określonego harmonogramu.	I1_K03

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
4	6	E	18	0	0	18	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
-----	-------------	----------------	---------------

1	W	Projektowanie układów o programowalnej funkcjonalności, liczniki o programowalnej pojemności, układy transmisji szeregowej i równoległej jako przykłady magistral procesora i komputera. Arytmetyka komputerów, metody reprezentacji liczb całkowitych w różnych systemach binarnych, a w szczególności binarny kod uzupełnień do dwóch. Funkcjonalność jednostki arytmetyczno-logicznej na przykładzie układów półsumatora i sumatora, układ dodawania/odejmowania liczb całkowitych, sumator szeregowy, sumator równoległy, algorytmy mnożenia liczb binarnych, schematy blokowe - zasada działania. Algorytm mnożenia liczb w zapisie U2, schemat blokowy zasada działania, algorytmy dzielenia liczb binarnych, budowa i zasada działania jednostki arytmetyczno-logicznej procesora i sumatora akumulacyjnego. Budowa procesora, budowa procesora opartego na architekturze von Neumanna, podstawowe jego elementy i ich przeznaczenie, formaty i typy rozkazów procesora, wpływ długości rozkazów procesora na wielkość zajmowanej pamięci przez program oraz na jego czas wykonania, tryby adresowania: adresowanie natychmiastowe, adresowanie bezpośrednie, adresowanie rejestrowe, adresowanie indeksowe, adresowanie bazowe, adresowanie stosowe. Cykl pracy procesora, cykl pobrania, cykl adresowania pośredniego, cykl wykonawczy, cykl przerwania. Budowa i zasady projektowania jednostki sterującej, jednostka sterowana mikroprogramem, jednostka sterowana układowo. Budowa i funkcjonowanie procesora, procesory oparte na zmodyfikowanej architekturze von Neumanna, procesor potokowy, procesor CISC, procesor RISC, procesor wektorowy. Budowa komputera, system przerwań i jego rola w systemie komputerowym, typy przerwań - programowe, zewnętrzne, wewnętrzne, mechanizm obsługi przerwania. Tryby przesyłania danych w systemie komputerowym, przesyłanie danych zrealizowane programowo, przesyłanie danych inicjowane przerwaniem, przesłanie danych zrealizowane przez bezpośredni dostęp do pamięci. Budowa magistrali i jej przeznaczenie.	18
2	LK	Budowa podstawowych elementów logicznych wykorzystywanych w systemach komputerowych. Projekt układu, obliczenia algebraiczne, wykonanie układu w dedykowanym oprogramowaniu symulacyjnym, sprawdzenie działania. Budowa układów o programowalnej funkcjonalności, liczniki o programowalnej pojemności, układy transmisji szeregowej i równoległej jako przykłady magistral procesora i komputera. Projekt, obliczenia, wykonanie i testowanie rozwiązania. Budowa układów stanowiących przykładowe elementy funkcjonalne jednostki arytmetyczno-logicznej w różnych systemach binarnych - półsumator, sumator, układ mnożący. Projektowanie i budowa układów licznikowych, dwójka licząca, liczniki binarne, liczniki BCD, liczniki o zadanej pojemności, liczniki rewersyjne, licznik z wejściem dodającym i odejmującym, układ antykoincydencyjny. Projektowanie i budowa układów kodowania, szyfrowania, układów o zadanej specyficznej funkcjonalności.	9
3	LK	Mikroprocesory: Wstęp do programowania w asemblerze 8051, Obsługa portów wejścia/wyjścia mikrokontrolera, Liczniki i timery, Przerwania, Obsługa portu szeregowego	9

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Konsultacje przedmiotowe (w bezpośrednim kontakcie z prowadzącym)	34
2	Egzamin i zaliczenia (w bezpośrednim kontakcie z prowadzącym)	20

3	Praca własna - przygotowanie do zajęć, przygotowanie sprawozdań	60
---	---	----

Metody dydaktyczne:

konsultacje, wykład, laboratorium komputerowe, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, egzamin pisemny, odpowiedź ustna, test, sprawozdanie

Kryteria oceny:

1. Na ocenę 5,0 student nabył wiedzy treści wykładowych w stopniu co najmniej 90%
2. Na ocenę 5,0 student zalicza ćwiczenia laboratoryjne uzyskując średnią arytmetyczną ocen co najmniej 4,75
3. Na ocenę 5,0 student wykonuje zadania i oddaje je bez opóźnienia. W pracy zespołowej swoją część pracy wykonuje na czas.

Literatura:

obowiązkowa:

1. W. Stallings - Organizacja i architektura systemu komputerowego, Warszawa, 2022, PWN
2. Atmel 8051 Microcontrollers Hardware Manual

zalecana/fakultatywna:

1. Biernat J - Architektura komputerów, Wydawnictwa Naukowo-Techniczne
2. Biernat J. - Arytmetyka Komputerów,, Wydawnictwa Naukowo-Techniczne

Bazy danych I nazwa przedmiotu
Introduction to Database Management Systems nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Znajomość języka angielskiego umożliwiającą korzystanie z literatury i dokumentacji
2. Podstawowa znajomość obsługi systemu operacyjnego

Cele przedmiotu:

1. Zaznajomienie studentów z popularnymi relacyjnymi systemami zarządzania bazami danych: MySQL/MariaDB, w tym programami klienckimi
2. Zaznajomienie studentów z językiem SQL, w tym z grupami poleceń SQL: DQL, DML, DDL, TCL, DCL
3. Umiejętność tworzenia podstawowych obiektów w relacyjnym systemie zarządzania bazą danych
4. Wiedza z zakresu optymalizacji systemu bazodanowego

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW2	Student po ukończeniu kursu posiadać będzie szczegółową wiedzę teoretyczną na temat systemów baz danych, ze szczególnym uwzględnieniem MariaDB. Zrozumie fundamentalne koncepcje relacyjnych baz danych, w tym model relacyjny, język SQL, normalizację, transakcje, mechanizmy zapewniania spójności i bezpieczeństwa danych. Student zostanie wprowadzony w specyfikę MariaDB, w tym jej architekturę, zaawansowane funkcje, różnice w porównaniu z innymi systemami zarządzania bazami danych oraz jej integrację z aplikacjami webowymi. Nabędzie również wiedzę na temat optymalizacji zapytań i wydajności bazy danych, co umożliwi mu projektowanie i utrzymanie efektywnych, bezpiecznych i skalowalnych systemów bazodanowych. Zrozumienie teoretycznych aspektów baz danych, z naciskiem na MariaDB, przygotuje studenta do praktycznego zastosowania tej wiedzy w realnych scenariuszach informatycznych, zwiększając jego kompetencje jako specjalisty ds. baz danych.	I1_W08

Umiejętności Absolwent potrafi:		
EU4	Absolwent jest gotów do: dokonywania samooceny własnych kompetencji w zakresie systemów bazodanowych, wyznaczania kierunków własnego rozwoju i kształcenia w w/w zakresie, ponieważ rozumie potrzebę ciągłego dokształcania się zawodowego i rozwoju osobistego	I1_U02
EU5	Student po zakończeniu kursu będzie posiadał umiejętności niezbędne do efektywnego tworzenia diagramów ER (Entity-Relationship), które są fundamentem w projektowaniu schematów baz danych. Będzie zdolny do identyfikacji i modelowania encji, ich atrybutów oraz relacji między nimi, co pozwoli na precyzyjne odwzorowanie struktury danych wymaganej przez systemy informatyczne. Student nauczy się stosować zasady normalizacji w celu optymalizacji schematu bazy danych, co przyczyni się do zwiększenia wydajności i spójności danych. Ponadto, zdobędzie umiejętność interpretacji istniejących schematów ER, co umożliwi analizę i modyfikację baz danych w celu ich adaptacji do zmieniających się wymagań biznesowych i technologicznych. Nabyte kompetencje pozwolą studentowi na projektowanie skutecznych i skalowalnych baz danych, stanowiących solidne fundamenty dla aplikacji i systemów informatycznych.	I1_U09
EU6	Student po ukończeniu kursu będzie posiadał praktyczne umiejętności w zakresie projektowania, implementacji i zarządzania prostymi systemami bazodanowymi, korzystając z co najmniej jednego z najpopularniejszych systemów zarządzania bazą danych, takich jak MySQL lub MariaDB. Zdobędzie wiedzę na temat tworzenia efektywnych schematów bazy danych, w tym projektowania diagramów ERD (Entity-Relationship Diagram), które są kluczowe dla zrozumienia i wizualizacji struktury danych i relacji między nimi. Student nauczy się również podstawowych operacji na bazach danych, w tym tworzenia, modyfikacji i zapytań SQL, co pozwoli na efektywne zarządzanie danymi i dostosowywanie systemów bazodanowych do konkretnych potrzeb aplikacji. Dodatkowo, kurs wyposaży studenta w umiejętności oceny wydajności systemów bazodanowych oraz implementacji podstawowych zasad bezpieczeństwa danych.	I1_U16
Kompetencje społeczne Absolwent jest gotów do:		

EK3	Student po ukończeniu kursu będzie posiadał umiejętność identyfikacji i analizy głównych przyczyn awarii i nieefektywności w systemach baz danych, które mogły przyczynić się do poważnych strat finansowych oraz negatywnych konsekwencji społecznych. Zdobyta wiedza na temat projektowania, implementacji oraz utrzymania baz danych pozwoli studentowi na zrozumienie, jak błędy w tych obszarach mogą prowadzić do krytycznych awarii systemów informatycznych. Nabyte kompetencje pozwolą na podnoszenie świadomości o znaczeniu odpowiedzialnego zarządzania danymi i systemami informatycznymi w kontekście zapobiegania stratom finansowym i minimalizowania ryzyka negatywnych skutków społecznych.	I1_K01
EK4	Student po ukończeniu kursu będzie zdolny do efektywnej pracy zarówno indywidualnie, jak i w zespole programistów i specjalistów ds. baz danych. Będzie potrafił skutecznie zarządzać własnym czasem w kontekście realizacji projektów bazodanowych, co obejmuje planowanie zadań, ustalanie priorytetów oraz terminowe wykonywanie zobowiązań. Student nabędzie umiejętność adaptacji do zmieniających się wymagań projektowych i będzie zdolny do konstruktywnej współpracy z zespołem, w tym komunikacji dotyczącej problemów technicznych, rozwiązań i postępów w pracy. Dodatkowo, zdobędzie kompetencje w zakresie efektywnego rozwiązywania problemów i podejmowania decyzji, co jest kluczowe przy projektowaniu i implementacji systemów baz danych.	I1_K03

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
3	6	E	18	9	0	18	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
18	C	Omówienie tematyki projektów. Utworzenie grup projektowych	2
21	C	Oprogramowanie elementów funkcjonalnych - omówienie modelu MVC w praktyce.	5
22	C	Analiza poprawności projektów bazodanowych. Korekty schematów	2
23	LK	Tworzenie schematów tabel z uwzględnieniem relacji i typów danych, w celu zamodelowania typowych problemów świata zewnętrznego.	3
24	LK	Tworzenie tabel: - za pomocą programu klienckiego - za pomocą poleceń SQL. Polecenia SQL DDL, klucze główne, klucze obce, referencje z akcjami na referencjach, indeksy, Wersjonowanie tabel. Partycjonowanie tabel.	5

25	LK	Polecenia SQL: DQL, DML. Umieszczanie danych w tabelach, usuwanie danych, odczytywanie danych. Zapytania select, podzapytania, operatory: any, all, in, exists. Operatory komparacji (np. between and, is, like). Funkcje: matematyczne (np. abs, sqrt, mod, round), ciągowe (np. concat, instr, lower), dotyczące daty i czasu (np. to_date, datediff, timediff). Funkcje agregujące (avg, count, min, max, sum), operator group by, having.	7
26	LK	Złączenia: - wewnętrzne (inner) - operator using, złączenie naturalne (natural join), - zewnętrzne (outer) - left, right, full. Widoki. Unie.	3
27	W	Zasady zaliczenia przedmiotu. Przedstawienie zakresu materiału. Omówienie konfiguracji połączenia z systemem zarządzania bazą danych. Omówienie popularnych programów klienckich. Charakterystyka popularnych rodzajów systemów bazodanowych (np. relacyjnych, obiektowych, relacyjno-obiektowych, nierelacyjnych). Omówienie cech relacyjnego systemu zarządzania bazą danych. Podstawowe różnice pomiędzy wybranymi systemami, tj.: Oracle, MySQL/MariaDB, Microsoft SQL Server, SQLite	1
28	W	Projektowanie efektywnych schematów tabel w relacyjnych bazach danych, podkreślając znaczenie kluczy głównych (PK) i obcych (FK) w zapewnianiu integralności danych, rodzaje diagramów ERD. Zostaną omówione referencje między tabelami i ich znaczenie dla relacji oraz różne akcje na referencjach, takie jak kaskadowe usuwanie czy aktualizacja, które są kluczowe dla utrzymania spójności danych. Ponadto, szczegółowe wyjaśnienie procesu normalizacji bazy danych, mające na celu redukcję redundancji i eliminację anomalii danych przez organizację danych w tabelach według określonych reguł. Normalizacja jest przedstawiona jako krytyczny element w projektowaniu schematów tabel, pozwalający na osiągnięcie optymalnej struktury bazy danych, co zapewnia większą wydajność i łatwość w zarządzaniu danymi. Na koniec, omówiona zostanie denormalizacja jako proces stosowany w celu optymalizacji wydajności zapytań, przez świadome wprowadzanie redundancji danych w specyficznych przypadkach. Denormalizacja jest przedstawiona jako technika, która, stosowana z rozważą, może znacząco poprawić czas odpowiedzi systemu, szczególnie w dużych i złożonych bazach danych, gdzie wymagania dotyczące szybkiego dostępu do danych przeważają nad potrzebą ich ścisłej normalizacji.	2
29	W	Wykład przedstawia kompleksowe wprowadzenie do języka SQL, poczynwszy od jego historii i roli w zarządzaniu relacyjnymi bazami danych. Omówione zostaną główne kategorie języka SQL: DML (Data Manipulation Language) służący do manipulacji danymi, DQL (Data Query Language) do zapytań danych, DDL (Data Definition Language) do definicji struktur danych, DCL (Data Control Language) do kontroli dostępu do danych, oraz TCL (Transaction Control Language) do zarządzania transakcjami. Dodatkowo, kurs podejmie temat typów danych w MariaDB, w tym zarówno standardowych typów danych, jak i specyficznych dla MariaDB, podkreślając ich zastosowanie i najlepsze praktyki w projektowaniu bazy danych. Uczestnicy zyskają wiedzę na temat wyboru odpowiednich typów danych dla różnych scenariuszy, co jest kluczowe dla optymalizacji wydajności i skuteczności przechowywania danych.	1

30	W	Wykład poświęcony wbudowanym funkcjom matematycznym, ciągowym oraz funkcjom operującym na datach i czasie, które są niezbędne do efektywnego przetwarzania i analizy danych w bazach. Omówione zostaną najczęściej używane funkcje, takie jak operacje arytmetyczne, manipulacja ciągami znaków oraz konwersja i formatowanie dat i czasów, aby uczestnicy mogli skutecznie zarządzać danymi i wydobywać z nich wartościowe informacje. Ponadto, przedstawione zostaną operatory komparacji danych, umożliwiające wykonanie zaawansowanych zapytań i analiz w MariaDB. Przez połączenie teoretycznych podstaw z praktycznymi przykładami, studenci zdobędą umiejętności niezbędne do wykorzystywania bogatego zestawu narzędzi MariaDB do rozwiązywania realnych problemów bazodanowych.	1
31	W	Tematyka wykładu skupi się na kluczowych funkcjach agregujących, m.in. takich jak: średnia (avg), suma (sum), minimum (min), maksimum (max) oraz licznik (count), które są nieodzownymi narzędziami w analizie i przetwarzaniu danych zgromadzonych w bazach danych. Omówione zostaną praktyczne zastosowania tych funkcji w różnorodnych scenariuszach biznesowych i naukowych, umożliwiając uczestnikom efektywne podsumowywanie i analizowanie dużych zbiorów danych. Dodatkowo, wykład przedstawi operacje pionowe: union (połączenie), intersect (przecięcie) oraz except (różnica) - w aspekcie MariaDB, które pozwalają na efektywne manipulowanie i łączenie zestawów wyników zapytań. Przez zrozumienie i stosowanie tych operacji, uczestnicy nauczą się budować zaawansowane zapytania, które umożliwiają głębszą analizę i porównanie danych. Wykład ten wyposaży słuchaczy w umiejętności niezbędne do wykorzystywania funkcji agregujących i operacji pionowych w celu uzyskania wartościowych informacji z danych.	1
32	W	Podczas tego wykładu szczegółowo omówione zostaną różne rodzaje złączeń (join) w bazach danych, w tym złączenia wewnętrzne (inner join), zewnętrzne (left, right oraz full outer join), zastosowanie operatora USING dla uproszczenia zapytań z złączeniami na wspólnych kolumnach, oraz złączenia naturalne (natural join), które automatycznie łączą tabeli na podstawie kolumn o tych samych nazwach. Te techniki są kluczowe dla efektywnego łączenia danych z różnych tabel, umożliwiając tworzenie złożonych zapytań i raportów. Dodatkowo, wykład wprowadzi pojęcie widoków (views) w bazach danych. Widoki są wirtualnymi tabelami, które są wynikiem zapytania i służą do zapisywania skomplikowanych zapytań dla łatwiejszego ponownego użycia. Wyjaśnione zostanie to, dlaczego widoki są ważne, czyli, że pozwalają na uproszczenie dostępu do danych, zapewniają dodatkową warstwę abstrakcji i bezpieczeństwa, a także umożliwiają efektywne zarządzanie zmianami w strukturze baz danych bez wpływu na aplikacje korzystające z tych danych. Poprzez połączenie teorii z praktycznymi przykładami, studenci zdobędą umiejętności niezbędne do wykorzystywania złączeń i widoków do tworzenia zaawansowanych zapytań i zarządzania danymi w bardziej efektywny sposób.	1

33	W	Wykład poświęcony zrozumieniu i praktycznemu zastosowaniu podzapytań w MariaDB, wraz z omówieniem operatorów takich jak ANY, ALL, IN oraz EXISTS. Podzapytania, będące zapytaniami zagnieżdżonymi wewnątrz innego zapytania, są potężnym narzędziem umożliwiającym wykonanie skomplikowanych operacji na danych, takich jak filtrowanie, selekcja czy obliczenia, w oparciu o wyniki innych zapytań. Omówione zostaną różne rodzaje podzapytań, w tym te występujące w klauzuli WHERE, FROM oraz SELECT, oraz pokazane zostanie, jak mogą one zwiększać czytelność i wydajność zapytań. Wykład ten zapewni uczestnikom niezbędne narzędzia do budowania bardziej złożonych i efektywnych zapytań SQL, wykorzystując zaawansowane techniki podzapytań i operatorów w MariaDB.	1
34	W	W ramach tego wykładu zostanie omówione programowanie w języku SQL/PSM (Persistent Stored Modules), które pozwala na tworzenie zaawansowanych procedur i funkcji w bazie danych MariaDB. Przedstawione zostaną konstrukcje programistyczne takie jak pętle, instrukcje warunkowe oraz obsługa wyjątków, które są fundamentem dla tworzenia elastycznych i wydajnych procedur składowanych i funkcji. Szczególna uwaga zostanie poświęcona różnym rodzajom parametrów (in, out, inout) wykorzystywanym w procedurach, co umożliwia efektywne zarządzanie danymi wejściowymi i wynikami. Dodatkowo, zaprezentowane zostaną techniki tworzenia własnych funkcji, w tym funkcji agregujących, które rozszerzają możliwości standardowych operacji SQL o specyficzne, dostosowane do potrzeb użytkownika operacje na danych. Ten wykład pozwoli uczestnikom zrozumieć, jak wykorzystać pełnię możliwości języka SQL/PSM w MariaDB do budowania zaawansowanych rozwiązań bazodanowych.	1
35	W	Wykład poświęcony wyzwalaczom (triggerom) w MariaDB, czyli narzędziom, które pozwalają na automatyczne wykonywanie określonych operacji w bazie danych w odpowiedzi na zmiany (insert, update, delete). Studenci zostaną wprowadzeni w koncept wyzwalaczy, w tym ich zastosowanie, korzyści i potencjalne pułapki. Szczególna uwaga zostanie poświęcona czasom wykonania wyzwalaczy, czyli różnicom pomiędzy wyzwalaczami 'before' (przed operacją na danych) i 'after' (po operacji na danych), oraz ich wpływowi na integralność i logikę przetwarzania danych. Poprzez praktyczne przykłady uczestnicy nauczą się, jak projektować i implementować wyzwalacze do automatyzacji zadań, zarządzania złożonymi regułami biznesowymi oraz monitorowania i reagowania na zmiany danych. Celem wykładu jest zapewnienie głębokiego zrozumienia mechanizmu wyzwalaczy w MariaDB i umożliwienie studentom skutecznego wykorzystania tej funkcjonalności w projektach bazodanowych.	1

36	W	Ten wykład zapozna uczestników z fundamentalnymi zasadami zarządzania transakcjami w MariaDB, kładąc szczególny nacisk na praktyczne zastosowanie modelu ACID (Atomowość, Spójność, Izolacja, Trwałość) w celu zapewnienia niezawodności i bezpieczeństwa operacji bazodanowych. Omówione zostaną mechanizmy transakcyjne, w tym sposoby ich inicjowania, zarządzania zmianami oraz potwierdzania lub wycofywania operacji, by zagwarantować integralność danych nawet w przypadku błędów lub awarii. Wykład szczegółowo przedstawi również system blokad rekordów, niezbędny do utrzymania izolacji transakcji, w tym 'for update' i 'lock in share mode', które kontrolują dostęp do danych w scenariuszach konkurencyjnych. Przez połączenie teorii z praktycznymi przykładami, studenci zdobędą umiejętności niezbędne do efektywnego wykorzystania transakcji i mechanizmów blokad w MariaDB, co jest kluczowe dla zapewnienia spójności danych i zapobiegania konfliktom dostępu.	1
37	W	Wykład skupi się na omówieniu wersjonowania tabel w MariaDB, techniki umożliwiającej przechowywanie historii zmian danych w tabelach. Studenci dowiedzą się, dlaczego wersjonowanie jest ważne w aspekcie funkcjonowania całości oprogramowania opartego o bazę danych. Dowiedzą się w jaki sposób odtworzyć dane po niezamierzonych modyfikacjach lub usunięciach. Zostaną przedstawione praktyczne zastosowania wersjonowania, w tym jak może ono wspierać procesy biznesowe poprzez zapewnienie większej przejrzystości operacji na danych i ułatwienie śledzenia historii zmian. Dodatkowo, omówione zostanie, jak szybkie gromadzenie danych w tabelach wersjonowanych wpływa na wydajność i zarządzanie przestrzenią dyskową, co naturalnie prowadzi do konieczności optymalizacji za pomocą partycjonowania. Wprowadzenie do partycjonowania tabel będzie kluczowym elementem wykładu, podkreślającym, jak poprzez odpowiednie strategie partycjonowania można zoptymalizować zarządzanie dużymi zbiorami danych wersjonowanych, zwiększając wydajność zapytań i zarządzanie danymi. Ten wykład zapewni studentom wiedzę i narzędzia niezbędne do efektywnego wykorzystania wersjonowania tabel w MariaDB oraz wprowadzi do kolejnego wykładu związanego z zaawansowanymi technikami partycjonowania, mającymi na celu optymalizację i zarządzanie wydajnością baz danych.	1
38	W	W trakcie tego wykładu zostanie omówione partycjonowanie tabel w MariaDB jako kluczowa technika zarządzania i optymalizacji dużych zbiorów danych. Uczestnicy poznają różne techniki partycjonowania poziomego - według zakresu, listy, klucza, czasu, czy podziału hash - i zrozumieją, jak każda z nich może być wykorzystana do rozwiązania konkretnych problemów związanych z wydajnością. Szczególny nacisk położymy na praktyczne aspekty optymalizacji wydajności zapytań poprzez partycjonowanie, w tym na mechanizmach pruningu partycji i selekcji, które pozwalają na automatyczne pomijanie nieistotnych partycji podczas wykonywania zapytań. Dzięki pruningowi, system baz danych może ograniczyć obszar poszukiwań do partycji zawierających relewantne dane, co znacząco zwiększa szybkość zapytań i efektywność przetwarzania danych. Wykład przedstawi także, jak odpowiednio zaprojektowane partycjonowanie wspiera zarządzanie przestrzenią dyskową, poprawia czas dostępu do danych i ułatwia zarządzanie dużymi ilościami informacji. Poprzez analizę przypadków użycia i najlepszych praktyk, studenci zdobędą umiejętności niezbędne do efektywnego implementowania strategii partycjonowania w swoich projektach bazodanowych w MariaDB.	1

39	W	W ramach tego wykładu omówione zostaną mechanizmy zdarzeń w MariaDB, które umożliwiają automatyzację i harmonogramowanie zadań w bazie danych. Studenci nauczą się, jak definiować i zarządzać eventami, aby wykonywać regularne operacje, takie jak czyszczenie danych, aktualizacje statystyk czy reorganizacja partycji, bez potrzeby interwencji manualnej. Szczególna uwaga zostanie poświęcona automatyzacji procesu reorganizacji partycji, co jest kluczowe dla utrzymania optymalnej wydajności i zarządzania przestrzenią dyskową w dużych systemach bazodanowych. Przykłady praktyczne pokażą, jak za pomocą zdarzeń można automatycznie dodawać, usuwać lub modyfikować partycje w tabelach partycjonowanych, co pozwala na efektywne zarządzanie cyklem życia danych. Ten wykład zapewni uczestnikom wiedzę i narzędzia niezbędne do wykorzystania zdarzeń w MariaDB do automatyzacji zadań administracyjnych i optymalizacji działań bazodanowych.	1
40	W	Wykład skupi się na zaawansowanym zastosowaniu tzw. Wspólnych Wyrażeń Tablicowych (Common Table Expressions, CTE) w MariaDB, umożliwiających budowanie złożonych zapytań w bardziej czytelny i modularny sposób. Szczególna uwaga zostanie poświęcona CTE rekurencyjnemu, czyli narzędziu umożliwiającemu rozwiązywanie problemów, które zwykle wymagałyby iteracji lub rekursji, takich jak przetwarzanie hierarchicznych struktur danych lub generowanie sekwencji. Przez praktyczne przykłady uczestnicy nauczą się, jak efektywnie wykorzystywać CTE do uproszczenia złożonych operacji na danych, poprawy czytelności zapytań oraz realizacji operacji, które byłyby trudne lub niemożliwe do wykonania za pomocą standardowych zapytań SQL. Dodatkowo, zostaną omówione najlepsze praktyki i potencjalne pułapki związane z wykorzystaniem CTE, zwłaszcza w kontekście wydajności zapytań i zapętlenia.	1
41	W	Wykład poświęcony będzie replikacji w bazach danych, z kluczowym naciskiem na MariaDB, jako niezbędnej technice zapewniającej wysoką dostępność, skalowalność oraz bezpieczeństwo danych. Uczestnicy zostaną wprowadzeni w podstawowe koncepcje replikacji, w tym różnice między replikacją synchroniczną a asynchroniczną, oraz zrozumieją, jak replikacja wpływa na wydajność i niezawodność systemów baz danych. Szczegółowo omówione zostaną metody replikacji dostępne w MariaDB, takie jak replikacja standardowa, czy replikacja oparta na galera cluster. Przez analizę przypadków użycia i scenariuszy implementacji, studenci nauczą się, jak skonfigurować i zarządzać replikacją w MariaDB, aby zapewnić ciągłość działania usług i ochronę danych przed awariami. Wykład ten nie tylko zwiększy zrozumienie uczestników na temat krytycznych aspektów replikacji w kontekście zarządzania danymi, ale także wyposaży ich w praktyczne umiejętności niezbędne do projektowania i utrzymania zaawansowanych, rozproszonych systemów bazodanowych.	1

42	W	Wykład rozszerza zakres omawianego materiału o praktyczne zastosowania funkcjonalności dedykowanych, w tym wyszukiwania pełnotekstowego (full-text search), umożliwiającego efektywne przeszukiwanie dużych zbiorów tekstowych oraz WKT GIS, które wspiera przechowywanie, przetwarzanie i analizę danych geograficznych. Poprzez konkretne przykłady zastosowań, studenci rozumieją, jak te dodatkowe funkcjonalności mogą być wykorzystywane do rozszerzenia możliwości aplikacji bazodanowych, zwiększając ich wartość i użyteczność w różnorodnych scenariuszach biznesowych i naukowych. Wykład zamyka zakres materiału omówionego w bieżącym semestrze, jednocześnie stanowiąc most ku przyszłym przedmiotom dotyczącym baz danych. Poprzez wprowadzenie do zaawansowanych funkcjonalności, takich jak wyszukiwanie pełnotekstowe i obsługa danych geoprzestrzennych, zachęca do odkrywania nowych funkcjonalności w obszarze baz danych, przygotowując grunt pod przyszłe wyzwania akademickie i zawodowe.	2
----	---	---	---

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury	35
2	Tworzenie projektu	30
3	Zadanie domowe: CTE - rekurencja. Indeks Full-Text. Obiekty i funkcje WKT GIS.	3
4	Zadanie domowe: Transakcje bazodanowe w praktyce. Polecenia SQL TCL.	4
5	Zadanie domowe: Tworzenie procedur i funkcji w: MySQL/MariaDB.	4
6	Zadanie domowe: Harmonogram zdarzeń. Tworzenie wyzwalaczy na tabelach.	2
7	Zadanie domowe: Optymalizacja tabel poprzez partycjonowanie. Wersjonowanie tabel i partycjonowanie tabel zwersjonowanych.	4
8	Zadanie domowe: Utworzenie schematów ERD na podstawie tabel i referencji PK-FK. Postacie normalne w praktyce.	4
9	Zadanie domowe: Zadanie dotyczące replikacji w bazach danych, z kluczowym naciskiem na MariaDB, jako niezbędnej technice zapewniającej wysoką dostępność, skalowalność oraz bezpieczeństwo danych. Podstawowe koncepcje replikacji, w tym różnice między replikacją synchroniczną a asynchroniczną, jak replikacja wpływa na wydajność i niezawodność systemów baz danych. Metody replikacji dostępne w MariaDB, takie jak replikacja standardowa, czy replikacja oparta na galera cluster.	4
10	Konsultacje przedmiotowe (w bezpośrednim kontakcie z prowadzącym)	10
11	Egzaminy i zaliczenia (w bezpośrednim kontakcie z prowadzącym)	5

Metody dydaktyczne:

ćwiczenia, dyskusja, konsultacje, praca w grupach, projekt, wykład, laboratorium komputerowe, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, dyskusja, egzamin ustny, kolokwium, odpowiedź ustna, projekt, rozwiązanie zadania problemowego, sprawozdanie z pracy zespołowej, kolokwium z zadań, praca własna-zadanie, quiz, test, rysunek, sprawozdanie

Kryteria oceny:

1. Ocena 5.0 - Student wykazuje się doskonałym zrozumieniem i umiejętnościami praktycznymi we wszystkich aspektach kursu, w tym w projektowaniu schematów tabel, generowaniu diagramów ERD, zrozumieniu i zastosowaniu normalizacji, silników bazodanowych, kluczy, indeksów, i operacji SQL. - Potrafi samodzielnie rozwiązywać zaawansowane problemy bazodanowe, w tym stosowanie funkcji użytkownika, procedur, zarządzanie transakcjami, partycjonowanie, oraz efektywne wykorzystanie zaawansowanych funkcji takich jak CTE, Full-Text. - Demonstruje głęboką wiedzę teoretyczną połączoną z umiejętnością jej praktycznego zastosowania. Przykłady zagadnień: Doskonałe zrozumienie i zastosowanie postaci normalnych w projektowaniu baz danych; samodzielne tworzenie zaawansowanych diagramów ERD; biegła praca z silnikami bazodanowymi MariaDB/MySQL; skuteczne stosowanie kluczy głównych, obcych, indeksów oraz kluczy złożonych; zaawansowane zapytania SQL obejmujące wszystkie typy poleceń (DQL, DML, DDL, TCL, DCL); głębokie zrozumienie i praktyczne zastosowanie typów danych, w tym blob i json; optymalizacja zapytań za pomocą funkcji i procedur użytkownika; zaawansowane zarządzanie transakcjami i poziomami izolacji; efektywne wykorzystanie partycjonowania tabel i replikacji danych.
Ocena 4.5 - Student posiada bardzo dobrą wiedzę i umiejętności praktyczne w większości obszarów kursu, ale może mieć drobne braki w zrozumieniu lub zastosowaniu niektórych bardziej zaawansowanych konceptów. - Wykazuje się umiejętnością samodzielnego rozwiązywania problemów i stosowania zaawansowanych funkcji z niewielkimi błędami lub pominięciami. Przykłady zagadnień: Bardzo dobre zrozumienie normalizacji i umiejętność tworzenia diagramów ERD; efektywne stosowanie kluczy i indeksów; dobry poziom znajomości i zastosowania poleceń SQL; zrozumienie i stosowanie większości typów danych; umiejętność tworzenia i stosowania podstawowych funkcji i procedur; rozumienie zasad transakcji i poziomów izolacji; praktyczne zastosowanie partycjonowania i podstaw replikacji.
Ocena 4.0 - Student posiada solidną wiedzę i umiejętności praktyczne w zakresie podstawowych i większości zaawansowanych tematów kursu. - Potrafi samodzielnie rozwiązywać problemy i stosować zaawansowane funkcje, choć może potrzebować okazjonalnej pomocy lub przypomnienia o najlepszych praktykach. Przykłady zagadnień: Solidne zrozumienie projektowania schematów tabel i podstawowych diagramów ERD; właściwe stosowanie kluczy i prostych indeksów; umiejętność realizacji zapytań SQL w zakresie DQL, DML i DDL; podstawowa znajomość typów danych i ich zastosowania; realizacja prostych funkcji i procedur; podstawowe zrozumienie i zastosowanie transakcji.
Ocena 3.5 - Student ma ogólne zrozumienie materiału i potrafi stosować podstawowe oraz niektóre zaawansowane funkcje i techniki. - Demonstruje zdolność do rozwiązywania standardowych problemów, ale może mieć trudności z bardziej skomplikowanymi zagadnieniami i potrzebuje dodatkowej praktyki lub instrukcji. Przykłady zagadnień: Ogólne zrozumienie projektowania schematów tabel; podstawowe zapytania SQL; ograniczone zrozumienie kluczy i indeksów; realizacja bardzo prostych funkcji i procedur; podstawowe rozumienie transakcji i partycjonowania bez głębokiej wiedzy o optymalizacji.
Ocena 3.0 - Student posiada podstawową wiedzę i umiejętności praktyczne w zakresie omawianego materiału. - Rozumie kluczowe koncepcje i potrafi wykonać proste zadania bazodanowe, ale wykazuje braki w zaawansowanych tematach i może mieć trudności z samodzielnym rozwiązywaniem problemów. Przykłady zagadnień: Podstawowa wiedza o schematach tabel i diagramach ERD; podstawowe zapytania SQL ograniczające się do najprostszych operacji DQL i DML; ograniczone rozumienie kluczy i indeksów; podstawowe zrozumienie typów danych bez szczegółowego zastosowania; minimalne wykorzystanie funkcji i procedur; ograniczone zrozumienie transakcji i brak zrozumienia zaawansowanych technik takich jak partycjonowanie czy replikacja.

2. Ocena 5.0 - Student wykazuje się doskonałym zrozumieniem i umiejętnościami praktycznymi we wszystkich aspektach kursu, w tym w projektowaniu schematów tabel, generowaniu diagramów ERD, zrozumieniu i zastosowaniu normalizacji, silników bazodanowych, kluczy, indeksów, i operacji SQL. - Potrafi samodzielnie rozwiązywać zaawansowane problemy bazodanowe, w tym stosowanie funkcji użytkownika, procedur, zarządzanie transakcjami, partycjonowanie, oraz efektywne wykorzystanie zaawansowanych funkcji takich jak CTE, Full-Text. - Demonstruje głęboką wiedzę teoretyczną połączoną z umiejętnością jej praktycznego zastosowania. Przykłady zagadnień: Doskonałe zrozumienie i zastosowanie postaci normalnych w projektowaniu baz danych; samodzielne tworzenie zaawansowanych diagramów ERD; biegła praca z silnikami bazodanowymi MariaDB/MySQL; skuteczne stosowanie kluczy głównych, obcych, indeksów oraz kluczy złożonych; zaawansowane zapytania SQL obejmujące wszystkie typy poleceń (DQL, DML, DDL, TCL, DCL); głębokie zrozumienie i praktyczne zastosowanie typów danych, w tym blob i json; optymalizacja zapytań za pomocą funkcji i procedur użytkownika; zaawansowane zarządzanie transakcjami i poziomami izolacji; efektywne wykorzystanie partycjonowania tabel i replikacji danych.
- Ocena 4.5 - Student posiada bardzo dobrą wiedzę i umiejętności praktyczne w większości obszarów kursu, ale może mieć drobne braki w zrozumieniu lub zastosowaniu niektórych bardziej zaawansowanych konceptów. - Wykazuje się umiejętnością samodzielnego rozwiązywania problemów i stosowania zaawansowanych funkcji z niewielkimi błędami lub pominięciami. Przykłady zagadnień: Bardzo dobre zrozumienie normalizacji i umiejętność tworzenia diagramów ERD; efektywne stosowanie kluczy i indeksów; dobry poziom znajomości i zastosowania poleceń SQL; zrozumienie i stosowanie większości typów danych; umiejętność tworzenia i stosowania podstawowych funkcji i procedur; rozumienie zasad transakcji i poziomów izolacji; praktyczne zastosowanie partycjonowania i podstaw replikacji.
- Ocena 4.0 - Student posiada solidną wiedzę i umiejętności praktyczne w zakresie podstawowych i większości zaawansowanych tematów kursu. - Potrafi samodzielnie rozwiązywać problemy i stosować zaawansowane funkcje, choć może potrzebować okazjonalnej pomocy lub przypomnienia o najlepszych praktykach. Przykłady zagadnień: Solidne zrozumienie projektowania schematów tabel i podstawowych diagramów ERD; właściwe stosowanie kluczy i prostych indeksów; umiejętność realizacji zapytań SQL w zakresie DQL, DML i DDL; podstawowa znajomość typów danych i ich zastosowania; realizacja prostych funkcji i procedur; podstawowe zrozumienie i zastosowanie transakcji.
- Ocena 3.5 - Student ma ogólne zrozumienie materiału i potrafi stosować podstawowe oraz niektóre zaawansowane funkcje i techniki. - Demonstruje zdolność do rozwiązywania standardowych problemów, ale może mieć trudności z bardziej skomplikowanymi zagadnieniami i potrzebuje dodatkowej praktyki lub instrukcji. Przykłady zagadnień: Ogólne zrozumienie projektowania schematów tabel; podstawowe zapytania SQL; ograniczone zrozumienie kluczy i indeksów; realizacja bardzo prostych funkcji i procedur; podstawowe rozumienie transakcji i partycjonowania bez głębokiej wiedzy o optymalizacji.
- Ocena 3.0 - Student posiada podstawową wiedzę i umiejętności praktyczne w zakresie omawianego materiału. - Rozumie kluczowe koncepcje i potrafi wykonać proste zadania bazodanowe, ale wykazuje braki w zaawansowanych tematach i może mieć trudności z samodzielnym rozwiązywaniem problemów. Przykłady zagadnień: Podstawowa wiedza o schematach tabel i diagramach ERD; podstawowe zapytania SQL ograniczające się do najprostszych operacji DQL i DML; ograniczone rozumienie kluczy i indeksów; podstawowe zrozumienie typów danych bez szczegółowego zastosowania; minimalne wykorzystanie funkcji i procedur; ograniczone zrozumienie transakcji i brak zrozumienia zaawansowanych technik takich jak partycjonowanie czy replikacja.
3. Ocena 5.0 Student wykazuje się wybitną zdolnością do pracy zarówno indywidualnej, jak i zespołowej, demonstrując wysoką efektywność w zarządzaniu czasem i zadaniami. Doskonale adaptuje się do zmieniających się wymagań, aktywnie współpracuje z zespołem,

wykazując się inicjatywą w rozwiązywaniu problemów i podejmowaniu decyzji. Jego działania przekładają się na znaczące wkłady w projekty bazodanowe, przewyższając oczekiwania. Ocena 4.5 Student skutecznie pracuje indywidualnie i w zespole, dobrze zarządza czasem, efektywnie planując i realizując zadania. Jest zdolny do adaptacji do nowych wymagań i aktywnie uczestniczy w życiu zespołu, komunikując się efektywnie i przyczyniając się do rozwiązywania problemów oraz podejmowania ważnych decyzji, choć może okazjonalnie potrzebować drobnych korekt. Ocena 4.0 Student efektywnie pracuje zarówno samodzielnie, jak i w grupie, wykazując się dobrą organizacją czasu. Adaptuje się do zmieniających się warunków projektowych i współpracuje z zespołem, choć może czasem wymagać wsparcia w trudniejszych sytuacjach. Aktywnie uczestniczy w procesie rozwiązywania problemów i podejmowania decyzji, co przekłada się na solidny wkład w realizację projektów bazodanowych. Ocena 3.5 Student jest zdolny do pracy indywidualnej i zespołowej, lecz może wykazywać pewne braki w efektywnym zarządzaniu czasem i adaptacji do zmieniających się wymagań. Jego współpraca z zespołem jest ogólnie pozytywna, ale czasami potrzebuje zachęty do aktywniejszego udziału w rozwiązywaniu problemów i podejmowaniu decyzji. Ocena 3.0 Student potrafi pracować indywidualnie i w zespole, ale wykazuje podstawowe umiejętności w zarządzaniu czasem i adaptacji do nowych wymagań. Jego wkład w pracę zespołową i komunikację może być ograniczony, co wymaga dodatkowego wsparcia i motywacji. Rozwiązywanie problemów i podejmowanie decyzji często odbywa się przy znacznym wsparciu ze strony innych.

4. Ocena 5.0 Student aktywnie i krytycznie ocenia swoje umiejętności i wiedzę w zakresie baz danych, dokładnie identyfikuje obszary do rozwoju oparte na solidnym zrozumieniu zaawansowanych tematów bazodanowych. Ma jasno określone cele rozwojowe i plan działania, który obejmuje ciągłe kształcenie się i śledzenie najnowszych trendów w technologiach baz danych. Wykazuje inicjatywę w poszukiwaniu możliwości nauki i rozwoju. Ocena 4.5 Student skutecznie ocenia swoje mocne i słabe strony w obszarze baz danych, identyfikując kluczowe obszary do rozwoju. Posiada plan rozwoju zawodowego, który uwzględnia dalsze kształcenie się, ale może jeszcze bardziej zwiększyć swoją aktywność w poszukiwaniu nowych źródeł wiedzy i możliwości nauki. Ocena 4.0 Student posiada ogólną świadomość swoich kompetencji w zakresie baz danych i rozumie znaczenie ciągłego kształcenia się. Określił obszary wymagające dalszego rozwoju i ma ogólny plan na dalszą naukę, choć może jeszcze bardziej skonkretyzować swoje cele i działania. Ocena 3.5 Student ma podstawową zdolność do samooceny w zakresie baz danych i rozumie potrzebę ciągłego kształcenia, ale jego analiza może być powierzchowna, a plan rozwoju nie w pełni rozwinięty lub niekonsekwentnie realizowany. Potrzebuje wsparcia w identyfikacji konkretnych kierunków nauki i rozwoju. Ocena 3.0 Student wykazuje ograniczoną zdolność do samooceny swoich kompetencji bazodanowych i ma ograniczone zrozumienie, jak kształcić się dalej w tym obszarze. Może mieć trudności z określeniem konkretnych obszarów do rozwoju lub brakować mu inicjatywy w planowaniu i realizacji działań edukacyjnych. Potrzebuje wyraźnych wskazówek i motywacji do ciągłego doskonalenia się.
5. Ocena 5.0 - Student wykazuje się doskonałym zrozumieniem i umiejętnościami praktycznymi we wszystkich aspektach kursu, w tym w projektowaniu schematów tabel, generowaniu diagramów ERD, zrozumieniu i zastosowaniu normalizacji, silników bazodanowych, kluczy, indeksów, i operacji SQL. - Potrafi samodzielnie rozwiązywać zaawansowane problemy bazodanowe, w tym stosowanie funkcji użytkownika, procedur, zarządzanie transakcjami, partycjonowanie, oraz efektywne wykorzystanie zaawansowanych funkcji takich jak CTE, Full-Text. - Demonstruje głęboką wiedzę teoretyczną połączoną z umiejętnością jej praktycznego zastosowania. Przykłady zagadnień: Doskonałe zrozumienie i zastosowanie postaci normalnych w projektowaniu baz danych; samodzielne tworzenie zaawansowanych diagramów ERD; biegła praca z silnikami bazodanowymi MariaDB/MySQL; skuteczne stosowanie kluczy głównych, obcych, indeksów oraz kluczy złożonych; zaawansowane

zapytania SQL obejmujące wszystkie typy poleceń (DQL, DML, DDL, TCL, DCL); głębokie zrozumienie i praktyczne zastosowanie typów danych, w tym blob i json; optymalizacja zapytań za pomocą funkcji i procedur użytkownika; zaawansowane zarządzanie transakcjami i poziomami izolacji; efektywne wykorzystanie partycjonowania tabel i replikacji danych.

Ocena 4.5 - Student posiada bardzo dobrą wiedzę i umiejętności praktyczne w większości obszarów kursu, ale może mieć drobne braki w zrozumieniu lub zastosowaniu niektórych bardziej zaawansowanych konceptów. - Wykazuje się umiejętnością samodzielnego rozwiązywania problemów i stosowania zaawansowanych funkcji z niewielkimi błędami lub pominięciami. Przykłady zagadnień: Bardzo dobre zrozumienie normalizacji i umiejętność tworzenia diagramów ERD; efektywne stosowanie kluczy i indeksów; dobry poziom znajomości i zastosowania poleceń SQL; zrozumienie i stosowanie większości typów danych; umiejętność tworzenia i stosowania podstawowych funkcji i procedur; rozumienie zasad transakcji i poziomów izolacji; praktyczne zastosowanie partycjonowania i podstaw replikacji.

Ocena 4.0 - Student posiada solidną wiedzę i umiejętności praktyczne w zakresie podstawowych i większości zaawansowanych tematów kursu. - Potrafi samodzielnie rozwiązywać problemy i stosować zaawansowane funkcje, choć może potrzebować okazjonalnej pomocy lub przypomnienia o najlepszych praktykach. Przykłady zagadnień: Solidne zrozumienie projektowania schematów tabel i podstawowych diagramów ERD; właściwe stosowanie kluczy i prostych indeksów; umiejętność realizacji zapytań SQL w zakresie DQL, DML i DDL; podstawowa znajomość typów danych i ich zastosowania; realizacja prostych funkcji i procedur; podstawowe zrozumienie i zastosowanie transakcji.

Ocena 3.5 - Student ma ogólne zrozumienie materiału i potrafi stosować podstawowe oraz niektóre zaawansowane funkcje i techniki. - Demonstruje zdolność do rozwiązywania standardowych problemów, ale może mieć trudności z bardziej skomplikowanymi zagadnieniami i potrzebuje dodatkowej praktyki lub instrukcji. Przykłady zagadnień: Ogólne zrozumienie projektowania schematów tabel; podstawowe zapytania SQL; ograniczone zrozumienie kluczy i indeksów; realizacja bardzo prostych funkcji i procedur; podstawowe rozumienie transakcji i partycjonowania bez głębokiej wiedzy o optymalizacji.

Ocena 3.0 - Student posiada podstawową wiedzę i umiejętności praktyczne w zakresie omawianego materiału. - Rozumie kluczowe koncepcje i potrafi wykonać proste zadania bazodanowe, ale wykazuje braki w zaawansowanych tematach i może mieć trudności z samodzielnym rozwiązywaniem problemów. Przykłady zagadnień: Podstawowa wiedza o schematach tabel i diagramach ERD; podstawowe zapytania SQL ograniczające się do najprostszych operacji DQL i DML; ograniczone rozumienie kluczy i indeksów; podstawowe zrozumienie typów danych bez szczegółowego zastosowania; minimalne wykorzystanie funkcji i procedur; ograniczone zrozumienie transakcji i brak zrozumienia zaawansowanych technik takich jak partycjonowanie czy replikacja.

6. Ocena 5.0 - Student wykazuje się doskonałym zrozumieniem i umiejętnościami praktycznymi we wszystkich aspektach kursu, w tym w projektowaniu schematów tabel, generowaniu diagramów ERD, zrozumieniu i zastosowaniu normalizacji, silników bazodanowych, kluczy, indeksów, i operacji SQL. - Potrafi samodzielnie rozwiązywać zaawansowane problemy bazodanowe, w tym stosowanie funkcji użytkownika, procedur, zarządzanie transakcjami, partycjonowanie, oraz efektywne wykorzystanie zaawansowanych funkcji takich jak CTE, Full-Text. - Demonstruje głęboką wiedzę teoretyczną połączoną z umiejętnością jej praktycznego zastosowania. Przykłady zagadnień: Doskonałe zrozumienie i zastosowanie postaci normalnych w projektowaniu baz danych; samodzielne tworzenie zaawansowanych diagramów ERD; biegła praca z silnikami bazodanowymi MariaDB/MySQL; skuteczne stosowanie kluczy głównych, obcych, indeksów oraz kluczy złożonych; zaawansowane zapytania SQL obejmujące wszystkie typy poleceń (DQL, DML, DDL, TCL, DCL); głębokie zrozumienie i praktyczne zastosowanie typów danych, w tym blob i json; optymalizacja zapytań za pomocą funkcji i procedur użytkownika; zaawansowane zarządzanie transakcjami

i poziomami izolacji; efektywne wykorzystanie partycjonowania tabel i replikacji danych.

Ocena 4.5 - Student posiada bardzo dobrą wiedzę i umiejętności praktyczne w większości obszarów kursu, ale może mieć drobne braki w zrozumieniu lub zastosowaniu niektórych bardziej zaawansowanych konceptów. - Wykazuje się umiejętnością samodzielnego rozwiązywania problemów i stosowania zaawansowanych funkcji z niewielkimi błędami lub pominięciami. Przykłady zagadnień: Bardzo dobre zrozumienie normalizacji i umiejętność tworzenia diagramów ERD; efektywne stosowanie kluczy i indeksów; dobry poziom znajomości i zastosowania poleceń SQL; zrozumienie i stosowanie większości typów danych; umiejętność tworzenia i stosowania podstawowych funkcji i procedur; rozumienie zasad transakcji i poziomów izolacji; praktyczne zastosowanie partycjonowania i podstaw replikacji.

Ocena 4.0 - Student posiada solidną wiedzę i umiejętności praktyczne w zakresie podstawowych i większości zaawansowanych tematów kursu. - Potrafi samodzielnie rozwiązywać problemy i stosować zaawansowane funkcje, choć może potrzebować okazjonalnej pomocy lub przypomnienia o najlepszych praktykach. Przykłady zagadnień: Solidne zrozumienie projektowania schematów tabel i podstawowych diagramów ERD; właściwe stosowanie kluczy i prostych indeksów; umiejętność realizacji zapytań SQL w zakresie DQL, DML i DDL; podstawowa znajomość typów danych i ich zastosowania; realizacja prostych funkcji i procedur; podstawowe zrozumienie i zastosowanie transakcji.

Ocena 3.5 - Student ma ogólne zrozumienie materiału i potrafi stosować podstawowe oraz niektóre zaawansowane funkcje i techniki. - Demonstruje zdolność do rozwiązywania standardowych problemów, ale może mieć trudności z bardziej skomplikowanymi zagadnieniami i potrzebuje dodatkowej praktyki lub instrukcji. Przykłady zagadnień: Ogólne zrozumienie projektowania schematów tabel; podstawowe zapytania SQL; ograniczone zrozumienie kluczy i indeksów; realizacja bardzo prostych funkcji i procedur; podstawowe zrozumienie transakcji i partycjonowania bez głębokiej wiedzy o optymalizacji.

Ocena 3.0 - Student posiada podstawową wiedzę i umiejętności praktyczne w zakresie omawianego materiału. - Rozumie kluczowe koncepcje i potrafi wykonać proste zadania bazodanowe, ale wykazuje braki w zaawansowanych tematach i może mieć trudności z samodzielnym rozwiązywaniem problemów. Przykłady zagadnień: Podstawowa wiedza o schematach tabel i diagramach ERD; podstawowe zapytania SQL ograniczające się do najprostszych operacji DQL i DML; ograniczone rozumienie kluczy i indeksów; podstawowe zrozumienie typów danych bez szczegółowego zastosowania; minimalne wykorzystanie funkcji i procedur; ograniczone zrozumienie transakcji i brak zrozumienia zaawansowanych technik takich jak partycjonowanie czy replikacja.

Literatura:

obowiązkowa:

1. SQL : przewodnik dla początkujących : jak zacząć efektywną pracę z danymi / Tytuł oryginału: SQL quickstart guide : the simplified beginner's guide to managing, analyzing, and manipulating data with SQL, 2019 Walter Shields ; przekład Agnieszka Górczyńska. - Gliwice : Helion, 2023

zalecana/fakultatywna:

1. Getting Started with MariaDB. Explore the powerful features of MariaDB with practical examples. Daniel Bartholomew. Helion, 2015
2. Tablice informatyczne. SQL. Marcin Szeliga. Helion, 2022
3. MariaDB Server Documentation. (<https://mariadb.com/kb/en/documentation/>)
4. SQL Tutorial w3schools.com. (<https://www.w3schools.com/sql/>)
5. MySQL 8.0 Reference Manual (<https://dev.mysql.com/doc/refman/8.0/en/>)

Bazy danych II nazwa przedmiotu
Data bases II nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Znajomość relacyjnych baz danych oraz języka SQL

Cele przedmiotu:

1. Poznanie zasad projektowania i zarządzania nierelacyjnymi systemami bazodanowymi
2. Poznanie języka PL/SQL na przykładzie systemu Oracle (system obiektowo-relacyjny)
3. Zapoznanie się z bazami NoSQLowymi ze szczególnym uwzględnieniem bazy dokumentowej MongoDB
4. Zapoznanie się z podstawami konteneryzacji

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		

EW1	Zna wybrane relacyjne i nierelacyjne systemy zarządzania bazą danych.	I1_W08
Umiejętności Absolwent potrafi:		
EU1	Potrafi posługiwać się językiem PL/SQL, korzystać z bazy dokumentowej MongoDB oraz zastosować mechanizm konteneryzacji w technologii Docker.	I1_U16
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Potrafi współpracować w grupie, uczestniczyć w dyskusjach, zaprezentować efekty realizowanych zadań.	I1_K03

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
4	4	E	18	0	0	18	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	System zarządzania bazą danych Oracle. Składnia języka PL/SQL. Zmienne rekordowe, kursory, pobieranie rekordu z kursora. Obsługa wyjątków w PL/SQL. Procedury, funkcje, pakiety, wyzwalacze.	6
2	W	NoSQLowe bazy danych. Bazy dokumentowe - MongoDB. Bazy kolumnowe - Cassandra. Rozproszone bazy danych, hurtownie danych. Obiektowe bazy danych.	6
3	W	Podstawy konteneryzacji w technologii Docker: obraz a kontener, kontenery i sieci, repozytorium hub.docker.com. Konfiguracja kontenerów w zastosowaniach bazodanowych, volumeny i porty, tworzenie własnych kontenerów. Konfiguracja wielu serwerów z wykorzystaniem docker-compose, środowisko rozproszone NoSQL w oparciu o kontenery, MongoDB. Replikacja i sharding w środowisku kontenerów MongoDB.	6
4	LK	Implementacja schematu koncepcyjnego bazy danych - uruchomienie skryptów tworzących obiekty bazodanowe wraz z wypełnieniem danymi. Rozwiązywanie zadań z wykorzystaniem języka SQL (zmienne rekordowe, kursory, zapytania, obsługa wyjątków, procedury, funkcje, wyzwalacze, pakiety).	6
5	LK	Rozwiązywanie zadań z wykorzystaniem bazy dokumentowej MongoDB.	6

6	LK	Skonfigurowanie środowiska MongoDB w oparciu o kontenery na maszynie wirtualnej. Wymiana danych między bazami w oparciu o driver PyMongo. Replikacja i Sharding konstrukcja i testy środowiska.	6
---	----	---	---

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Przygotowanie się do zajęć i dyskusji, w tym studiowanie zalecanej literatury, opracowanie wyników przeprowadzonych badań, przygotowanie raportu, projektu, prezentacji itp.	44
2	Konsultacje przedmiotowe (w bezpośrednim kontakcie z prowadzącym)	15
3	Egzaminy i zaliczenia (w bezpośrednim kontakcie z prowadzącym)	5

Metody dydaktyczne:

dyskusja, konsultacje, praca w grupach, wykład, laboratorium komputerowe, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, egzamin pisemny, kolokwium

Kryteria oceny:

1. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji wiedzy.
2. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji umiejętności.
3. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji kompetencji społecznych.

Literatura:

obowiązkowa:

1. Bill Pribyl, Steven Feuerstein, Oracle PL/SQL. Wprowadzenie, Warszawa, 2002, Helion
2. Sean P. Kane, Karl Matthias, Docker. Praktyczne zastosowania, Warszawa, 2019, Helion
3. Shannon Bradshaw, Eoin Brazil, Kristina Chodorow. Przewodnik po MongoDB. Wydajna i skalowalna baza danych., Warszawa, 2020, Helion

zalecana/fakultatywna:

1. Guy Harrison. NoSQL, NewSQL i BigData. Bazy danych następnej generacji (ebook)

Bezpieczeństwo systemów komputerowych

nazwa przedmiotu

Computer systems security nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Znajomość podstawowych protokołów sieciowych (m.in. Ethernet, ARP, RARP, ICMP, IP, TCP, UDP, DNS, HTTP, SMTP) i zasad działania sieci komputerowych.
2. Zaliczenie przedmiotu podstawy sieci komputerowych.
3. Znajomość języka angielskiego w stopniu umożliwiającym studiowanie literatury i uczestnictwo w zajęciach.

Cele przedmiotu:

1. Wprowadzenie w tematykę bezpieczeństwa systemów komputerowych.
2. Zapoznanie z tematyką zabezpieczania przesyłania informacji.
3. Zapoznanie studentów z metodami bezpiecznego łączenia geograficznie rozproszonych lokalizacji w logiczną wirtualną sieć.
4. Zapoznanie studentów z metodami zapewniania bezpiecznego zdalnego dostępu do zasobów sieci chronionej.
5. Zapoznanie studentów z technikami filtrowania ruchu sieciowego w warstwach od 3 wzwyż siedmiowarstwowego modelu sieci.
6. Zapoznanie studentów ze sposobami konfigurowania filtrów pakietów oraz proxy filtrujących.
7. Prezentacja metod realizacji redundancji systemów zabezpieczeń sieciowych.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
------------------------	-------------------------	-------------------------------------

Wiedza Absolwent zna i rozumie:		
EW1	podstawowe metody, techniki i narzędzia stosowane przy rozwiązywaniu prostych zadań informatycznych z zakresu budowy systemów komputerowych, sieci komputerowych i technologii sieciowych oraz systemów wbudowanych.	I1_W12
Umiejętności Absolwent potrafi:		
EU1	zabezpieczyć dane przed nieuprawnionym dostępem	I1_U13
Kompetencje społeczne Absolwent jest gotów do:		
EK1	zrozumienia przyczyny wadliwie działających systemów informatycznych, które doprowadziły do poważnych strat finansowych oraz społecznych.	I1_K01

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
6	4	E	18	0	0	18	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	Wstęp, omówienie zasad zaliczenia, omówienie podstawowych pojęć związanych z bezpieczeństwem	2
2	W	Podstawy kryptografii	2
3	W	Podpis cyfrowy, PKI, standard X.509	2
4	W	Bezpieczeństwo sieci komputerowych, omówienie zagrożeń poszczególnych warstw modelu OSI, przedstawienie koncepcji różnego rodzaju technik ataków	4
5	W	Prezentacja zagrożeń bezpieczeństwa, technik ataków i obrony	2
6	W	Firewall, VPN, PGP	2
7	W	IPSec, IKE, Kerberos	2
8	W	Bezpieczeństwo sieci bezprzewodowych	2

9	LK	Omówienie zasad, prezentacja sprzętu, zapoznanie z wykorzystywanym oprogramowaniem. Zapoznanie z WSM i FSM.	3
10	LK	Tworzenie polityk, logi. Tworzenie polityk proxy	3
11	LK	Serwer zarządzania, serwer raportów. Autentykacja użytkowników.	3
12	LK	MOVPN	3
13	LK	BOVPN, MultiWAN	3
14	LK	Troubleshooting	1
15	LK	Podsumowanie zajęć, kolokwium zaliczeniowe	2

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Konsultacje przedmiotowe w bezpośrednim kontakcie z prowadzącym	10
2	Egzaminy i zaliczenia w sesji w bezpośrednim kontakcie z prowadzącym	4
3	Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury	70

Metody dydaktyczne:

wykład, laboratorium komputerowe

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, egzamin pisemny, kolokwium, kolokwium z zadań

Kryteria oceny:

1. Na ocenę 5.0 student potrafi wymienić podstawowe zagrożenia systemów komputerowych, porównać poziom niebezpieczeństwa podstawowych zagrożeń systemów komputerowych, opisać i porównać różne typy ataków DoS, przedstawić podstawowe ataki kierowane przeciwko sieci Web, przedstawić scenariusze ataków na różne warstwy modelu OSI, wymienić podstawowe metody bezpiecznej transmisji danych, porównać zalety poszczególnych metod bezpiecznej transmisji danych, opisać zasadę działania protokołu IPSec, opisać zasadę działania, strukturę oraz zależności pomiędzy elementami infrastruktury PKI, opisać zasady ukrywania ruchu sieciowego w innych protokołach, wymienić podstawowe metody zabezpieczania systemów komputerowych, opisać zasadę działania zapory sieciowej, zaprezentować różnice pomiędzy poszczególnymi typami zapór sieciowych, zaprezentować zasadę działania oraz typy systemów ochrony przed intruzami (IDS), przedstawić typy ochrony antyspamowej oraz opisać ich zasadę działania, opisać zasadę działania Proxy filtrujących.
2. Na ocenę 5.0 student potrafi wymienić podstawowe zagrożenia systemów komputerowych, porównać poziom niebezpieczeństwa podstawowych zagrożeń systemów komputerowych, opisać i porównać różne typy ataków DoS, przedstawić podstawowe ataki kierowane przeciwko sieci Web, przedstawić scenariusze ataków na różne warstwy modelu OSI, wymienić podstawowe metody bezpiecznej transmisji danych, porównać zalety poszczególnych metod bezpiecznej transmisji danych, opisać zasadę działania protokołu IPSec, opisać zasadę działania, strukturę oraz zależności pomiędzy elementami infrastruktury PKI, opisać zasady ukrywania ruchu sieciowego w innych protokołach, wymienić podstawowe metody zabezpieczania systemów komputerowych, opisać zasadę działania zapory sieciowej, zaprezentować różnice pomiędzy poszczególnymi typami zapór sieciowych, zaprezentować zasadę działania oraz typy systemów ochrony przed intruzami (IDS), przedstawić typy ochrony

antyspamowej oraz opisać ich zasadę działania, opisać zasadę działania Proxy filtrujących. Student potrafi konfigurować podstawowe ustawienia sieciowe urządzeń klasy XTM, konfigurować filtry pakietów, konfigurować proxy filtrujące dla protokołów HTTP, HTTPS, DNS, FTP, konfigurować tunele VPN, analizować zebrane logi oraz generować raporty aktywności użytkowników, konfigurować polityki bezpieczeństwa bazujące na tożsamości użytkowników, konfigurować zaawansowane ustawienia sieciowe urządzeń, konfigurować MultiWAN, wyszukać i rozwiązać problemy występujące w konfiguracji urządzeń.

3. Na ocenę 5.0 student potrafi konfigurować podstawowe ustawienia sieciowe urządzeń klasy XTM, konfigurować filtry pakietów, konfigurować proxy filtrujące dla protokołów HTTP, HTTPS, DNS, FTP, konfigurować tunele VPN, analizować zebrane logi oraz generować raporty aktywności użytkowników, konfigurować polityki bezpieczeństwa bazujące na tożsamości użytkowników, konfigurować zaawansowane ustawienia sieciowe urządzeń, konfigurować MultiWAN, wyszukać i rozwiązać problemy występujące w konfiguracji urządzeń.

Literatura:

obowiązkowa:

1. WatchGuard?http://www.watchguard.com/help/docs/wsm/XTM_11/en-US/v11_9_WSM_User_Guide_(en-US).pdf, www, 2014, WatchGuard
2. William Stallings, Lawrie Brown ? Bezpieczeństwo systemów informatycznych. Zasady i praktyka. Wydanie IV. Tom 1 i 2, Gliwice, 2019, Helion
3. Libor Dostalek - Bezpieczeństwo protokołu TCP/IP: kompletny przewodnik,, Warszawa, 2006, PWN
4. Josef Pieprzyk, Thomas Hardjono, Jennifer Seberry - Teoria bezpieczeństwa systemów komputerowych, 2005, Helion

zalecana/fakultatywna:

1. Marek Serafin, Sieci VPN: zdalna praca i bezpieczeństwo danych, Gliwice, 2010, Helion

Fizyka nazwa przedmiotu
Physics nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

Nie określono wymagań wstępnych

Cele przedmiotu:

1. Zapoznanie studentów z podstawowymi zagadnieniami mechaniki klasycznej oraz wybranymi zagadnieniami elektrodynamiki i fizyki współczesnej.
2. Zapoznanie studentów z wybranymi metodami rozwiązywania prostych zadań i problemów ilustrujących modele i zagadnienia fizyczne.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	Student zna podstawowe zagadnienia i prawa mechaniki klasycznej.	I1_W02
EW2	Student zna podstawowe zagadnienia dotyczące: własności pól elektrycznego i magnetycznego oraz prądu elektrycznego, zna prawa elektrodynamiki i podstawowe własności fal elektromagnetycznych, potrafi wskazać ich wykorzystanie w technice	I1_W02
EW3	Student zna podstawowe zagadnienia z elementów fizyki współczesnej, głównie będące przygotowaniem do zrozumienia podstawowych zagadnień fizyki kwantowej	I1_W02
Umiejętności Absolwent potrafi:		
EU1	Student potrafi rozwiązywać proste zadania i problemy ilustrujące wybrane zagadnienia i modele z zakresu mechaniki, umie analizować i interpretować otrzymane wyniki.	I1_U01b
EU1	Student potrafi rozwiązywać proste zadania i problemy ilustrujące wybrane zagadnienia i modele z zakresu mechaniki, umie analizować i interpretować otrzymane wyniki.	I1_U02

EU1	Student potrafi rozwiązywać proste zadania i problemy ilustrujące wybrane zagadnienia i modele z zakresu mechaniki, umie analizować i interpretować otrzymane wyniki.	I1_U06b
EU2	Student potrafi rozwiązywać proste zadania i problemy ilustrujące wybrane zagadnienia i modele z zakresu elektrodynamiki, umie analizować i interpretować otrzymane wyniki.	I1_U01b
EU2	Student potrafi rozwiązywać proste zadania i problemy ilustrujące wybrane zagadnienia i modele z zakresu elektrodynamiki, umie analizować i interpretować otrzymane wyniki.	I1_U02
EU2	Student potrafi rozwiązywać proste zadania i problemy ilustrujące wybrane zagadnienia i modele z zakresu elektrodynamiki, umie analizować i interpretować otrzymane wyniki.	I1_U06b
EU3	Student potrafi rozwiązywać proste zadania i problemy ilustrujące wybrane podstawowe zagadnienia z fizyki współczesnej.	I1_U01b
EU3	Student potrafi rozwiązywać proste zadania i problemy ilustrujące wybrane podstawowe zagadnienia z fizyki współczesnej.	I1_U02
EU3	Student potrafi rozwiązywać proste zadania i problemy ilustrujące wybrane podstawowe zagadnienia z fizyki współczesnej.	I1_U06b
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Student rozumie wpływ znajomości zagadnień fizyki na rozwój społeczeństwa i cywilizacji.	I1_K06

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
2	4	Z	18	18	0	0	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	Wstęp do fizyki. Przedmiot i metody badawcze fizyki. Matematyka językiem fizyki. Notacja fizyczna i jednostki układu SI. Obserwacja, pomiar i model teoretyczny zjawisk. Mechanika klasyczna: Opis ruchu w różnych układach odniesienia. Klasyfikacja ruchów. Względność ruchu. Oddziaływania fundamentalne i pola fizyczne. Prawa dynamiki klasycznej Newtona. Układy inercjalne i nieinercjalne. Siły tarcia. Zasada zachowania pędu.	3
2	W	Praca i energia. Moc. Oscylator harmoniczny. Ruch z oporem powietrza. Zasada zachowania energii mechanicznej i całkowitej. Prawo powszechnego ciążenia. Pole grawitacyjne jako przykład pola zachowawczego.	3

3	W	Drgania harmoniczne. Superpozycja drgań. Ruch drgający tłumiony i wymuszony. Zjawisko rezonansu w fizyce. Opis i klasyfikacja fal. Fale harmoniczne. Klasyczne równanie falowe. Transport energii i natężenie fali. Zjawiska charakterystyczne dla fal: odbicie i załamanie, interferencja, dyfrakcja i polaryzacja fal.	3
4	W	Elektryczność i magnetyzm: Pole elektryczne i jego opis. Prawo Gaussa i jego zastosowania. Potencjał elektryczny. Pojemność. Prąd elektryczny. Pole magnetyczne, siła Lorentza.	3
5	W	Prawo Ampere'a i prawo Biota-Savarta. Indukcja elektromagnetyczna. Równania Maxwella i ich sens fizyczny. Równanie falowe dla fali elektromagnetycznej. Fale elektromagnetyczne i ich właściwości. Światło jako fala elektromagnetyczna. Polaryzacja światła.	3
6	W	Elementy fizyki współczesnej. Efekt fotoelektryczny. Dualizm korpuskularno-falowy. Eksperyment z dwiema szczelinami. Funkcje falowe. Równanie Schroedingera zależne i niezależne od czasu. Stany kwantowe. Qubity. Splątanie kwantowe.	3
7	C	Elementy rachunku wektorowego i analizy matematycznej. Obliczanie prędkości i przyspieszenia w ruchu prostoliniowym i krzywoliniowym w różnych układach odniesienia.	3
8	C	Rozwiązywanie równań ruchu dla prostych przykładów z dynamiki klasycznej. Opis ruchu w układach nieinercjalnych. Zagadnienia pracy i energii w polu sił. Zastosowanie zasad zachowania pędu, momentu pędu i energii mechanicznej w układach izolowanych.	3
9	C	Oscylator harmoniczny przykłady. Zastosowanie funkcji falowej do opisu zjawisk falowych. Superpozycja i interferencja fal, fale stojące. Funkcja falowa, jako rozwiązanie równania falowego.	3
10	C	Operatory różniczkowe: gradient, dywergencja, rotacja. Ruch ładunków w polach: elektrycznym i magnetycznym. Zastosowanie prawa Gaussa do wyznaczania pola elektrycznego. Pojemność. Prąd elektryczny.	3
11	C	Siła Lorentza. Proste przykłady zastosowania prawa Ampere'a i prawa Faradaya. Prawa Maxwella	3
12	C	Rozwiązywanie prostych problemów związanych z podstawami mechaniki kwantowej i szczególnej teorii względności.	3

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury	50
2	Konsultacje (w bezpośrednim kontakcie z prowadzącym)	14

Metody dydaktyczne:

ćwiczenia rachunkowe, wykład, ćwiczenia tablicowe

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, test, kolokwium z zadań

Kryteria oceny:

1. Na ocenę 5.0 student opanował materiał z wykładów nt mechaniki w stopniu co najmniej 90%
2. Na ocenę 5.0 student opanował materiał z wykładów nt elektryczności i magnetyzmu w stopniu co najmniej 90%
3. Na ocenę 5.0 student opanował materiał z wykładów nt podstawowych zagadnień fizyki współczesnej w stopniu co najmniej 90%
4. Na ocenę 5.0 student potrafi poprawnie rozwiązywać co najmniej 90% zadań dotyczących mechaniki
5. Na ocenę 5.0 student potrafi poprawnie rozwiązywać co najmniej 90% zadań dotyczących elektryczności i magnetyzmu

Literatura:

obowiązkowa:

1. D. Halliday, R. Resnick, J. Walker, Podstawy fizyki, Tomy 1,2,3,4,5
2. M. Massalska, J. Massalski, Fizyka dla inżynierów, część 1
3. A. A. Dietla, B. M. Jaworski, Fizyka. Poradnik Encyklopedyczny

zalecana/fakultatywna:

Grafika komputerowa nazwa przedmiotu
Computer graphics nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Podstawy programowania
2. Algebra i analiza matematyczna

Cele przedmiotu:

1. Zapoznanie z podstawowymi metodami w grafice komputerowej rastrowej i wektorowej oraz metodami komunikacji człowiek-komputer.
2. Implementacja wybranych metod graficznych oraz interfejsu użytkownika z wykorzystaniem biblioteki OpenGL
3. Praktyczne zastosowanie podstawowych operacji na obrazach rastrowych z wykorzystaniem pakietu PhotoShop lub GIMP
4. Praktyczne zastosowanie technik grafiki wektorowej do rysowania schematów technicznych 2D z wykorzystaniem oprogramowania AutoCAD.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW2	Obrazy rastrowe i wektorowe: budowa, właściwości, formaty, metody przetwarzania, modele barwne.	I1_W07
EW3	Podstawowe metody grafiki komputerowej: prymitywy graficzne, transformacje obiektów, obcinanie, przesłonięcia, rzutowanie	I1_W07
EW4	Zagadnienia komunikacji człowiek-komputer. Ugruntowanie standardów i estetyki tworzenia interfejsu graficznego.	I1_W07
Umiejętności Absolwent potrafi:		
EU2	Programowanie prostych efektów graficznych oraz interfejsu użytkownika z wykorzystaniem biblioteki OpenGL	I1_U12
EU3	Przetwarzanie obrazów rastrowych oraz konwersja barw z wykorzystaniem oprogramowania PhotoShop	I1_U12
EU4	Rysowanie schematów technicznych 2D z wykorzystaniem oprogramowania AutoCAD	I1_U12
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Umiejętność przekazywania informacji o metodach i zastosowaniach grafiki komputerowej.	I1_K05

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
5	4	Z	18	0	0	18	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	Systemy grafiki. Historia grafiki komputerowej. Grafika rastrowa i wektorowa. Zastosowania GK.	1
2	W	Grafika rastrowa - Prymitywy graficzne 2D: kreślenie odcinków, okręgów, wypełnianie obszarów.	1
3	W	OpenGL: historia, typy danych i reprezentacje, podstawy wyświetlania, rysowanie obiektów, przekształcenia graficzne, projektowanie interfejsu graficznego i interakcji (GLUT).	3
4	W	Transformacje w 2D, współrzędne jednorodnie. Składanie przekształceń.	1
5	W	Algorytmy obcinania i generowania przesłonięć.	1
6	W	Reprezentacje rastrowe, proste algorytmy poprawy jakości obrazu rastrowego (Photoshop). Metody interpolacji obrazów.	1
7	W	Reprezentacje wektorowe, operacje na obrazach wektorowych (AutoCAD). Interpretacja rysunku technicznego.	2
8	W	Percepcja obrazu, modele barwne.	2
13	LK	Grafika wektorowa podstawowe zasady tworzenia grafiki 2D w przykładowym środowisku: Autodesk AutoCAD	5
14	LK	Grafika rastrowa podstawowe zasady tworzenia grafiki rastrowej w przykładowym środowisku Adobe Photoshop lub GIMP	2
15	LK	Algorytmy poprawy jakości obrazu w praktyce	2
16	LK	Implementacja wybranych metod grafiki komputerowej z wykorzystaniem biblioteki OpenGL	3
17	LK	Implementacja interfejsu graficznego z wykorzystaniem biblioteki GLUT.	2
18	LK	Podstawowe transformacje obrazu (skalowanie, obrót, translacja) za pomocą mechanizmów standardowego API graficznego, implementacja prostych procedur dokonujących transformacji obrazów 2-wymiarowych.	4
19	W	Sprzęt dla potrzeb grafiki komputerowej.	1
20	W	Formaty graficzne rastrowe i wektorowe.	1
21	W	Metody rzutowania.	1
22	W	Podstawy komunikacji człowiek-komputer. Zasady budowania prostych interfejsów graficznych.	3

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Przygotowanie treści teoretycznych do kolokwium	20

2	Konsultacje przedmiotowe (w bezpośrednim kontakcie z prowadzącym)	10
3	Zaliczenia (w bezpośrednim kontakcie z prowadzącym)	4
4	Praca własna w zakresie realizowanych ćwiczeń	50

Metody dydaktyczne:

ćwiczenia laboratoryjne, konsultacje, wykład

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, kolokwium, sprawozdanie z pracy indywidualnej

Kryteria oceny:

1. Student na ocenę 5.0 bardzo dobrze opanował umiejętność implementacji prostych efektów graficznych skorelowanych z interfejsem użytkownika z wykorzystaniem biblioteki OpenGL.
2. Student na ocenę 5.0 potrafi zastosować metody przetwarzania obrazów rastrowych i konwersji barw z wykorzystaniem oprogramowania PhotoShop oraz potrafi przewidzieć rezultat przetwarzania obrazów przy zastosowaniu określonej sekwencji metod.
3. Student na ocenę 5.0 potrafi w sposób precyzyjny odtworzyć rysunek techniczny 2D na podstawie schematu z wykorzystaniem oprogramowania AutoCAD, uwzględniając zasady wymiarowania, z wykorzystaniem skrótów klawiaturowych, oraz przy zastosowaniu zaawansowanych technik programowych.
4. Student na ocenę 5.0 potrafi scharakteryzować i zaimplementować wybrane metody reprezentacji obrazów i modeli barwnych.
5. Student na ocenę 5.0 potrafi biegle stosować podstawowe metody grafiki komputerowej.
6. Student na ocenę 5.0 ma ugruntowaną wiedzę w zakresie komunikacji człowiek-komputer, oraz zna zasady praktycznego projektowania interfejsu graficznego zgodnego ze standardami i regułami estetyki.
7. Student na ocenę 5.0 potrafi przekazywać informacje o metodach i zastosowaniach grafiki komputerowej w sposób powszechnie zrozumiały.

Literatura:

obowiązkowa:

1. D. Foley, Andries van Dam, Steven K. Feiner, John F. Hughes, Richard L. Phillips, Wprowadzenie do grafiki komputerowej, Warszawa, 2002, WNT
2. Piotr Andrzejewski, Jakub Kurzak, Wprowadzenie do OpenGL. Programowanie zastosowań graficznych, Warszawa, 2000, Kwantum
3. Angel, Interactive Computer Graphics, 2009, Addison-Wesley
4. Witold Malina, Jakub Smiatacz, Metody cyfrowego przetwarzania obrazów, Warszawa, 2005, Akademicka Oficyna wydawnicza EXIT
5. Tricia Austin, Richard Doust, Projektowanie dla nowych mediów, PWN.
6. Steve Krug, Nie każ mi myśleć. O życiowym podejściu do funkcjonalności stron internetowych, Helion.

zalecana/fakultatywna:

1. Przemysław Kiciak, OpenGL i GLSL, 2019, PWN
2. Michał Jankowski, Elementy grafiki komputerowej, WNT.

3. Damian Karwowski, Zrozumieć Kompresję Obrazu - Podstawy Techniki Kodowania stratnego oraz Bezstratnego Obrazów.

Inżynieria oprogramowania nazwa przedmiotu
Software engineering nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Znajomość dowolnego języka programowania wysokiego poziomu w zakresie pozwalającym na implementację prostej aplikacji
2. Elementy algorytmów i struktur danych oraz podstawy baz danych

Cele przedmiotu:

1. Nabycie umiejętności dekompozycji rzeczywistości będącej podstawą w budowaniu kolejnych modeli inżynierii oprogramowania: od modelu opisowego, przez model wymagań i specyfikacji do numerycznego. Umiejętność identyfikacji procesów analizy i syntezy w procesach wytwórczych oprogramowania.
2. Poznanie podstawowych modeli wytwarzania oprogramowania (cykli życia): kaskadowych, cyklicznych i ewolucyjnych. Poznanie poszczególnych faz procesu wytwórczego oraz tworzonych w efekcie ich realizacji artefaktów. Umiejętność adaptacji poszczególnych faz w procesach ciężkich i lekkich.
3. Poznanie metodyk identyfikacji i specyfikacji wymagań oraz procesów zarządzania nimi. Zdobywanie umiejętności modelowania przy wykorzystaniu UML2 oraz BPMN2 w modelowaniu procesów biznesowych. Zapoznanie się z metodami estymacji kosztów wytwarzania

oprogramowania.

4. Zaznajomienie się zasadami projektowania, implementacji ,testowania i wdrożenia oprogramowania. Zdobywanie umiejętności identyfikacji metodyk formalnych i agilnych w procesach twórczych oprogramowania. Nabycie zdolności i umiejętności dokumentowania kolejnych kroków i faz procesu wytwarzania oprogramowania.
5. Zdobywanie doświadczeń w wykorzystywaniu współczesnych narzędzi automatyzacji procesów wytwarzania oprogramowania. Poznanie elementów zarządzania procesem twórczym, zarządzanie ryzykiem; zarządzanie wersjami; wprowadzenie do zarządzania projektami informatycznymi.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW5	Zdobywa wiedzę dotyczącą zagadnień dekompozycji rzeczywistości. Zna sposoby prowadzenia analizy zachodzących procesów oraz rozumie techniki ich modelowania. Rozumie na czym polega wykorzystywanie abstrakcji przy przejściu od rzeczywistości do jej cyfrowego modelu poprzez analizę modelu wymagań i syntezę do modelu systemu (I1_W01,I1_W04,I1W_05)	I1_W01
EW5	Zdobywa wiedzę dotyczącą zagadnień dekompozycji rzeczywistości. Zna sposoby prowadzenia analizy zachodzących procesów oraz rozumie techniki ich modelowania. Rozumie na czym polega wykorzystywanie abstrakcji przy przejściu od rzeczywistości do jej cyfrowego modelu poprzez analizę modelu wymagań i syntezę do modelu systemu (I1_W01,I1_W04,I1W_05)	I1_W04
EW5	Zdobywa wiedzę dotyczącą zagadnień dekompozycji rzeczywistości. Zna sposoby prowadzenia analizy zachodzących procesów oraz rozumie techniki ich modelowania. Rozumie na czym polega wykorzystywanie abstrakcji przy przejściu od rzeczywistości do jej cyfrowego modelu poprzez analizę modelu wymagań i syntezę do modelu systemu (I1_W01,I1_W04,I1W_05)	I1_W05
EW6	Zdobywa wiedzę na temat procesu twórczego oprogramowania i realizacji poszczególnych faz jego cyklu życia (I1_W09). Ma wiedzę w zakresie projektowania i implementacji prostych i złożonych systemów informatycznych przy wykorzystaniu współczesnych technik programistycznych i języków oprogramowania (I1_W10,I1_W12)	I1_W09
EW6	Zdobywa wiedzę na temat procesu twórczego oprogramowania i realizacji poszczególnych faz jego cyklu życia (I1_W09). Ma wiedzę w zakresie projektowania i implementacji prostych i złożonych systemów informatycznych przy wykorzystaniu współczesnych technik programistycznych i języków oprogramowania (I1_W10,I1_W12)	I1_W10

EW6	Zdobywa wiedzę na temat procesu wytwórczego oprogramowania i realizacji poszczególnych faz jego cyklu życia (I1_W09). Ma wiedzę w zakresie projektowania i implementacji prostych i złożonych systemów informatycznych przy wykorzystaniu współczesnych technik programistycznych i języków oprogramowania (I1_W10,I1_W12)	I1_W12
EW7	Zdobywa wiedzę w zakresie syntezy modeli numerycznych w notacji obiektowej i strukturalnej (I1_W06); posiada wiedzę pozwalającą na właściwą ocenę jakości tworzonego oprogramowania (I1_W13)	I1_W06
EW7	Zdobywa wiedzę w zakresie syntezy modeli numerycznych w notacji obiektowej i strukturalnej (I1_W06); posiada wiedzę pozwalającą na właściwą ocenę jakości tworzonego oprogramowania (I1_W13)	I1_W13
EW8	Zdobywa podstawową wiedzę w zakresie organizacji i prowadzenia procesu wytwórczego systemów informatycznych (I1_W14) z wykorzystaniem narzędzi interpretowania i modelowania procesów biznesowych. Zapoznaje się z narzędziami wspierającymi automatyzację procesu wytwórczego oprogramowania (I1_W08). Zapoznaje się z podstawowymi zagadnieniami prowadzenia projektu informatycznego. Zdobywa wiedzę pozwalającą na wybór właściwej metodyki realizacji systemu informatycznego.	I1_W08
EW8	Zdobywa podstawową wiedzę w zakresie organizacji i prowadzenia procesu wytwórczego systemów informatycznych (I1_W14) z wykorzystaniem narzędzi interpretowania i modelowania procesów biznesowych. Zapoznaje się z narzędziami wspierającymi automatyzację procesu wytwórczego oprogramowania (I1_W08). Zapoznaje się z podstawowymi zagadnieniami prowadzenia projektu informatycznego. Zdobywa wiedzę pozwalającą na wybór właściwej metodyki realizacji systemu informatycznego.	I1_W14
Umiejętności Absolwent potrafi:		
EU5	Nabywa umiejętności dekompozycji rzeczywistości w procesie tworzenia modeli opisowych z uwzględnieniem ich aspektów społecznych i prawnych (I1_U06). Potrafi przygotować odpowiednią specyfikację wymagań (I1_U22) i potrafi estymować koszty i pracochłonność wytwarzania oprogramowania (I1_U19).	I1_U06b
EU5	Nabywa umiejętności dekompozycji rzeczywistości w procesie tworzenia modeli opisowych z uwzględnieniem ich aspektów społecznych i prawnych (I1_U06). Potrafi przygotować odpowiednią specyfikację wymagań (I1_U22) i potrafi estymować koszty i pracochłonność wytwarzania oprogramowania (I1_U19).	I1_U19
EU5	Nabywa umiejętności dekompozycji rzeczywistości w procesie tworzenia modeli opisowych z uwzględnieniem ich aspektów społecznych i prawnych (I1_U06). Potrafi przygotować odpowiednią specyfikację wymagań (I1_U22) i potrafi estymować koszty i pracochłonność wytwarzania oprogramowania (I1_U19).	I1_U22
EU6	Zdobywa umiejętności transformacji specyfikowanego modelu wymagań na odpowiedni model obiektowy lub strukturalny (I1_U08, I1_U23) z wykorzystaniem prostych systemów bazodanowych (I1_U13)	I1_U08

EU6	Zdobywa umiejętności transformacji specyfikowanego modelu wymagań na odpowiedni model obiektowy lub strukturalny (I1_U08, I1_U23) z wykorzystaniem prostych systemów bazodanowych (I1_U13)	I1_U13
EU6	Zdobywa umiejętności transformacji specyfikowanego modelu wymagań na odpowiedni model obiektowy lub strukturalny (I1_U08, I1_U23) z wykorzystaniem prostych systemów bazodanowych (I1_U13)	I1_U23
EU7	Potrafi w procesie projektowania systemu określić jego czaso-i kosztochłonność (I1_U09). Nabywa umiejętności wykorzystania odpowiednich technologii programistycznych w transformacji modelu numerycznego (diagramy) na odpowiedni model cyfrowy (I1_U07)	I1_U07b
EU7	Potrafi w procesie projektowania systemu określić jego czaso-i kosztochłonność (I1_U09). Nabywa umiejętności wykorzystania odpowiednich technologii programistycznych w transformacji modelu numerycznego (diagramy) na odpowiedni model cyfrowy (I1_U07)	I1_U09
EU8	Umie przeprowadzić weryfikację zaprojektowanego systemu cyfrowego poprzez kompleksowe jego testowanie (I1_U14) Dla wdrożenia systemu potrafi ocenić stworzone narzędzie i określić jego przydatność jak również efektywność zrealizowanej funkcjonalności (I1_U20). Gwałtowny rozwój technologii informatycznych, wymuszony rozwojem sprzętu, budzi świadomość konieczności podążania za rozwijającymi się technologiami informatycznymi(I1_U23)	I1_U14
EU8	Umie przeprowadzić weryfikację zaprojektowanego systemu cyfrowego poprzez kompleksowe jego testowanie (I1_U14) Dla wdrożenia systemu potrafi ocenić stworzone narzędzie i określić jego przydatność jak również efektywność zrealizowanej funkcjonalności (I1_U20). Gwałtowny rozwój technologii informatycznych, wymuszony rozwojem sprzętu, budzi świadomość konieczności podążania za rozwijającymi się technologiami informatycznymi(I1_U23)	I1_U20
EU8	Umie przeprowadzić weryfikację zaprojektowanego systemu cyfrowego poprzez kompleksowe jego testowanie (I1_U14) Dla wdrożenia systemu potrafi ocenić stworzone narzędzie i określić jego przydatność jak również efektywność zrealizowanej funkcjonalności (I1_U20). Gwałtowny rozwój technologii informatycznych, wymuszony rozwojem sprzętu, budzi świadomość konieczności podążania za rozwijającymi się technologiami informatycznymi(I1_U23)	I1_U23
Kompetencje społeczne Absolwent jest gotów do:		
EK6	Uzyskanie zdolności komunikacji interpersonalnej - słuchania i wyrażania, pozwalające na właściwe interpretowanie rzeczywistości do modelu wymagań (I1_K02). Praca przy projekcie systemu wymaga pracy zespołowej, stąd nabywa się zdolności pracy w grupie (I1_K04). Zarówno w zespole jak i w grupie każdy musi się kierować zasadami profesjonalizmu i etyki (I1_K03)	I1_K02

EK6	Uzyskanie zdolności komunikacji interpersonalnej - słuchania i wyrażania, pozwalające na właściwe interpretowanie rzeczywistości do modelu wymagań (I1_K02 .Praca przy projekcie systemu wymaga pracy zespołowej, stąd nabywa się zdolności pracy w grupie (I1_K04). Zarówno w zespole jak i w grupie każdy musi się kierować zasadami profesjonalizmu i etyki (I1_K03)	I1_K03
EK6	Uzyskanie zdolności komunikacji interpersonalnej - słuchania i wyrażania, pozwalające na właściwe interpretowanie rzeczywistości do modelu wymagań (I1_K02 .Praca przy projekcie systemu wymaga pracy zespołowej, stąd nabywa się zdolności pracy w grupie (I1_K04). Zarówno w zespole jak i w grupie każdy musi się kierować zasadami profesjonalizmu i etyki (I1_K03)	I1_K04
EK7	Potrafi analizować procesy nie tylko techniczne ale również i społeczne. Umie je analizować i prowadzić ich modelowanie, nie tracąc kontaktu z rzeczywistością. Poznaje przyzwyczajenia i nawyki użytkowników systemów informatycznych. Zdobywa świadomość roli społecznej Inżyniera Oprogramowania (I1_K07, I1_K06, I1_K05)	I1_K04
EK7	Potrafi analizować procesy nie tylko techniczne ale również i społeczne. Umie je analizować i prowadzić ich modelowanie, nie tracąc kontaktu z rzeczywistością. Poznaje przyzwyczajenia i nawyki użytkowników systemów informatycznych. Zdobywa świadomość roli społecznej Inżyniera Oprogramowania (I1_K07, I1_K06, I1_K05)	I1_K05
EK7	Potrafi analizować procesy nie tylko techniczne ale również i społeczne. Umie je analizować i prowadzić ich modelowanie, nie tracąc kontaktu z rzeczywistością. Poznaje przyzwyczajenia i nawyki użytkowników systemów informatycznych. Zdobywa świadomość roli społecznej Inżyniera Oprogramowania (I1_K07, I1_K06, I1_K05)	I1_K06

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
6	4	E	18	0	0	0	18	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	Wprowadzenie do przedmiotu Inżynieria Oprogramowania - tło historyczne, odniesienie do innych dyscyplin inżynierskich - ogólna charakterystyka. Procesy i metody produkcji oprogramowania. Modele cyklu życia oprogramowania: kaskadowe, iteracyjne, ewolucyjne. Porównanie cykli życia oprogramowania Problemy projektów dużej i małej skali. Wprowadzenie do Nauki o Modelach, zagadnienia właściwej transformacji wymagań.	3

3	W	Analiza od rzeczywistości przez model opisowy, specyfikacje wymagań do modelu numerycznego i następnie modelu cyfrowego rzeczywistości, czyli systemu. Rola W3 wymagań i sposoby ich specyfikacji oraz dokumentacji. Szacowanie kosztów 4 realizacji systemu. Dynamika i statyka systemu informatycznego. Języki i paradygmaty wspomagania W4 modelowania systemów informatycznych (diagramy UML) i procesów biznesowych 4 (diagramy BPMN)	6
4	W	Projektowanie oprogramowania, model strukturalny a obiektowy. Transformacja od modelu numerycznego do modelu cyfrowego, architektura aplikacji, strategie, narzędzia. Wzorce projektowe, model komponentowy. Implementacja systemu informatycznego. Dobór technologii do wymagań i projektu. Standardy.	4
5	W	Zagadnienia jakości, testowanie - sposoby testowania: white/black box testing, testy regresyjne, ciągła integracja, inspekcje, przeglądy artefaktów, metryki jakości. Procesy i praktyki zwinne "agilne" na przykładzie XP i Scrum. Dobór metodyki do wielkości projektu.	3
6	W	Wybrane zagadnienia inżynierii oprogramowania: zarządzanie konfiguracją, zarządzanie zmianami, zarządzanie ryzykiem, elementy pracy w zespole projektowym.	2
7	P	Studenci w grupach 3 - 5 osób realizują kolejne elementy przedsięwzięcia projektowego. Kolejne etapy jego realizacji stanowią poszczególne przedsięwzięcia projektowego: Wybór zadania projektowego - Rozmowa z klientem, budowa modelu opisowego i modelu wymagań.	3
8	P	Identyfikacja przypadków użycia systemu dla określonej grupy aktorów. Analiza problemu. Specyfikacja wymagań i ich weryfikacja. Wstępny projekt struktury systemu. Projekt struktury systemu. Model UML lub BPMN. Budowa ostatecznej specyfikacji systemu.	5
9	P	Implementacja systemu. Wybór technologii i narzędzi. Kodowanie systemu w wybranej technologii. Opracowanie testów systemu, jednostkowych, modułowych, funkcjonalnych.	5
10	P	Wybrane zagadnienia inżynierii oprogramowania: zarządzanie konfiguracją, zarządzanie zmianami, zarządzanie ryzykiem, elementy pracy w zespole projektowym.	5

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Konsultacje przedmiotowe (w bezpośrednim kontakcie z prowadzącym)	18
2	Egzaminy i zaliczenia w sesji (w bezpośrednim kontakcie z prowadzącym)	11
3	Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury	25
4	Opracowanie wyników	10
5	Przygotowanie raportu, projektu, prezentacji, dyskusji	20

Metody dydaktyczne:

dyskusja, praca w grupach, projekt, wykład, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, dyskusja, projekt, sprawozdanie z pracy indywidualnej, test

Kryteria oceny:

1. Na ocenę 5.0 student umie zdekomponować elementy modelowanej rzeczywistości. Zna sposoby prowadzenia analizy zachodzących procesów oraz rozumie techniki ich modelowania. Potrafi wykorzystać elementy abstrakcji przy przejściu od rzeczywistości do jej cyfrowego modelu poprzez analizę modelu wymagań i syntezę do modelu cyfrowego systemu
2. Student na ocenę 5.0 potrafi implementować wiedzę na temat procesu wytwórczego oprogramowania i realizacji poszczególnych faz jego cyklu życia. Posiada wiedzę w zakresie projektowania i implementacji prostych i złożonych systemów informatycznych przy wykorzystaniu współczesnych technik programistycznych i języków oprogramowania
3. Student na ocenę 5.0 umie przeprowadzić proces syntezy modeli numerycznych w notacji obiektowej i strukturalnej; potrafi ocenić jakości budowanego oprogramowania
4. Student na ocenę 5.0 potrafi poprowadzić właściwie organizację i prowadzenie procesu wytwórczego systemów informatycznych z wykorzystaniem narzędzi interpretowania i modelowania procesów biznesowych. Zna narzędzia wspierające automatyzację procesu wytwórczego oprogramowania Potrafi właściwie poprowadzić wytwórczy projekt informatyczny. Potrafi dobrać właściwą metodykę realizacji systemu informatycznego.
5. Student na ocenę 5.0 umie przeprowadzić dekompozycję rzeczywistości w procesie tworzenia modeli opisowych z uwzględnieniem ich aspektów społecznych i prawnych Potrafi przygotować odpowiednią specyfikację wymagań i umie estymować koszty i pracochłonność wytwarzania oprogramowania .
6. Student na ocenę 5.0 sprawnie przygotowuje transformacje specyfikowanego modelu wymagań na odpowiedni model obiektowy lub strukturalny z wykorzystaniem prostych systemów bazodanowych.
7. Na ocenę 5.0 student w procesie projektowania systemu określa jego czaso- i kosztochłonność. Potrafi wykorzystać odpowiednie technologie programistyczne w transformacji modelu numerycznego (diagramy) na odpowiedni model cyfrowy
8. Student na ocenę 5.0 potrafi przeprowadzić weryfikację zaprojektowanego systemu cyfrowego poprzez kompleksowe jego testowanie. Dla wdrożenia systemu potrafi ocenić zbudowane narzędzie i określić jego przydatność jak również efektywność zrealizowanej funkcjonalności.
9. Student na ocenę 5.0 potrafi właściwie prowadzić komunikację interpersonalną - słuchanie i wyrażanie, pozwalające na właściwe interpretowanie rzeczywistości do modelu wymagań Prowadzi prace we współpracy zespołowej. Potrafi się komunikować z grupą. Etycznie podchodzi do postawionych zadań
10. Student na ocenę 5.0 potrafi analizować procesy nie tylko techniczne ale również i społeczne. Umie je analizować i prowadzić ich modelowanie, nie tracąc kontaktu z rzeczywistością. Poznaje przyzwyczajenia i nawyki użytkowników systemów informatycznych. Posiada świadomość roli społecznej Inżyniera Oprogramowania

Literatura:

obowiązkowa:

1. Sommerville J.; Inżynieria oprogramowania - klasyka informatyki, Warszawa, 2003, WNT
2. Steve McConnell; Kod doskonały, Warszawa, 2010, Helion
3. Maciaszek L.A., Liong B.L.; Practical Software Engineering a Case Study Approach, Melbourn, 2005, Pearson Education Limited
4. Śmiałek M.; Zrozumieć UML 2.0, Gliwice, 2005, Helion

5. Socha K.; Inżynieria Oprogramowania, Warszawa, 2010, WNT
6. Len Bass, Paul Clements, Rick Kazman,; Architektura oprogramowania w praktyce, warszawa, 2011, Helion

zalecana/fakultatywna:

1. Brooks F.P.; Mityczny osobomiesiąc, Warszawa, 2000, WNT
2. Hunt A., Thomas D.; Pragmatyczny programista - od czeladnika do mistrza, Warszawa, 2002, WNT

Język angielski nazwa przedmiotu
English Language nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty ogólne kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Pozytywne zaliczenie poprzedniego semestru potwierdzone wpisem w eHMS.

Cele przedmiotu:

1. Rozwijanie u studentów kompetencji komunikacyjnych i językowych oraz umiejętności udziału w dyskusji na tematy techniczne związane z informatyką.
2. Rozwijanie u studentów umiejętności czytania i słuchania ze zrozumieniem tekstów technicznych związanych z informatyką i dyscyplinami pokrewnymi.
3. Przygotowanie studenta do samodzielnej pracy z tekstami technicznymi związanymi z jego specjalnością.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
BRAK DANYCH	BRAK DANYCH	BRAK DANYCH
Umiejętności Absolwent potrafi:		
EU1	posługiwać się językiem angielskim na poziomie B2 w stopniu pozwalającym na porozumienie się, przeczytanie ze zrozumieniem tekstów i opisów programowych oraz przedstawienie prezentacji problemu z zakresu własnego kierunku studiów.	I1_U04
EU2	pozyskiwać informacje z różnorodnych źródeł w języku angielskim, integrować je, dokonywać ich interpretacji oraz wyciągać wnioski i formułować opinie	I1_U01b
Kompetencje społeczne Absolwent jest gotów do:		
EK1	przekazywania informacji o osiągnięciach informatyki i wielu aspektach zawodu informatyka w sposób powszechnie zrozumiały w języku angielskim.	I1_K05

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
2	2	Z	0	18	0	0	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	C	Zagadnienia leksykalne związane z wybranym kierunkiem studiów obejmujące mikrocipy, tranzystory, architekturę systemów komputerowych; systemy operacyjne oraz interfejsy.	9

2	C	Rozwijanie umiejętności udziału w dyskusji na tematy techniczne, argumentowania, definiowania, - uzyskiwania i udzielania informacji, opisywania procesów i zjawisk, formułowania hipotez.	5
3	C	Rozwijanie umiejętności czytania i słuchania ze zrozumieniem tekstów technicznych w obrębie tematyki związanej z kierunkiem studiów.	4

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Aktywności do opracowane samodzielnie w kursie e-learningowym na platformie Delta zatwierdzonym do użytku na kierunku Informatyka (zamiast zajęć z nauczycielem - do 30% zajęć w semestrze)	8
2	Konsultacje przedmiotowe i zaliczenie w bezpośrednim kontakcie z prowadzącym	13
3	Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury	11

Metody dydaktyczne:

ćwiczenia, dyskusja, gra dydaktyczna, konsultacje, praca w grupach, praca z tekstem/materiałem audio i wideo, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, referat, test, praca własna-zadanie, quiz, test, rysunek

Kryteria oceny:

1. Student na ocenę 5.0 bardzo dobrze zna wszystkie zagadnienia leksykalne i z łatwością używa ich do przekazywania informacji o osiągnięciach informatyki i wielu aspektach zawodu informatyka w sposób powszechnie zrozumiały w języku angielskim.
2. W zakresie umiejętności związanych z rozumieniem tekstu czytanego student na ocenę 5.0: potrafi z łatwością wyodrębnić myśl przewodnią całego komunikatu i poszczególnych jego części, jest w stanie wyodrębnić żadaną informację, z łatwością czyta ze zrozumieniem teksty specjalistyczne ze swojej dziedziny.
3. W zakresie umiejętności mówienia student na ocenę 5.0 potrafi płynnie posługiwać się językiem angielskim na poziomie B2 w stopniu pozwalającym na porozumienie się, przeczytanie ze zrozumieniem tekstów specjalistycznych oraz przedstawienie prezentacji problemu z zakresu własnego kierunku studiów.

Literatura:

obowiązkowa:

1. materiały autorskie przygotowane przez uczących przedmiot
2. kurs na platformie e-learningowej Delta zatwierdzony do użytku na kierunku Informatyka

zalecana/fakultatywna:

1. branżowe platformy internetowe dla informatyków, np. zdnet.com, gigaom.com
2. M. Błaszczak "English 4 IT", Gliwice, 2017, Helion

Język angielski nazwa przedmiotu
English Language nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty ogólne kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Pozytywne zaliczenie poprzedniego semestru potwierdzone wpisem w eHMS.

Cele przedmiotu:

1. Rozwijanie u studentów kompetencji komunikacyjnych i językowych oraz umiejętności udziału w dyskusji na tematy techniczne związane z informatyką.
2. Rozwijanie u studentów umiejętności czytania i słuchania ze zrozumieniem tekstów technicznych związanych z informatyką i dyscyplinami pokrewnymi.
3. Przygotowanie studenta do samodzielnej pracy z tekstami technicznymi w j.angielskim związanymi z jego specjalnością.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
	Wiedza Absolwent zna i rozumie:	

BRAK DANYCH	BRAK DANYCH	BRAK DANYCH
Umiejętności Absolwent potrafi:		
EU1	posługiwać się językiem angielskim na poziomie B2 w stopniu pozwalającym na porozumienie się, przeczytanie ze zrozumieniem tekstów i opisów programowych oraz przedstawienie prezentacji problemu z zakresu własnego kierunku studiów.	I1_U04
EU2	pozyskiwać informacje z różnorodnych źródeł w języku angielskim, integrować je, dokonywać ich interpretacji oraz wyciągać wnioski i formułować opinie	I1_U01b
Kompetencje społeczne Absolwent jest gotów do:		
EK1	przekazywania informacji o osiągnięciach informatyki i wielu aspektach zawodu informatyka w sposób powszechnie zrozumiały w języku angielskim.	I1_K05

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
3	2	Z	0	18	0	0	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	C	Zagadnienia leksykalne związane z wybranym kierunkiem studiów obejmujące technologie sieciowe i internetowe, cloud computing i internet rzeczy, bezpieczeństwo i ochronę danych, przestępstwa komputerowe, języki programowania; inżynierię programowania, technologie mobilne.	6
2	C	Rozwijanie umiejętności udziału w dyskusji na tematy techniczne, argumentowania, definiowania, - uzyskiwania i udzielania informacji, opisywania procesów i zjawisk, formułowania hipotez.	6
3	C	Rozwijanie umiejętności czytania i słuchania ze zrozumieniem tekstów technicznych w obrębie tematyki związanej z kierunkiem studiów.	6

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Aktywności do opracowane samodzielnie w kursie e-learningowym na platformie Delta zatwierdzonym do użytku na kierunku Informatyka (zamiast zajęć z nauczycielem - do 30% zajęć w semestrze)	5
2	Przygotowanie raportu, projektu, prezentacji, dyskusji	9
3	Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury	6
4	Konsultacje przedmiotowe i zaliczenie w bezpośrednim kontakcie z prowadzącym	12

Metody dydaktyczne:

ćwiczenia, dyskusja, gra dydaktyczna, konsultacje, praca w grupach, praca z tekstem/materiałem audio i wideo, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, referat, test, praca własna-zadanie, quiz, test, rysunek

Kryteria oceny:

1. Student na ocenę 5.0 bardzo dobrze zna wszystkie zagadnienia leksykalne i z łatwością używa ich do przekazywania informacji o osiągnięciach informatyki i wielu aspektach zawodu informatyka w sposób powszechnie zrozumiały w języku angielskim.
2. W zakresie umiejętności związanych z rozumieniem tekstu czytanego student na ocenę 5.0: potrafi z łatwością wyodrębnić myśl przewodnią całego komunikatu i poszczególnych jego części, jest w stanie wyodrębnić żadaną informację, z łatwością czyta ze zrozumieniem teksty specjalistyczne ze swojej dziedziny.
3. W zakresie umiejętności mówienia student na ocenę 5.0 potrafi płynnie posługiwać się językiem angielskim na poziomie B2 w stopniu pozwalającym na porozumienie się, przeczytanie ze zrozumieniem tekstów specjalistycznych oraz przedstawienie prezentacji problemu z zakresu własnego kierunku studiów.

Literatura:

obowiązkowa:

1. materiały autorskie przygotowane przez uczących przedmiot
2. kurs na platformie e-learningowej Delta zatwierdzony do użytku na kierunku Informatyka

zalecana/fakultatywna:

1. branżowe platformy internetowe dla informatyków, np. zdnet.com, gigaom.com
2. M. Błaszczuk "English 4 IT", Gliwice, 2017, Helion

Język angielski

nazwa przedmiotu

English Language nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty ogólne kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Pozytywne zaliczenie poprzedniego semestru potwierdzone wpisem w eHMS.

Cele przedmiotu:

1. Rozwijanie u studentów kompetencji komunikacyjnych i językowych oraz umiejętności udziału w dyskusji na tematy techniczne związane z informatyką.
2. Rozwijanie u studentów umiejętności czytania i słuchania ze zrozumieniem tekstów technicznych związanych z informatyką i dyscyplinami pokrewnymi.
3. Przygotowanie studenta do samodzielnej pracy z tekstami technicznymi związanymi z jego specjalnością.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
BRAK DANYCH	BRAK DANYCH	BRAK DANYCH

Umiejętności		
Absolwent potrafi:		
EU1	posługiwać się językiem angielskim na poziomie B2 w stopniu pozwalającym na porozumienie się, przeczytanie ze zrozumieniem tekstów i opisów programowych oraz przedstawienie prezentacji problemu z zakresu własnego kierunku studiów.	I1_U04
EU2	pozyskiwać informacje z różnorodnych źródeł w języku angielskim, integrować je, dokonywać ich interpretacji oraz wyciągać wnioski i formułować opinie	I1_U01b
Kompetencje społeczne		
Absolwent jest gotów do:		
EK1	przekazywania informacji o osiągnięciach informatyki i wielu aspektach zawodu informatyka w sposób powszechnie zrozumiały w języku angielskim.	I1_K05

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
4	2	Z	0	18	0	0	0	0	0
5	2	Z	0	18	0	0	0	0	0

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	C	Zagadnienia leksykalne związane z wybranym kierunkiem studiów obejmujące sieci neuronowe, machine learning, sztuczną inteligencję, big data, data mining, bazy danych, robotykę oraz technologie przyszłości.	17
2	C	Rozwijanie umiejętności udziału w dyskusji na tematy techniczne, argumentowania, definiowania, - uzyskiwania i udzielania informacji, opisywania procesów i zjawisk, formułowania hipotez.	12
3	C	Rozwijanie umiejętności czytania i słuchania ze zrozumieniem tekstów technicznych w obrębie tematyki związanej z kierunkiem studiów.	4
4	C	Prezentacje - język, zasady, metodyka przygotowania i prowadzenia prezentacji.	3

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
-----	--------------------	---------------

1	Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury	12
2	Przygotowanie raportu, projektu, prezentacji, dyskusji	18
3	Aktywności do opracowane samodzielnie w kursie e-learningowym na platformie Delta zatwierdzonym do użytku na kierunku Informatyka (zamiast zajęć z nauczycielem - do 30% zajęć w semestrze)	18
4	Egzaminy/zaliczenia w sesji (w bezpośrednim kontakcie z prowadzącym)	16

Metody dydaktyczne:

ćwiczenia, dyskusja, gra dydaktyczna, konsultacje, praca w grupach, praca z tekstem/materiałem audio i wideo, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, egzamin pisemny, egzamin ustny, referat, test, praca własna-zadanie, quiz, test, rysunek

Kryteria oceny:

1. Student na ocenę 5.0 bardzo dobrze zna wszystkie zagadnienia leksykalne i z łatwością używa ich do przekazywania informacji o osiągnięciach informatyki i wielu aspektach zawodu informatyka w sposób powszechnie zrozumiały w języku angielskim.
2. W zakresie umiejętności związanych z rozumieniem tekstu czytanego student na ocenę 5.0: potrafi z łatwością wyodrębnić myśl przewodnią całego komunikatu i poszczególnych jego części, jest w stanie wyodrębnić żadaną informację, z łatwością czyta ze zrozumieniem teksty specjalistyczne ze swojej dziedziny.
3. W zakresie umiejętności mówienia student na ocenę 5.0 potrafi płynnie posługiwać się językiem angielskim na poziomie B2 w stopniu pozwalającym na porozumienie się, przeczytanie ze zrozumieniem tekstów specjalistycznych oraz przedstawienie prezentacji problemu z zakresu własnego kierunku studiów.

Literatura:

obowiązkowa:

1. materiały autorskie przygotowane przez uczących przedmiot
2. kurs na platformie e-learningowej Delta zatwierdzony do użytku na kierunku Informatyka

zalecana/fakultatywna:

1. branżowe platformy internetowe dla informatyków, np. zdnet.com, gigaom.com
2. M. Błaszczuk "English 4 IT", Gliwice, 2017, Helion

Język angielski

nazwa przedmiotu

English Language nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty ogólne kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Świadectwo maturalne z języka obcego na poziomie podstawowym lub rozszerzonym.

Cele przedmiotu:

1. Rozwijanie u studentów umiejętności udziału w dyskusji na tematy techniczne.
2. Rozwijanie u studentów umiejętności czytania i słuchania ze zrozumieniem tekstów technicznych.
3. Przygotowanie studenta do samodzielnej pracy z tekstami technicznymi związanymi z jego specjalnością.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
BRAK DANYCH	BRAK DANYCH	BRAK DANYCH

Umiejętności		
Absolwent potrafi:		
EU1	posługiwać się językiem angielskim na poziomie B2 w stopniu pozwalającym na porozumienie się, przeczytanie ze zrozumieniem tekstów specjalistycznych raz przedstawienie prezentacji problemu z zakresu własnego kierunku studiów.	I1_U04
EU2	pozyskiwać informacje z różnorodnych źródeł w języku angielskim, integrować je, dokonywać ich interpretacji oraz wyciągać wnioski i formułować opinie	I1_U01b
Kompetencje społeczne		
Absolwent jest gotów do:		
EK1	przekazywania informacji o osiągnięciach informatyki i wielu aspektach zawodu informatyka w sposób powszechnie zrozumiały w języku angielskim.	I1_K05

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
1	2	Z	0	18	0	0	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	C	Zagadnienia leksykalne związane z nauką i techniką : podstawowe pojęcia i działania matematyczne, symbole matematyczne; zbiory oraz funkcje matematyczne; figury geometryczne; wykresy i grafy; rachunek różniczkowy, logarytmy, macierze, elementy statystyki	9
2	C	Rozwijanie umiejętności czytania i słuchania ze zrozumieniem tekstów technicznych w obrębie tematyki związanej z kierunkiem studiów.	4
3	C	Rozwijanie umiejętności udziału w dyskusji na tematy techniczne, argumentowania, definiowania, - uzyskiwania i udzielania informacji, opisywania procesów i zjawisk, formułowania hipotez.	5

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury	21

2	Konsultacje przedmiotowe i zaliczenie w bezpośrednim kontakcie z prowadzącym	3
3	Aktywności do opracowane samodzielnie w kursie e-learningowym na platformie Delta zatwierdzonym do użytku na kierunku Informatyka (zamiast zajęć z nauczycielem - do 30% zajęć w semestrze)	8

Metody dydaktyczne:

ćwiczenia, dyskusja, gra dydaktyczna, konsultacje, praca w grupach, praca z tekstem/materiałem audio i wideo, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, referat, test, praca własna-zadanie, quiz, test, rysunek

Kryteria oceny:

1. Student na ocenę 5.0 bardzo dobrze zna wszystkie zagadnienia leksykalne i z łatwością używa ich do przekazywania informacji o osiągnięciach informatyki i wielu aspektach zawodu informatyka w sposób powszechnie zrozumiały w języku angielskim.
2. W zakresie umiejętności związanych z rozumieniem tekstu czytanego student na ocenę 5.0: potrafi z łatwością wyodrębnić myśl przewodnią całego komunikatu i poszczególnych jego części, jest w stanie wyodrębnić żadaną informację, z łatwością czyta ze zrozumieniem teksty specjalistyczne ze swojej dziedziny.
3. W zakresie umiejętności mówienia student na ocenę 5.0 potrafi płynnie posługiwać się językiem angielskim na poziomie B2 w stopniu pozwalającym na porozumienie się, przeczytanie ze zrozumieniem tekstów specjalistycznych oraz przedstawienie prezentacji problemu z zakresu własnego kierunku studiów.

Literatura:

obowiązkowa:

1. materiały autorskie przygotowane przez uczących przedmiot
2. kurs na platformie e-learningowej Delta zatwierdzony do użytku na kierunku Informatyka

zalecana/fakultatywna:

1. A. Łyczko "English for Mathematics", Kraków, 2015, Politechnika Krakowska
2. A. Krukiewicz-Gacek, A. Trzaska "English for Mathematics", Kraków, 2010, AGH

Języki funkcyjne nazwa przedmiotu
Functional languages nazwa przedmiotu w języku angielskim

polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Znajomość podstaw obsługi komputera.

Cele przedmiotu:

1. Poznanie teoretycznych aspektów programowania funkcyjnego.
2. Poznanie praktycznych aspektów programowania funkcyjnego na przykładzie wybranych języków programowania.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	Student zna teoretyczne aspekty programowania funkcyjnego.	I1_W06
EW1	Student zna teoretyczne aspekty programowania funkcyjnego.	I1_W11
EW2	Student zna wybrane języki funkcyjne.	I1_W06
EW2	Student zna wybrane języki funkcyjne.	I1_W11
Umiejętności Absolwent potrafi:		

EU1	Student potrafi używać wybranych języków funkcyjnych do rozwiązywania problemów.	I1_U01b
EU1	Student potrafi używać wybranych języków funkcyjnych do rozwiązywania problemów.	I1_U02
EU1	Student potrafi używać wybranych języków funkcyjnych do rozwiązywania problemów.	I1_U03
EU1	Student potrafi używać wybranych języków funkcyjnych do rozwiązywania problemów.	I1_U04
EU1	Student potrafi używać wybranych języków funkcyjnych do rozwiązywania problemów.	I1_U05
EU1	Student potrafi używać wybranych języków funkcyjnych do rozwiązywania problemów.	I1_U08
EU1	Student potrafi używać wybranych języków funkcyjnych do rozwiązywania problemów.	I1_U18
EU1	Student potrafi używać wybranych języków funkcyjnych do rozwiązywania problemów.	I1_U20
EU1	Student potrafi używać wybranych języków funkcyjnych do rozwiązywania problemów.	I1_U22
EU1	Student potrafi używać wybranych języków funkcyjnych do rozwiązywania problemów.	I1_U23
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Student zna i docenia rolę programowania funkcyjnego we współczesnym świecie IT.	I1_K01
EK1	Student zna i docenia rolę programowania funkcyjnego we współczesnym świecie IT.	I1_K02
EK1	Student zna i docenia rolę programowania funkcyjnego we współczesnym świecie IT.	I1_K03
EK1	Student zna i docenia rolę programowania funkcyjnego we współczesnym świecie IT.	I1_K04
EK1	Student zna i docenia rolę programowania funkcyjnego we współczesnym świecie IT.	I1_K05
EK1	Student zna i docenia rolę programowania funkcyjnego we współczesnym świecie IT.	I1_K06

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
5	5	Z	18	0	0	18	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	Rachunek lambda.	1
2	W	Listy i rekurencja jako podstawowe metody używane w programowaniu funkcyjnym.	1
3	W	Podstawy teorii kategorii i jej zastosowanie do modelowania języków funkcyjnych oraz modelowania świata.	1
4	W	Scheme jak język reprezentujący rodzinę języków LISPowych. Porównanie z językiem LISP.	5
5	W	Język Erlang i platforma OTP.	5
6	W	Podstawy języka Haskell jako czysto funkcyjnego języka programowania.	5
7	LK	Listy i rekurencja jako podstawowe metody używane w programowaniu funkcyjnym.	1
8	LK	Podstawy języka Haskell jako czysto funkcyjnego języka programowania.	6
9	LK	Modelowania świata przy użyciu teorii kategorii.	1
10	LK	Scheme jak język reprezentujący rodzinę języków LISPowych. Porównanie z językiem LISP.	4
11	LK	Język Erlang i platforma OTP.	4
12	LK	Praca nad projektem, prezentacja i dyskusja wyników.	2

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Konsultacje przedmiotowe (w bezpośrednim kontakcie z prowadzącym)	10
2	Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury	20
3	Opracowanie wyników (w bezpośrednim kontakcie z prowadzącym)	16
4	Przygotowanie raportu, projektu, prezentacji, dyskusji	41
5	Zaliczenie (w bezpośrednim kontakcie z prowadzącym)	2

Metody dydaktyczne:

analiza przypadku, ćwiczenia laboratoryjne, dyskusja, konsultacje, praca w grupach, praca z tekstem/materiałem audio i wideo, projekt, symulacja, wykład, laboratorium komputerowe, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, dyskusja, kolokwium, odpowiedź ustna, prezentacja, projekt, referat, rozwiązanie zadania problemowego, sprawozdanie z pracy indywidualnej, sprawozdanie z pracy zespołowej, test, zaliczenie pisemne, zaliczenie ustne, sprawdzian wiadomości z wykładów, kolokwium z zadań, praca własna-zadanie, quiz, test, rysunek, sprawozdanie

Kryteria oceny:

1. Na ocenę 5.0 Student zna teoretyczne aspekty programowania funkcyjnego powyżej 90% materiału omawianego na wykładzie.

2. Na ocenę 5.0 Student zna wybrane języki funkcyjne powyżej 90% materiału omawianego na wykładzie.
3. Na ocenę 5.0 Student potrafi używać wybranych języków funkcyjnych do rozwiązywania problemów powyżej 90% materiału omawianego na laboratoriach.
4. Na ocenę 5.0 student zna i docenia rolę programowania funkcyjnego we współczesnym świecie IT.

Literatura:

obowiązkowa:

1. Miran Lipova, 'Learn You a Haskell for Great Good! A Beginner's Guide', 2011, No Starch Press
2. Harold Abelson, Gerald Jay Sussman, 'Struktura i Interpretacja Programów Komputerowych', 2002, WNT
3. Brendan Fong, Bartosz Milewski, David I. Spivak, 'Programming with Categories', 2020, Skrypt MIT
4. Brendan Fong, David I Spivak, 'Seven Sketches in Compositionality: An Invitation to Applied Category Theory', 2018, Skrypt MIT
5. Fred Hebert, 'Learn You Some Erlang for Great Good! A Beginner's Guide', 2013, No Starch Press

zalecana/fakultatywna:

1. Peter Smith, 'Beginning Category Theory', Cambridge, 2022, Skrypt
2. David I. Spivak, 'Category theory for scientists', 2013, MIT

Języki i paradygmaty programowania nazwa przedmiotu
Programming languages and paradigms nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia
Forma studiów: niestacjonarne

Wymagania wstępne:

1. Brak wymagań wstępnych

Cele przedmiotu:

1. Poznanie podstawowych paradygmatów programowania, w tym paradygmatu proceduralnego oraz obiektowego. Osiągnięcia umiejętności w ocenie przydatności paradygmatów do rozwiązywania różnego typu problemów; projektowania, implementacji, testowania i debugowania prostych programów proceduralnych i obiektowych.
2. Poznanie składni języków C/C++ jako przykładów języka strukturalnego oraz obiektowego.
3. Poznanie podstaw proceduralnego podejścia do programowania - tworzenia funkcji oraz procedur, które potrafią obsłużyć dowolne typy danych. Zamykanie danych w oddzielnych plikach, udostępnianie działań nad danymi przez funkcje - elementy stylu obiektowego w języku C.
4. Poznanie podstaw obiektowego podejścia do programowania - enkapsulacja, dziedziczenie i polimorfizm.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	zagadnienia w zakresie języków i paradygmatów programowania, programowania obiektowego oraz inżynierii oprogramowania.	I1_W06
EW2	podstawowe metody, techniki i narzędzia stosowane przy rozwiązywaniu prostych zadań informatycznych z zakresu analizy złożoności obliczeniowej algorytmów oraz implementacji języków programowania.	I1_W11
Umiejętności Absolwent potrafi:		
EU1	wykorzystać nabytą wiedzę matematyczną do opisu procesów, tworzenia i analizy modeli matematycznych, analizy statystycznej wyników symulacji i pomiarów.	I1_U06b
EU2	stworzyć model obiektowy prostego systemu.	I1_U09
Kompetencje społeczne Absolwent jest gotów do:		

EK1	zrozumienia potrzeby zachowań profesjonalnych i przestrzegania zasad etyki, w tym uczciwości.	I1_K02
-----	---	--------

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
1	5	Z	18	0	0	18	0	0	BRAK DANYCH
2	6	E	18	0	0	18	9	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	Cz. I. Programowanie strukturalne. Pojęcie języka strukturalnego, struktura programu napisanego w języku C, pliki źródłowe i nagłówkowe. Struktura biblioteki CRT. Pojęcie o IDE. Pojęcie funkcji, funkcji statyczne, strumienie stdin, stdout, znaki specjalne (escape sequence), typy i formaty zmiennych, dyrektywy preprocesora.	1
2	W	Operatory, wyrażenia, instrukcje, operator przypisania. Biblioteka funkcji matematycznych. Konwersja typów, instrukcja warunkowa, operatory arytmetyczne, relacyjne, logiczne, priorytety operatorów.	1
3	W	Definicja a deklaracja funkcji, lista argumentów, argumenty formalne i faktyczne, sposoby przekazywania argumentów do funkcji, wartość zwracana przez funkcję.	1
4	W	Instrukcje pętli for, while, do-while, instrukcja goto. Pojęcie o wskaźnikach. Konwersje typów - jawne i niejawne	1
5	W	Stałe i zmienne znakowe. Tablice typu char, wchar_t, int, double. Stałe tekstowe. Tablice dwuwymiarowe. Tablice w argumentach funkcji.	2
6	W	Wskaźniki do tablic. Dynamiczne alokowanie i zwolnienie pamięci dla tablic jednowymiarowych. Poruszanie się po tablicy za pomocą wskaźnika. Funkcje CRT obsługi tablic. Zmienna enum.	1
7	W	Statyczne wydzielenie pamięci i inicjalizacja. Dynamiczne alokowanie i zwolnienie pamięci. Pojęcie o fragmentowaniu przestrzeni adresowej i sposoby kompresji heap. Kopiowanie tablic na przykrywających się obszarach pamięci oraz inne operacje nad tablicami. Funkcje CRT obsługi heap.	1
8	W	Dynamiczne alokowanie i zwolnienie pamięci dla tablic dwuwymiarowych. Inicjalizacja, pojęcie "wskaźnik do wskaźnika".	2
9	W	Otwieranie i zamykanie pliku. Zapis i odczyt sformatowany. Pliki binarne. Sterowanie wskaźnikiem pozycji pliku. Zapis i odczyt do/z pliku binarnego. Tworzenie i usunięcie katalogu. Struktura pliku z rekordami o różnej długości.	2
10	W	Struktura programu w C. Zmienne lokalne, globalne, statyczne. Klasy pamięci i zasięg deklaracji (obszar widoczności).	1

11	W	Pojęcie struktury jako łączenie danych w całość. Obiekty struktur oraz tablice struktur. Dynamiczne alokowanie i zwolnienie pamięci.	2
12	W	Wektory i macierzy. Podstawowe algorytmy algebry liniowej i osobliwości ich implementacji w języku C. Rozwijanie pętli i blokowanie rejestrów.	1
13	W	Struktury danych: kolejka, stos, lista. Rozdzielenie obsługi kontenera od obsługi danych. Wskaźniki void * oraz wskaźniki do funkcji. Tablice wskaźników do funkcji.	1
14	W	Bitowe operacji logiczne. Maski bitowe.	1
15	W	Cz. II. Programowanie obiektowe: inkapsulacja, polimorfizm i dziedziczenie. 2. Zasięg deklaracji i czas trwania obiektów. Przestrzenie nazw. Wprowadzenie do klas. Konstruktory i destruktory.	2
16	W	Wprowadzenie w dziedziczenie. Specyfikatory private, protected, public. Funkcje inline. Przypisanie obiektów. Przekazywanie obiektów do funkcji.	2
17	W	Zwracanie obiektu przez funkcje. Funkcje zaprzyjaźnione. Przeciążanie funkcji, konstruktory kopiujące, argumenty domyślne.	2
18	W	Przeciążanie operatorów.	2
19	W	Dynamiczne alokowanie pamięci. Operatory new, delete. Dziedziczenie. Specyfikatory dostępu. Wielodziedziczenie. Klasy wirtualne.	2
20	W	Polimorfizm dynamiczny. Funkcje wirtualne, abstrakcyjne, klasy abstrakcyjne.	2
21	W	Identyfikację typu na etapie wykonania (RTTI).	2
22	W	Szablony funkcji i klas.	2
23	W	Wejście - wyjście w C++. Obsługa wyjątków. Statyczne składowe klasy. Specyfikatory const, volatile. Nie Polimorficzne rzutowanie typów.	2
24	LK	Wprowadzenie danych z klawiatury i wyprowadzenie na monitor. Formatowanie, tworzenie tabeli.	1
25	LK	Tablice jednowymiarowe i dwuwymiarowe. Pętle for, while, do - while.	1
26	LK	Obsługa wierszy tekstowych: wprowadzenie danych z monitora, kopiowanie, sklewanie, sortowanie, poszukiwanie kontekstu.	2
27	LK	Tablice o dwóch indeksach przy dynamicznym alokowaniu pamięci. Zmiana rozmiaru wiersza, wymiana wierszy pomiędzy sobą, dodawanie nowego wiersza.	2
28	LK	Tablica obiektów struktur przy dynamicznym alokowaniu pamięci. Obsługa tablicy, obsługa obiektu.	1
29	LK	Wskaźniki do funkcji: program obliczający całkę oznaczoną z gładkiej funkcji jednej zmiennej rzeczywistej.	1
30	LK	Stworzenie systemu zapisu do pliku binarnego tablicy obiektów. Różne rekordy mają różną długość. Odczyt z pliku binarnego rekordów w dowolnej kolejności.	2
31	LK	Hierarchia obiektów. Obsługa tablicy obiektów Triangle, zawierających obiekt Point (wierzchołek trójkąta). Generalna zasada: kontener (tablica) nie powinna zależeć od typu danych. Technika: stosowanie wskaźników void * oraz wskaźników do funkcji.	2
32	LK	Tworzymy dla poprzedniego zadania menadżer pamięci, odpowiadający za alokowanie, zwolnienie, re-alokowanie w celu przechwytywania ucieczek pamięci.	2
33	LK	Techniki obsługi macierzy i wektorów. Rozwijanie petli dla algorytmu obliczenia iloczynu skalarnego dwóch wektorów, funkcji pomiaru czasu.	1

34	LK	Przepisanie jednego obiektu do innego. Powstające problemy i sposoby rozwiązania.	2
35	LK	Operacje bitowe. Maski bitowe.	1
36	LK	Wprowadzenie w C++. Pierwsze programy. Enkapsulacja. Pojęcia klasy. Dane i metody klasy	2
37	LK	Inicjowanie i niszczenie obiektu. Konstruktory i destruktory. Hierarchia klas, wprowadzenie w dziedziczenie.	2
38	LK	Przekazywanie obiektów do funkcji (przez wartość, przez wskaźnik, przez referencje). Funkcja zwraca obiekt. Konstruktory kopiujące. Przeciążenie operatora przypisania.	2
39	LK	Operatory new, delete. Dynamiczne Alokowanie i zwolnienie pamięci.	2
40	LK	Funkcje zaprzyjaźnione do klasy. Przeciążanie operatorów binarnych.	2
41	LK	Funkcje-szablony. Szablony klas.	2
42	LK	Funkcje wirtualne, identyfikację typu RTTI.	2
43	LK	Wyjątki C++	2
44	LK	Wejście-Wyjście C++.	2
45	P	Tworzenie projektu GDI z jednym oknem (SDI - Single Document Interface) w postaci kilku podprojektów, jeden z których obsługuje interfejs graficzny SDI i generuje exe moduł, a pozostałe moduły obsługują dane, wyjątki i t.d. w postaci dynamicznie wiązanych bibliotek (dll).	3
46	P	Obsługa binarnego wejścia-wyjścia, rysowanie w oknie graficznym punktów i linii, odzwierciedlających dane zadania.	3
47	P	Tworzenie menu użytkownika za pomocą redaktora resursów VS 2022, tworzenie dialogów. Tworzenie tablicy dynamicznej w postaci kontenera przy użyciu klasy-szablonu.	3

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury	80
2	Przygotowanie raportu, projektu, prezentacji, dyskusji	60
3	Opracowanie wyników	20
4	Konsultacje (w bezpośrednim kontakcie z prowadzącym)	30
5	Zaliczenia, egzaminy (w bezpośrednim kontakcie z prowadzącym)	5

Metody dydaktyczne:

ćwiczenia laboratoryjne, dyskusja, konsultacje, projekt, wykład, ćwiczenia tablicowe, laboratorium komputerowe, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, egzamin pisemny, egzamin ustny, kolokwium, projekt

Kryteria oceny:

1. Na ocenę 5 student potrafi napisać kod optymalny pod względem wydajności i uzasadnić jego optymalność.

2. Na ocenę 5 student potrafi zaprojektować struktury, klasy, unie oraz ich obsługę w stylu obiektowym (język C), które modelują rzeczywiste złożone obiekty, umie definiować struktury/klasy, elementami których występują obiekty innych struktur/klas. W C++ potrafi tworzyć hierarchie klas w oparciu na dziedziczenie, stosować szablony funkcji i klas oraz implementować polimorfizm dynamiczny.
3. na ocenę 5 student zna zasady proceduralnego/obektowego programowania i potrafi stworzyć złożony program w języku C/C++, obsłużyć We/Wy do/z pliku binarnego, opracować w programie osobne systemy obsługi błędów i komunikatów, oddzielić obsługę kontenera od obsługi danych, umie tworzyć szablony klas i funkcji, stosować wszystkie narzędzia polimorfizmu dynamicznego.
4. Na ocenę 5 student zna składnię C/C++ w stopniu pozwalającym na tworzenie zwartych, szybkich i poprawnie działających programów, potrafi łączyć dane w struktury/klasy, unie, dynamicznie alokować i zwalniać pamięć dla tablic obiektów dowolnego typu, umie rozdzielić funkcjonalność kontenera od obsługi danych, umie stosować elementy STL.
5. Na ocenę 4.5 student potrafi zastosować kod używający dynamicznych struktur danych.
6. Na ocenę 4.5 student potrafi zrobić projekt współdziałania wszystkich struktur, klas i unii współdziałających w programie, potrafi zamykać dane i funkcje lokalne w pliku i udostępniać działania nad danymi poprzez tworzenie funkcji interfejsowych w języku C oraz stosować zasady enkapsulacji w C++.
7. Na ocenę 4.5 student zna zasady proceduralnego/obektowego programowania i potrafi stworzyć złożony program w języku C/C++, obsłużyć We/Wy do/z pliku binarnego, opracować w programie osobne systemy obsługi błędów i komunikatów, oddzielić obsługę kontenera od obsługi danych.
8. Na ocenę 4.5 student zna wszystkie elementy składni C/C++, potrafi zaprojektować program używając styl obiektowy, umie łączyć dane w struktury/klasy/unii oraz panuje arytmetyka wskaźników.
9. Na ocenę 4.0 student potrafi zrobić analizę pozwalającą ustalić wymagania jakie stawiamy programowi.
10. Na ocenę 4.0 student potrafi tworzyć własne biblioteki funkcji.
11. Na ocenę 4.0 student zna zasady proceduralnego/obektowego programowania i potrafi stworzyć złożony program w języku C/C++, obsłużyć We/Wy do/z pliku binarnego, opracować w programie osobne systemy obsługi błędów i komunikatów.
12. Na ocenę 4.0 student zna jak napisać definicje struktury/klasy/unii i stworzyć tablice obiektów złożonych.
13. Na ocenę 3.5 student potrafi napisać program w C/C++ realizujący przyjęte założenia.
14. Na ocenę 3.5 student umie zaprojektować strukturę, klasę, unie, enumeration dopasowane do potrzeb programu.
15. Na ocenę 3.5 student opanował składnię C/C++ w stopniu pozwalającym na zwarty zapis kodu prostych programów, potrafi analizować kod prostych programów, nie potrafi napisać programu w stylu obiektowym.
16. Na ocenę 3.5 student zna zasady proceduralnego/obektowego programowania i potrafi stworzyć złożony program w języku C/C++, składający się z wielu plików.
17. Na ocenę 3.0. Student nie zna wielu elementów składni, ale potrafi je zastąpić innymi; potrafi napisać prosty program w wersji proceduralnej/obektowej, ma kłopoty ze zrozumieniem działania programów w C/C++ na podstawie analizy kodu źródłowego napisanego przez innych programistów. Student rozumie działanie programu na podstawie lektury kodu źródeł.
18. Na ocenę 3.0 potrafi napisać definicje prostej struktury/klasy bez odniesienia się do potrzeb programu.
19. Na ocenę 3.0. Student zna zasady proceduralnego/obektowego programowania i potrafi stworzyć prosty program w języku C/C++.

20. Na ocenę 3.0. Student nie zna wielu elementów składni, ale potrafi je zastąpić innymi; potrafi napisać prosty program w wersji proceduralnej/obiektywnej, ma kłopoty ze zrozumieniem działania programów w C/C++ na podstawie analizy kodu źródłowego napisanego przez innych programistów.
21. Na ocenę 5.0: Opanowanie więcej niż 90% materiału.
22. Na ocenę 4.5: Opanowanie więcej niż 80% materiału.
23. Na ocenę 4.0: Opanowanie więcej niż 70% materiału.
24. Na ocenę 3.5: Opanowanie więcej niż 60% materiału.
25. Na ocenę 3.0: Opanowanie więcej niż 50% materiału.

Literatura:

obowiązkowa:

1. B.Kernighan, D.Ritchie. Język ANSI C, Warszawa, 2002, WNT
2. Herbert Schildt. The Complete Reference C++. 2020, Tata McGraw-Hill.
3. B. Stroustrup, Piwko Łukasz. Język C++. Kompendium wiedzy, Warszawa, 2012, Helion
4. MSDN - Microsoft Developer Network, , 2023, Dokumentacja Microsoft

zalecana/fakultatywna:

1. J. Richter, C. Nasarre. Windows via C/C++, Microsoft Press, 2008, APN PROMISE
2. S.Prata. Język C. Szkoła Programowania, Gliwice, 2006, Helion

Komunikacja w biznesie nazwa przedmiotu
Business communication nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty ogólne kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. brak wymagań wstępnych

Cele przedmiotu:

1. Zapoznanie studentów z podstawowymi zasadami komunikacji interpersonalnej w warunkach współczesnego społeczeństwa opartego na pozyskiwaniu informacji i przepływie wiedzy.
2. Wykształcenie umiejętności analizy różnorodnych źródeł informacji i komunikowania się w różnych sytuacjach zawodowych i biznesowych.
3. Wykształcenie postawy otwartości na różne formy i techniki komunikacji.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	podstawowe pojęcia i zasady komunikacji społecznej wykorzystywane w różnych kontekstach zawodowych.	I1_W14
Umiejętności Absolwent potrafi:		
EU1	zaplanować i przeprowadzić z wykorzystaniem poznanych metod i technik skuteczną komunikację w miejscu pracy oraz z otoczeniem biznesowym.	I1_U05
Kompetencje społeczne Absolwent jest gotów do:		
EK1	efektywnego komunikowania się w różnych kontekstach zawodowych i społecznych.	I1_K03

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
3	1	Z	0	9	0	0	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	C	Współczesne teorie komunikacji interpersonalnej, ich zasady, reguły i sposoby aplikacji. Psychologia komunikowania się.	2
2	C	Cele, cechy i modele skutecznej komunikacji. Rola nadawcy i odbiorcy w procesie komunikacji. Ćwiczenie technik efektywnej komunikacji w małej grupie. Opanowanie umiejętności aktywnego słuchania i analizy treści w procesach komunikacji interpersonalnej	2
3	C	Komunikacja międzykulturowa, komunikacja w warunkach konfliktu i różnic społecznych.	1
4	C	Zastosowanie technik autoprezentacji zawodowej w typowych sytuacjach komunikacyjnych - symulacje indywidualne zespołowe.	2
5	C	Techniki autoprezentacji społecznej i zawodowej. Budowanie wizerunku w sieciach społecznych. Podstawowe zasady Public Relation w działalności biznesowej	2

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Konsultacje przedmiotowe w bezpośrednim kontakcie z prowadzącym	2
2	Przygotowanie do kolokwium i rozwiązania zadania problemowego	14

Metody dydaktyczne:

analiza przypadku, ćwiczenia, praca w grupach, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, kolokwium, rozwiązanie zadania problemowego

Kryteria oceny:

1. Student na ocenę 5.0 zna pojęcia i metody psychologii komunikacji oraz zasady efektywnego procesu komunikacji w środowisku pracy.
2. Student na ocenę 5.0 potrafi zaplanować i przeprowadzić efektywną komunikację w konkretnych sytuacjach biznesowych
3. Student na ocenę 5.0 potrafi efektywnie komunikować się i budować poprawne relacje z innymi podmiotami w środowisku zawodowym i pozazawodowym.

Literatura:

obowiązkowa:

1. Z. Necki, Komunikacja międzyludzka, Kraków, 2000, Antykwa
2. M. Leary, Wywieranie wrażenia na innych, Gdańsk, 2012, GWP
3. J. Stankiewicz, Komunikowanie się w organizacji, Wrocław, 1999, Rebis

zalecana/fakultatywna:

1. M. McKay i in., Sztuka skutecznego porozumiewania się, Gdańsk, 2001, GWP

Logika dla inżynierów nazwa przedmiotu
Logic for Engineers nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

Nie określono wymagań wstępnych

Cele przedmiotu:

1. Zapoznanie studentów z pojęciami, metodami wnioskowania i dowodzenia w klasycznym rachunku zdań.
2. Zapoznanie studentów z pojęciami, metodami wnioskowania i dowodzenia w rachunku predykatów.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		

EW1	Student zna i rozumie podstawowe pojęcia rachunku zdań, metody sprawdzania spełnialności i prawdziwości formuł rachunku zdań.	I1_W01
EW2	Student zna i rozumie podstawowe pojęcia, składnie i semantykę rachunku predykatów, metody sprawdzania spełnialności i prawdziwości formuł rachunku predykatów.	I1_W01
Umiejętności Absolwent potrafi:		
EU1	Student potrafi przekształcać formuły logiczne rachunku zdań, badać ich spełnialność oraz dowodzić ich prawdziwość.	I1_U06b
EU2	Student potrafi przekształcać formuły logiczne rachunku predykatów, badać ich spełnialność oraz dowodzić ich prawdziwość.	I1_U06b
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Student rozumie wagę wiedzy i umiejętności z zakresu logiki matematycznej w zastosowaniach praktycznych.	I1_K06

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
1	3	E	9	9	0	0	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	Klasyczny rachunek zdań, składnia rachunku zdań, tautologie rachunku zdań. Koniunkcyjna i alternatywna postać normalna formuł. Systemy dowodzenia i wyprowadzalność formuł rachunku zdań.	3
2	W	Postać klauzulowa formuł rachunku zdań. Metoda rezolucji w rachunku zdań.	1
3	W	Rachunek predykatów, składnia i semantyka rachunku predykatów. Uzgadnianie formuł rachunku predykatów.	2
4	W	Systemy dowodzenia oraz metoda rezolucji w rachunku predykatów.	2
5	W	Rozstrzygalność rachunku zdań i rachunku predykatów. Zastosowania.	1
6	C	Klasyczny rachunek zdań, spełnialność, konsekwencje logiczne, tautologie rachunku zdań.	3

7	C	Systemy dowodzenia i wyprowadzanie formuł rachunku zdań. Postać klauzulowa i metoda rezolucji w rachunku zdań.	2
8	C	Rachunek predykatów. Interpretacje. Uzgadnianie formuł rachunku predykatów. Systemy dowodzenia w rachunku predykatów. Sprowadzanie formuł rachunku predykatów do postaci klauzulowej i badanie spełnialności metoda rezolucji.	4

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury, rozwiązywanie przykładowych zadań (praca samodzielna bez udziału nauczyciela akademickiego).	52
2	Konsultacje przedmiotowe i egzamin w sesji (w bezpośrednim kontakcie z nauczycielem akademickim).	20

Metody dydaktyczne:

ćwiczenia, ćwiczenia rachunkowe, dyskusja, konsultacje, wykład, ćwiczenia tablicowe, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

egzamin pisemny

Kryteria oceny:

1. EW1: Weryfikacja osiągnięcia efektów uczenia odbywa się za pomocą egzaminu pisemnego z całości materiału. Ocena wystawiana jest zgodnie ze skalą: 5.0 wynik od 90% w górę, 4.5 wynik od 80% do 90%*, 4.0 wynik od 70% do 80%*, 3.5 wynik od 60% do 70%*, 3.0 wynik od 50% do 60%*, 2.0 wynik poniżej 50%. *przedziały prawostronnie otwarte
2. EW2: Weryfikacja osiągnięcia efektów uczenia odbywa się za pomocą egzaminu pisemnego z całości materiału. Ocena wystawiana jest zgodnie ze skalą: 5.0 wynik od 90% w górę, 4.5 wynik od 80% do 90%*, 4.0 wynik od 70% do 80%*, 3.5 wynik od 60% do 70%*, 3.0 wynik od 50% do 60%*, 2.0 wynik poniżej 50%. *przedziały prawostronnie otwarte
3. EU1: Weryfikacja osiągnięcia efektów uczenia odbywa się za pomocą egzaminu pisemnego z całości materiału. Ocena wystawiana jest zgodnie ze skalą: 5.0 wynik od 90% w górę, 4.5 wynik od 80% do 90%*, 4.0 wynik od 70% do 80%*, 3.5 wynik od 60% do 70%*, 3.0 wynik od 50% do 60%*, 2.0 wynik poniżej 50%. *przedziały prawostronnie otwarte
4. EU2: Weryfikacja osiągnięcia efektów uczenia odbywa się za pomocą egzaminu pisemnego z całości materiału. Ocena wystawiana jest zgodnie ze skalą: 5.0 wynik od 90% w górę, 4.5 wynik od 80% do 90%*, 4.0 wynik od 70% do 80%*, 3.5 wynik od 60% do 70%*, 3.0 wynik od 50% do 60%*, 2.0 wynik poniżej 50%. *przedziały prawostronnie otwarte
5. EK1: Weryfikacja osiągnięcia efektów uczenia odbywa się za pomocą egzaminu pisemnego z całości materiału. Ocena wystawiana jest zgodnie ze skalą: 5.0 wynik od 90% w górę, 4.5 wynik od 80% do 90%*, 4.0 wynik od 70% do 80%*, 3.5 wynik od 60% do 70%*, 3.0 wynik od 50% do 60%*, 2.0 wynik poniżej 50%. *przedziały prawostronnie otwarte

Literatura:

obowiązkowa:

1. M. Ben-Ari: Logika matematyczna w informatyce, Warszawa, 2005, WNT
2. W. Marek, W. Onyszkiewicz: Elementy logiki i teorii mnogości w zadaniach, Warszawa, 2004, PWN
3. A. W. Mostowski, Z. Pawlak: Logika dla inżynierów, Warszawa, 1970, PWN

zalecana/fakultatywna:

1. Hałkowska, Katarzyna; Piróg-Rzepecka, Krystyna; Słupecki, Jerzy: Logika matematyczna, PWN, 1999

Matematyka dyskretna nazwa przedmiotu
Discrete mathematics nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Podstawy programowania w Pythonie, podstawowe wiadomości z algebry, algebry liniowej oraz znajomość podstawowych własności dotyczących badania zbieżności szeregów. Nie jest wymagana pozytywna ocena z żadnego przedmiotu,

Cele przedmiotu:

1. Celem zajęć jest przedstawienie elementów kombinatoryki, teorii grafów, teorii liczb i kryptografii oraz wybranych metod matematycznych niezbędnych przy konstrukcji i analizie algorytmów.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW2	Student posiada umiejętność jasnego formułowania pytań, czynnego udziału w dyskusji i potrafi pracować w grupie nad niezbyt trudnymi zadaniami praktycznymi. Znajomość podstawowych pojęć, twierdzeń i algorytmów teorii grafów, podstawowych pojęć i algorytmów teorii liczb, kryptografii oraz pojęć i własności dotyczących równań rekurencyjnych i matematycznych podstaw informatyki.	I1_W01
Umiejętności Absolwent potrafi:		
EU2	Umiejętność zastosowania algorytmów do wyznaczania w grafach dróg (w szczególności zastosowanie algorytmów trasowania) i podgrafów spinających. Umiejętność sprawdzenia podstawowych własności grafu takich jak planarność, spójność wierzchołkowa i krawędziowa, istnienie cykli zawierających wybrane struktury (zbiory wierzchołków, zbiory krawędzi) w grafie wyznaczania dróg.	I1_U06b
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Student posiada umiejętność jasnego formułowania pytań, czynnego udziału w dyskusji i potrafi pracować w grupie nad niezbyt trudnymi zadaniami praktycznymi.	I1_K04

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
4	5	Z	18	9	0	9	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
-----	-------------	----------------	---------------

1	W	Indukcja matematyczna (zasada indukcji matematycznej, zasady minimum i maksimum) oraz rekurencja (definicje rekurencyjne, liczby Fibonacciego i rozwiązywanie równań rekurencyjnych)	3
2	W	Zasada szufladkowa Dirichleta i zasada włączania-wyłączania	1
3	W	Grafy: podstawowe pojęcia	2
4	W	Grafy: cykl Eulera i cykl Hamiltona	3
5	W	Grafy: drzewa, algorytmy Prima i Kruskala wyznaczające minimalnego drzewa spinającego	2
6	W	Grafy skierowane: Podstawowe pojęcia	1
7	W	Grafy skierowane: algorytmy Dijkstry i Bellmana-Forda zwracające minimalne drogi w grafach skierowanych	3
8	W	Grafy: grafy dwudzielne, skojarzenia i twierdzenie Halla spójność i twierdzenie Menger'a, sieci, przepływy, przekroje i twierdzenie Forda-Fulkersona	3
9	W	Grafy: kolorowanie grafów, algorytm zachłanny, algorytm RF i SL	2
10	W	Arytmetyka liczb całkowitych: podzielność, NWD, NWW, liczby pierwsze, algorytm Euklidesa, rozkład na czynniki pierwsze, równania diofantyczne	2
11	W	Arytmetyka modularna: twierdzenie Fermata, twierdzenie Eulera, chińskie twierdzenie o resztach	1
12	W	Kryptografia: klasyfikacja algorytmów szyfrujących, algorytm szyfrowania RSA, El Gamala, algorytm Diffiego-Hellmana	3
13	W	Kryptografia: szyfry strumieniowe i podpis cyfrowy	2
14	W	Matematyczne podstawy informatyki: automaty skończone, maszyna Turinga	2
16	C	Rozwiązywanie zadań dotyczących indukcji matematycznej oraz rekurencji.	1
17	C	Rozwiązywanie zadań dotyczących zasady szufladkowej Dirichleta, zasady włączania-wyłączania i kombinatoryki	1
18	C	Rozwiązywanie zadań dotyczących funkcji tworzących i równań rekurencyjnych	2
19	C	Rozwiązywanie zadań dotyczących podstawowych pojęć teorii grafów. Analiza przykładów grafów i ich własności.	1
20	C	Rozwiązywanie zadań dotyczących dróg i cykli Eulera oraz cykli Hamiltona w grafach.	1.5
21	C	Rozwiązywanie zadań dotyczących drzew i algorytmów Prima i Kruskala.	1
22	C	Rozwiązywanie zadań dotyczących grafów skierowanych oraz algorytmu Dijkstry i algorytmu Bellmana-Forda.	1.5
23	C	Rozwiązywanie zadań dotyczących grafów dwudzielnych, twierdzenie Halla i analiza problemów przepływu w sieciach.	1
24	C	Rozwiązywanie zadań dotyczących kolorowania grafów.	1
25	C	Rozwiązywanie zadań dotyczących arytmetyki liczb całkowitych, twierdzenia Fermata, twierdzenia Eulera i chińskiego twierdzenia o resztach oraz rozwiązywanie równań diofantycznych.	1
26	C	Rozwiązywanie zadań dotyczących algorytmu RSA, ElGamala, algorytmu Diffiego-Hellmana.	2
27	C	Rozwiązywanie zadań dotyczących szyfrowania strumieniowego i podpisu cyfrowego.	1
28	LK	Wprowadzenie do programowania w Pythonie i do wykorzystania pakietu SageMath	3

29	LK	Działania na liczbach całkowitych (algorytmy potęgowania w Z i potęgowania mod n, liczby pierwsze (generowanie, własności liczb pierwszych, rozkład liczby całkowitej na iloczyn potęg liczb pierwszych), równania diofantyczne z wykorzystaniem pakietu SageMath	2
30	LK	Algorytmy szyfrujące symetryczne i asymetryczne (w szczególności RSA, EL Gammala) Bezpieczeństwo szyfrów, klasyczna ataki na RSA z wykorzystaniem pakietu SageMath.	2
31	LK	Maszynowa reprezentacja grafów, generowanie i wizualizacja grafów, grafy losowe (podstawowe konstrukcje) oraz badanie podstawowych własności grafów z wykorzystaniem pakietu SageMath.	3
32	LK	Wykorzystanie pakietu SageMath do badania algorytmu Fleury'ego, algorytmu Prima, algorytmu Kruskala, algorytmu Dijkstry i algorytmu Bellmana-Forda	3
33	LK	Badanie sieci i analiza problemu przepływu w sieciach z wykorzystaniem pakietu SageMath.	1
34	LK	Algorytmy kolorowania grafów.	1

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury	45
2	Konsultacje przedmiotowe w bezpośrednim kontakcie z prowadzącym	34
3	Zaliczenie w bezpośrednim kontakcie z prowadzącym	10

Metody dydaktyczne:

ćwiczenia, laboratorium komputerowe

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, dyskusja, kolokwium, odpowiedź ustna, prezentacja, projekt, zaliczenie pisemne, zaliczenie ustne, sprawdzian wiadomości z wykładów

Kryteria oceny:

1. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji wiedzy.
2. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji umiejętności.
3. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji kompetencji społecznych.

Literatura:

obowiązkowa:

1. V. Bryant Aspekty kombinatoryki, Warszawa, 2007, WNT
2. S. Dasgupta, Ch. Papadimitriou, U. Vazirani Algorytmy, Warszawa, 2010, PWN
3. R. Graham, D. Knuth, O. Patashnik Matematyka konkretna, Warszawa, 2011, PWN
4. R. P. Grimaldi Discrete and combinatorial mathematics : an applied introduction, Boston, 2003, Addison-Wesley

5. A. Menezes, P. C. van Oorschot, S. A. Vanstone Kryptografia stosowana, Warszawa, 2005, WNT
6. K.A Ross, Ch.R.B. Wright Matematyka dyskretna, Warszawa, 2011, PWN

zalecana/fakultatywna:

1. W. Lipski, W. Marek Wprowadzenie do kombinatoryki, Warszawa, 1983, Pol. Akad. Nauk. Inst. Podstaw Informatyki
2. Pod redakcja M. Kubale Optymalizacja dyskretna : modele i metody kolorowania grafów, Warszawa, 2002, WNT
3. E. M. Reingold, J. Nievergelt, N. Deo Algorytmy kombinatoryczne, Warszawa, 1985, PWN
4. B. Schneier Kryptografia dla praktyków : protokoły, algorytmy i programy źródłowe w języku C, Warszawa, 2002, WNT

Metody obliczeniowe nazwa przedmiotu
Computational methods nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Znajomość matematyki na poziomie szkoły średniej oraz wstępnego kursu na poziomie szkoły wyższej
2. Umiejętność posługiwania się komputerami osobistymi na poziomie elementarnym
3. Podstawy programowania w języku C/C++

Cele przedmiotu:

1. Opanowanie podstaw teoretycznych w zakresie elementarnych metod numerycznych stosowanych w obliczeniach numerycznych za pomocą komputerów.
2. Nabycie praktycznych umiejętności w zakresie elementarnych metod numerycznych stosowanych w obliczeniach numerycznych za pomocą komputerów.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	podstawowe metody, techniki i narzędzia stosowane przy rozwiązywaniu prostych zadań informatycznych z zakresu analizy złożoności obliczeniowej algorytmów oraz implementacji języków programowania.	I1_W11
EW2	pojęcia z matematyki przydatne do formułowania i rozwiązywania złożonych zadań informatycznych z wykorzystaniem zagadnień analizy matematycznej, algebry, matematyki dyskretnej, logiki i teorii mnogości, metod probabilistycznych, statystyki i metod numerycznych.	I1_W01
Umiejętności Absolwent potrafi:		
EU1	wykorzystać nabytą wiedzę matematyczną do opisu procesów, tworzenia i analizy modeli matematycznych, analizy statystycznej wyników symulacji i pomiarów.	I1_U06b
EU2	planować i przeprowadzać proste eksperymenty obliczeniowe i wykorzystać je do rozwiązywania zadań informatycznych oraz wyciągać wnioski.	I1_U07b
Kompetencje społeczne Absolwent jest gotów do:		
EK1	pracy indywidualnie i w zespole informatyków, w tym także potrafi zarządzać swoim czasem oraz podejmować zobowiązania i dotrzymywać terminów.	I1_K03

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
---------	-------------	------------------------	-------------	---------------	-----------------	------------------------------	--------------	---------------	------------------------

5	5	E	18	9	0	9	0	0	BRAK DANYCH
---	---	---	----	---	---	---	---	---	-------------

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	Własności obliczeń numerycznych. Błędy obliczeń.	2
2	W	Metody rozwiązywania nieliniowych równań algebraicznych i ich układów.	2
3	W	Metody rozwiązywania układów liniowych równań algebraicznych.	2
4	W	Podstawy metod różnicowych.	2
5	W	Elementarne metody różnicowe rozwiązywania zagadnień z warunkiem początkowym dla równań różniczkowych zwyczajnych.	2
6	W	Elementarne metody różnicowe rozwiązywania zagadnień z warunkiem brzegowym dla równań różniczkowych zwyczajnych i eliptycznych równań różniczkowych cząstkowych.	2
7	W	Elementarne metody różnicowe rozwiązywania zagadnień z warunkiem początkowym i brzegowym dla równań różniczkowych cząstkowych zależnych od czasu.	2
8	W	Metody interpolacji.	2
9	W	Całkowanie numeryczne.	2
10	C	Rodzaje błędów w obliczeniach numerycznych, reprezentacja zmiennoprzecinkowa liczb rzeczywistych, standard IEEE 754. Własności zadań: uwarunkowanie zadań. Metody oceny błędów maszynowych. Zjawisko utraty cyfr znaczących przy odejmowaniu. Własności algorytmów: numeryczna poprawność, numeryczna stabilność	1
11	C	Metody rozwiązywania nieliniowych równań algebraicznych: Picarda, bisekcji, reguła falsi, Newtona, siecznych. Zbieżność metod iteracyjnych	1
12	C	Obliczanie norm wektorów i macierzy. Wskaźnik uwarunkowania macierzy. Eliminacja Gaussa. Dekompozycja LU macierzy pełnej. Metody bezpośrednie rozwiązywania układów liniowych równań algebraicznych z macierzą pełną Metody bezpośrednie rozwiązywania układów liniowych równań algebraicznych z macierzami rzadkimi. Algorytm Thomasa.	1
13	C	Metody iteracyjne rozwiązywania układów liniowych równań algebraicznych. Metody Jacobiego, Gaussa-Seidela, SOR. Kryteria zbieżności metod iteracyjnych	1
14	C	Podstawy przybliżeń różnicowych dla pochodnych funkcji. Wyznaczanie błędów obcięta przybliżeń. Metody różnicowe rozwiązywania zagadnień z warunkiem brzegowym dla równań różniczkowych zwyczajnych 2-go rzędu	1
15	C	Metody różnicowe rozwiązywania zagadnień z warunkiem początkowym dla równań różniczkowych zwyczajnych pierwszego rzędu. Metody bezpośrednie i pośrednie a kwestia stabilności numerycznej. Metody: bezpośrednia Eulera, pośrednia Eulera, reguła trapezów	1

16	C	Metody rozwiązywania zagadnień z warunkiem początkowym i brzegowym dla równań różniczkowych cząstkowych w przestrzeni jednowymiarowej, na przykładzie równań dyfuzji. Metody: Klasyczna metoda bezpośrednia, Metoda Laasonen, Metoda Cranka-Nicolson.	1
17	C	Interpolacja wielomianowa Lagrangea funkcji jednej zmiennej. Bazy: potęgowa, Lagrangea i Newtona wielomianów interpolacyjnych. Algorytm Hornera. Algorytm Nevillea. Zjawisko Rungego w interpolacji wielomianowej Lagrangea funkcji jednej zmiennej. Interpolacja wielomianowa Hermitea funkcji jednej zmiennej	1
18	C	Interpolacja funkcjami sklejanymi. Interpolacja biliniowa funkcji dwóch zmiennych. Numeryczne obliczanie całek oznaczonych. Kwadratury	1
19	LK	Rodzaje błędów w obliczeniach numerycznych, reprezentacja zmiennoprzecinkowa liczb rzeczywistych, standard IEEE 754. Własności zadań: uwarunkowanie zadań. Metody oceny błędów maszynowych. Zjawisko utraty cyfr znaczących przy odejmowaniu. Własności algorytmów: numeryczna poprawność, numeryczna stabilność	1
20	LK	Metody rozwiązywania nieliniowych równań algebraicznych: Picarda, bisekcji, reguła fałsi, Newtona, siecznych. Zbieżność metod iteracyjnych	2
21	LK	Obliczanie norm wektorów i macierzy. Wskaźnik uwarunkowania macierzy. Eliminacja Gaussa. Dekompozycja LU macierzy pełnej. Metody bezpośrednie rozwiązywania układów liniowych równań algebraicznych z macierzą pełną. Metody bezpośrednie rozwiązywania układów liniowych równań algebraicznych z macierzami rzadkimi. Algorytm Thomasa.	2
22	LK	Metody iteracyjne rozwiązywania układów liniowych równań algebraicznych. Metody Jacobiego, Gaussa-Seidela, SOR. Kryteria zbieżności metod iteracyjnych	2
23	LK	Podstawy przybliżeń różnicowych dla pochodnych funkcji. Wyznaczanie błędów obcięta przybliżeń. Metody różnicowe rozwiązywania zagadnień z warunkiem brzegowym dla równań różniczkowych zwyczajnych 2-go rzędu	2

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Konsultacje przedmiotowe (w bezpośrednim kontakcie z prowadzącym)	10
2	Egzaminy i zaliczenia w sesji (w bezpośrednim kontakcie z prowadzącym).	5
3	Przygotowanie do zajęć, w tym studiowanie zalecanej literatury.	45
4	Opracowanie wyników.	20
5	Przygotowanie raportu, projektu, prezentacji, dyskusji.	9

Metody dydaktyczne:

ćwiczenia laboratoryjne

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach

Kryteria oceny:

1. Na ocenę 5.0 student w pełni rozumie zagadnienia aproksymacji i modelowania numerycznego.
2. Na ocenę 5.0 student ma możliwość stosowania metod, technik i narzędzi informatycznych do napisania kodu stosującego metody analizy numerycznej.
3. Na ocenę 5.0 student w pełni rozumie zagadnienia przybliżeń numerycznych rozwiązań ścisłych problemów matematycznych.
4. Na ocenę 5.0 student rozwinął możliwość symulacji numerycznej zjawisk fizycznych.
5. Na ocenę 5.0 student potrafi zorganizować współpracę z innymi członkami zespołu w celu sprawnego wykonania powierzonych zadań informatycznych.

Literatura:

obowiązkowa:

1. D. Kincaid, W. Cheney, Analiza numeryczna, Warszawa, 2006, WNT
2. J. i M. Jankowscy, Przegląd metod i algorytmów numerycznych, cz. 1, Warszawa, 1981, WNT
3. J. i M. Jankowscy, Przegląd metod i algorytmów numerycznych. cz. 2, Warszawa, 1982, WNT

zalecana/fakultatywna:

1. A. Ralston, Wstęp do analizy numerycznej, Warszawa, 1983, PWN
2. A. Björck, G. Dahlquist, Metody numeryczne, Warszawa, 1983, PWN
3. Z. Fortuna, B. Macukow, L. Wąsowski, Metody numeryczne, Warszawa, 1982, WNT

Metody uczenia maszynowego nazwa przedmiotu
Machine learning methods nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Podstawy programowania ze wskazaniem na język Python.
2. Podstawowa wiedza teoretyczna i praktyczna w zakresie metod sztucznej inteligencji (zaliczenie przedmiotu Wstęp do Sztucznej Inteligencji).

Cele przedmiotu:

1. Zdobyć przez studentów wiedzy o podstawowych metodach uczenia maszynowego.
2. Umiejętność implementacji algorytmów uczenia maszynowego, znajomość popularnych bibliotek w wybranym środowisku programistycznym (zalecane: Python).

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	Znajomość bibliotek i frameworków przydatnych w implementacji tych metod.	I1_W08
EW2	Znajomość podstawowych problemów uczenia maszynowego i dedykowanych im metod.	I1_W04
Umiejętności Absolwent potrafi:		
EU1	Umiejętność implementacji wybranych metod uczenia maszynowego z wykorzystaniem dedykowanych bibliotek i frameworków.	I1_U23
EU2	Umiejętność przeprowadzenia analizy wydajności zaimplementowanych metod na wybranych zestawach danych testowych.	I1_U07b
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Student potrafi pracować indywidualnie i w grupie, potrafi w tym celu efektywnie korzystać z narzędzi informatycznych (platforma e-learningowa, komunikatory, media społecznościowe, narzędzia do nauczania zdalnego (np. MS TEAMS, ZOOM)), wykorzystując je pracy indywidualnej i kontaktu z nauczycielem.	I1_K05

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
5	3	Z	18	0	0	18	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	Uczenie maszynowe jako dział sztucznej inteligencji, klasyfikacja systemów uczących się i metod uczenia.	2
2	W	Uczenie nadzorowane i nienadzorowane - ogólna definicja i prosta analiza porównawcza. Analiza ryzyka.	2
3	W	Algorytm k-NN. - definicje i przykłady w oparciu o analizę danych.	2
4	W	Wybrane modele regresyjne. Drzewa decyzyjne i sieci neuronowe jednokierunkowe.	4
5	W	Maszyna Wektorów Nośnych (SVM).	2
6	W	Wprowadzenie do głębokich sieci neuronowych. Sieci konwolucyjne.	2
7	W	Uczenie nienadzorowane. Metody grupowania danych.	2
8	W	Sieci Hopfielda i mapy Kohonena.	2
9	LK	Implementacja i konfiguracja środowiska Python i biblioteki Scikit-learn, zapoznanie się ze strukturą biblioteki, analiza prostych przykładów.	4
10	LK	Zapoznanie się z bibliotekami pandas, seaborn, keras, numpy.	2
11	LK	Implementacja metod SVM, klasyfikatora Bayesa, k-NN i drzewa decyzyjnego do klasyfikacji danych medycznych, pozyskanie danych z ogólnodostępnych repozytoriów, pre-processing danych.	5
12	LK	Implementacja sieci głębokiej do rozwiązania problemu klasyfikacji obrazów medycznych. Wykonanie analizy eksperymentalnej.	5
13	LK	Prezentacja projektów zaliczeniowych.	2

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury.	15
2	Opracowanie wyników.	14
3	Przygotowanie i prezentacja sprawozdania i projektu -bezpośrednio kontakt z prowadzącym, udział w dyskusji.	5
4	Udział w konsultacjach w bezpośrednim kontakcie z prowadzącym	20

Metody dydaktyczne:

dyskusja, konsultacje, wykład, laboratorium komputerowe

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, dyskusja, prezentacja, projekt, sprawozdanie z pracy indywidualnej, sprawozdanie

Kryteria oceny:

1. Na ocenę 5,0 student zna biblioteki dedykowane nowym i klasycznym metodom uczenia maszynowego w różnych środowiskach programistycznych.
2. Na ocenę 5,0 student potrafi zdefiniować złożone problemy uczenia maszynowego, potrafi sklasyfikować i dobrać dedykowane im metody.
3. Na ocenę 5,0 student doskonale radzi sobie z implementacją zna i potrafi umiejętnie korzystać z dedykowanych bibliotek w różnych środowiskach programistycznych, potrafi skalować i optymalizować działanie zaimplementowanych algorytmów.
4. Na ocenę 5,0 student potrafi przeprowadzić wszechstronną analizę eksperymentalną i dokonać zaawansowanej analizy statystycznej otrzymanych wyników.
5. Na ocenę 5,0 student doskonale orientuje się w praktycznych zastosowaniach metod uczenia maszynowego i potrafi wytłumaczyć zalety ich stosowania w praktyce na przykładach flagowych praktycznych projektów, potrafi pracować w zespole i zaprezentować wyniki swojej pracy na forum grupy i forach zewnętrznych.

Literatura:

obowiązkowa:

1. M. Gągolewski, M. Bartoszek, A. Cena, - Przetwarzanie i analiza danych w języku Python, 2016, PWN.
2. M. Szeliga - Data science i uczenie maszynowe, Warszawa, 2017, PWN.

zalecana/fakultatywna:

1. A. Geron - Uczenie maszynowe z użyciem Scikit-Learn, Keras i TensorFlow. Wydanie III, Helion, 2023.

Modele biznesu w sektorze IT nazwa przedmiotu
Business models in the IT sector nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty ogólne kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Brak wymagań wstępnych.

Cele przedmiotu:

1. Zapoznanie studentów z istotą koncepcji modelu biznesu i jego zastosowaniem w działalności gospodarczej.
2. Wykształcenie umiejętności samodzielnego opisywania i projektowania modeli biznesu.
3. Doskonalenie kompetencji społecznych w zakresie samodzielnego kierowania przedsięwzięciami biznesowym.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	Student swobodnie posługuje się aparatem pojęciowym z zakresu modeli biznesu oraz wymienia i wyjaśnia ich główne składowe.	I1_W14
Umiejętności Absolwent potrafi:		
EU1	Student w oparciu o kluczowe zagadnienia modeli biznesu identyfikuje i opisuje istniejące modele biznesu.	I1_U01b
EU2	Student w oparciu o kluczowe zagadnienia projektuje własne modele biznesu.	I1_U01b
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Student myśli i działa w sposób przedsiębiorczy.	I1_K03

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
7	2	Z	9	9	0	0	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	Pojęcie i koncepcja modelu biznesowego. Ewolucja modeli biznesowych. Model biznesu a strategia firmy.	2
2	W	Elementy modelu biznesu (segmenty klientów, propozycje wartości, kanały komunikacji, relacje z klientami, strumienie przychodów, kluczowe zasoby, kluczowe działania, kluczowi partnerzy, struktura kosztów). Charakterystyka i dynamika relacji między nimi oraz determinanty wyboru.	3
3	W	Klasyfikacja modeli biznesu i uwarunkowania wyboru modelu dla danego przedsiębiorstwa.	1
4	W	Modele biznesu w sektorze IT.	3
5	C	Analiza wybranych modeli biznesu przedsiębiorstw sektora IT.	1
6	C	Zasady projektowania i implementacji modeli biznesu w odniesieniu do celów.	1
7	C	Aspekty rynkowe, organizacyjne i finansowe modelu biznesowego.	1
8	C	Operacjonalizacja modelu biznesu w formie kanwy modelu biznesu.	2
9	C	Prezentacja i dyskusja nad zaprojektowanymi modelami biznesu.	4

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury.	9
2	Przygotowanie raportu, projektu, prezentacji, dyskusji.	15
3	Konsultacje przedmiotowe w bezpośrednim kontakcie z prowadzącym	8

Metody dydaktyczne:

analiza przypadku, ćwiczenia, praca w grupach, projekt, wykład, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

projekt, test

Kryteria oceny:

1. Na ocenę 5.0 student poprawnie wyjaśnia procesy związane z tworzeniem i implementacją modeli biznesu posługując się w poprawny sposób językiem dyscypliny.
2. Na ocenę 5.0 student przeprowadza analizę przedsiębiorstwa pozwalającą na identyfikację modelu biznesu i jego składowych oraz przypisanie go do istniejących kategorii.

3. Na ocenę 5.0 student przygotowuje opis modelu biznesowego w wymiarze szczegółowych założeń i charakterystyki wszystkich składowych modelu biznesu oraz planuje ścieżkę rozwoju biznesu w oparciu o ten model.
4. Student akceptuje elastyczność i konieczność dostosowywania swoich postaw, decyzji, działań do szans i atutów oraz eliminacji zagrożeń i słabości towarzyszących aktywności zawodowej/gospodarczej.

Literatura:

obowiązkowa:

1. T. Waściński - Teoria budowy modeli biznesowych, Warszawa, 2019, Oficyna Wydawnicza Politechniki Warszawskiej
2. A. Osterwalder, Y. Pigneur - Tworzenie modeli biznesowych. Podręcznik wizjonera, Gliwice, 2021, One-Press

zalecana/fakultatywna:

1. P. van der Pijl, J. Lokitz, L. K. Solomon - Nowoczesne projektowanie modeli biznesowych. Narzędzia, kompetencje, mentalność i innowacje, które zapewnia firmie sukces, Gliwice, 2018, OnePress
2. T. Gołębiowski, T. M. Dudzik, M. Lewandowska, M. Witek-Hajduk - Modele biznesu polskich przedsiębiorstw, Warszawa, 2018, SGH
3. T. Doligalski - Modele biznesu polskich przedsiębiorstw, Warszawa, 2015, PWN
4. O. Gassmann, K. Frankenberger, M. Csik - Nawigator Modelu Biznesowego. 55 modeli, które zrewolucjonizują Twój biznes, Gliwice, 2016, Helion

Negocjacje i rozwiązywanie konfliktów nazwa przedmiotu
Negotiations and solving of social problems nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty ogólne kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. brak wymagań wstępnych

Cele przedmiotu:

1. Zapoznanie studentów z zasadami i technikami negocjacji i rozwiązywania konfliktów w różnych sytuacjach zawodowych i społecznych.
2. Wykształcenie umiejętności negocjowania i rozwiązywania konfliktów o różnym charakterze.
3. Uświadomienie studentom wagi twórczego poszukiwania rozwiązań niezbędnych w działalności zawodowej i społecznej.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	Zasady negocjacji i rozwiązywania konfliktów w różnych sytuacjach zawodowych i społecznych, wyjaśnia mechanizmy psychologiczne powstawania i rozwiązywania konfliktów	I1_W14
Umiejętności Absolwent potrafi:		
EU1	w sposób twórczy i samodzielny zaplanować i przeprowadzić negocjacje w różnych sytuacjach zawodowych i biznesowych.	I1_U01b
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Twórczego poszukiwania rozwiązań w różnych sytuacjach konfliktowych i negocjacyjnych	I1_K04

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
3	1	Z	0	9	0	0	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	C	Konflikty - struktura i typy konfliktów; interpersonalne, społeczne i organizacyjne przyczyny konfliktów. Strategie rozwiązywania	2
2	C	Podstawowe rozróżnienia w teorii i praktyce negocjacji. Podstawowe zasady i typy negocjacji; strategie i taktyki negocjacji. Negocjacje jako własność problemu.	2
3	C	Planowanie negocjacji, etapy negocjacji. Przygotowanie negocjacji	1
4	C	Techniki negocjacyjne, Zamknięcie negocjacji.	1
5	C	Podział ról w zespole negocjacyjnym. Emocje w negocjacjach, zasady i techniki wywierania wpływu. Argumentacja perswazyjna.	1
6	C	Przykładowe sytuacje negocjacyjne	2

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Konsultacje przedmiotowe w bezpośrednim kontakcie z prowadzącym	2
2	Przygotowanie do kolokwium i rozwiązania zadania problemowego	19

Metody dydaktyczne:

analiza przypadku, ćwiczenia, praca w grupach, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, kolokwium, rozwiązanie zadania problemowego

Kryteria oceny:

1. Student na ocenę 5.0 zna i potrafi zidentyfikować w konkretnych sytuacjach mechanizmy konfliktogenne oraz wyjaśnić zasady negocjacji w złożonych sytuacjach zawodowych i biznesowych.
2. Student na ocenę 5.0 potrafi wskazać cele negocjacji, zaplanować negocjacje, wskazać właściwe techniki negocjacyjne oraz doprowadzić do finalizacji procesu negocjacyjnego
3. Student na ocenę 5.0 dobiera właściwe metody negocjacji i jest gotów do twórczego rozwiązania problemu negocjacyjnego

Literatura:

obowiązkowa:

1. R.J. Lewicki, B. Barry, D.M. Saunders - Zasady negocjacji, Poznań, 2011, Rebis
2. W. Wilmot, J.L. Hocker - Konflikty między ludźmi, Warszawa, 2011, PWN

zalecana/fakultatywna:

1. Ch. W. Moore - Mediacje, Warszawa, 2012, Kluwer
2. D.T. Kenrick, S.L. Neuberg, R.B. Cialdini - Psychologia społeczna, Gdańsk, 2002, GWP

Pisanie tekstów specjalistycznych nazwa przedmiotu
Technical and Professional Writing nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty ogólne kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Znajomość na poziomie podstawowym edytorów tekstowych oraz zasad poprawnej polszczyzny.

Cele przedmiotu:

1. Zapoznanie studentów z zasadami pisania i redakcji tekstów specjalistycznych, w tym pracy dyplomowej.
2. Zapoznanie studentów z narzędziami informatycznymi do edycji tekstów.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
------------------------	-------------------------	-------------------------------------

Wiedza Absolwent zna i rozumie:		
EW1	Student zna słownictwo z dziedziny języka specjalistycznego związanego z kierunkiem studiów, rozumie różnice między językiem formalnym i nieformalnym oraz przekazem obiektywnym i nieobiektywnym. Zna strukturę podstawowych dokumentów, np. sprawozdania, raportu, pracy dyplomowej oraz zasady pisania dłuższych form pisemnych.	I1_W13
Umiejętności Absolwent potrafi:		
EU1	Student potrafi opracować teksty specjalistyczne zgodnie z zasadami redakcyjnymi wybranych edytorów, np. LaTeX, MS Word, przedstawić wybrane osiągnięcia w zakresie informatyki w oparciu o teksty specjalistyczne z w/w dziedziny.	I1_U01b
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Student widzi potrzebę przekazywania informacji i wiedzy w sposób etyczny, kompetentny, krótki, czytelny, wyczerpujący i uwzględniający możliwości i oczekiwania odbiorców.	I1_K04

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
4	2	Z	9	9	0	9	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
-----	-------------	----------------	---------------

1	W	Cel pisania tekstów naukowych i technicznych. Cechy i język tekstu naukowego i technicznego. Definicje i objaśnienia. Akapity. Rodzaje dokumentów: instrukcja, notatka, dokumenty typu project proposal (szablon), sprawozdanie, raport, praca dyplomowa. Struktura i proces pisania pracy naukowej. Rodzaje abstraktów, streszczenia. Znaczenie parafrazy. Rodzaje plagiatu. Etyczne podejście w pisaniu. Formaty cytowań. Zagadnienia gramatyczno-leksykalne: język formalny i nieformalny. Najczęstsze błędy w pisaniu. Omówienie najważniejszych funkcjonalności wybranych edytorów tekstowych. Student zna słownictwo z dziedziny języka specjalistycznego związanego z kierunkiem studiów, rozumie różnice między językiem formalnym i nieformalnym, oraz przekazem obiektywnym i nieobiektywnym. Zna strukturę podstawowych dokumentów, np. sprawozdania, raportu, pracy dyplomowej oraz zasady pisania dłuższych form pisemnych.	9
2	C	Cel pisania tekstów naukowych i technicznych. Cechy i język tekstu naukowego i technicznego. Definicje i objaśnienia. Akapity. Rodzaje dokumentów: instrukcja, notatka, dokumenty typu project proposal (szablon), sprawozdanie, raport, praca dyplomowa. Struktura i proces pisania pracy naukowej. Rodzaje abstraktów, streszczenia. Znaczenie parafrazy. Rodzaje plagiatu. Etyczne podejście w pisaniu. Formaty cytowań. Zagadnienia gramatyczno-leksykalne: język formalny i nieformalny. Najczęstsze błędy w pisaniu. Student widzi potrzebę przekazywania informacji i wiedzy w sposób etyczny, kompetentny, krótki, czytelny, wyczerpujący i uwzględniający możliwości i oczekiwania odbiorców.	9
3	LK	Realizacja zadań praktycznych. Podstawy formatowania tekstu: strona tytułowa, akapity, rozdziały, śródtytuły. Spis treści, spis ilustracji/tabel/listingów, bibliografia. Składanie tekstu, wstawianie elementów graficznych. Wyrażenia matematyczne. Student potrafi opracować teksty specjalistyczne zgodnie z zasadami redakcyjnymi wybranych edytorów, np. LaTeX, MS Word. Potrafi przedstawić wybrane osiągnięcia w zakresie informatyki w oparciu o teksty specjalistyczne z w/w dziedziny.	9

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Przygotowanie się do zajęć i dyskusji, w tym studiowanie zalecanej literatury, opracowanie wyników przeprowadzonych badań, przygotowanie raportu, projektu, prezentacji itp.	14
2	Konsultacje przedmiotowe i zaliczenie (w bezpośrednim kontakcie z prowadzącym)	9

Metody dydaktyczne:

dyskusja, praca w grupach, praca z tekstem/materiałem audio i wideo, projekt, wykład, laboratorium komputerowe, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, dyskusja, prezentacja, projekt, test, zaliczenie pisemne

Kryteria oceny:

1. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji wiedzy.
2. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji umiejętności.
3. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji kompetencji społecznych.

Literatura:

obowiązkowa:

1. Beata Stępień. Zasady pisania tekstów naukowych, Warszawa, 2022, PWN
2. Adam Wolański. Edycja tekstów Praktyczny poradnik, Warszawa, 2022, PWN

zalecana/fakultatywna:

1. Ewa Szczepaniak. Sztuka projektowania tekstów. Jak tworzyć treści, które podbiją internet, OnePress, 2019

Podstawy elektroniki i techniki cyfrowej nazwa przedmiotu
Fundamentals of Electronics and Digital Technology nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Wiadomości z fizyki z zakresu elektryczności

Cele przedmiotu:

1. Wprowadzenie w dziedzinę elektroniki, podstawowe prawa elektryczne, układy analogowe bierne i czynne, elementy elektroniczne półprzewodnikowe, układy elektroniki analogowej
2. Wprowadzenie w dziedzinę techniki cyfrowej, elementów i układów cyfrowych, których zrozumienie funkcjonowania jest niezbędne do dalszej nauki o budowie sprzętu komputerowego.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	Zrozumienie podstaw teorii obwodów, elektroniki analogowej i technik cyfrowych	I1_W03
Umiejętności Absolwent potrafi:		
EU1	Umiejętność wykonania obliczeń układów elektrycznych i elektronicznych, umiejętność projektowania układów cyfrowych.	I1_U01b
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Umiejętność testowania elementów i układów elektronicznych, projektowania i testowania układów elektroniki cyfrowej.	I1_K03

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
3	4	Z	18	0	0	18	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
-----	-------------	----------------	---------------

1	W	Podstawy elektroniki: Prąd, napięcie, podstawowe prawa w tym prawo Ohma, prawa Kirchoffa, Złącze półprzewodnikowe, Dioda, tranzystor - bipolarny, unipolarny, Wzmacniacze i układy przełączające, Teoria sprzężenia zwrotnego, Wzmacniacz różnicowy i operacyjny - zastosowania, Filtry analogowe, Charakterystyki amplitudowe i częstotliwościowe, częstotliwość graniczna, pasmo przenoszenia, Przetworniki cyfrowo-analogowe i analogowo-cyfrowe, Technika cyfrowa: Pojęcia podstawowe - podział układów cyfrowych, parametry układu cyfrowego, Układy kombinacyjne i sekwencyjne, Elementarne układy przełączające. Układy TTL, MOS i inne technologie, Podstawowe własności algebry Boole'a. Bramki logiczne, tablice prawdy, Realizacje złożonych funkcji logicznych i minimalizacja, tabele Karnaugh'a, Przerzutniki: typy, sposoby sterowania, własności dynamiczne, układy bistabilne, monostabilne i astabilne, Wprowadzenie do liczników, liczniki: asynchroniczne i synchroniczne, binarne i dziesiętne, liczniki rewersyjne, Rejestry: równoległe i szeregowo. Układy komutacyjne: multipleksery, demultipleksery, kodery, dekodery, konwertery kodów.	18
2	LK	Wirtualny warsztat elektronika, pakiet symulacji elementów i układów elektronicznych - poznanie środowiska, przykłady prostych obwodów, Badanie diody i tranzystora, charakterystyki prądowo napięciowe, Filtry pasywne i układy zasilające, Wzmacniacze tranzystorowe, różnicowe i operacyjne - podstawowe układy pracy, zastosowania, Pakiet symulacji elementów i układów cyfrowych, bramki logiczne, algebra Boole'a, Minimalizacja układów logicznych, tabele Karnaugh'a, Projektowanie układów kombinacyjnych, Przerzutniki.	18

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Konsultacje przedmiotowe (w bezpośrednim kontakcie z prowadzącym)	6
2	Sprawdziany i zaliczenia (w bezpośrednim kontakcie z prowadzącym)	4
3	Praca własna - przygotowanie do zajęć, przygotowanie sprawozdań	74

Metody dydaktyczne:

konsultacje, wykład, laboratorium komputerowe, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, kolokwium, odpowiedź ustna, test, sprawozdanie

Kryteria oceny:

1. Na ocenę 5,0 student nabył wiedzę treści wykładowych w stopniu co najmniej 90%
2. Na ocenę 5,0 student zalicza ćwiczenia laboratoryjne uzyskując średnią arytmetyczną ocen co najmniej 4,75
3. Na ocenę 5,0 student wykonuje zadania i oddaje je bez opóźnienia. W pracy zespołowej swoją część pracy wykonuje na czas.

Literatura:

obowiązkowa:

1. A. Skorupski - Podstawy elektroniki cyfrowej, Warszawa, WKiŁ
2. U. Tietze, Ch. Shenk - Układy półprzewodnikowe, Warszawa, WKiŁ
3. J. Kalisz - Podstawy elektroniki cyfrowej, Warszawa, WKiŁ

zalecana/fakultatywna:

1. B. Wilkinson - Układy cyfrowe, Warszawa, WKiŁ
2. M. Rusek, J. Pasierbiński - Elementy i układy elektroniczne, Warszawa, WNT
3. Piotr Malecki - materiały do wykładów, ii.pk.edu.pl/pmalecki/PEiTC

Podstawy przetwarzania w chmurze nazwa przedmiotu
Cloud Computing Fundamentals nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Wiedza i umiejętności z zakresu podstaw sieci komputerowych.
2. Wiedza i umiejętności z zakresu podstaw baz danych.

Cele przedmiotu:

1. Zapoznanie Studentów z ideą chmury obliczeniowej oraz uświadomienie jej wpływu na obecne oblicze świata technologii.
2. Zapoznanie Studentów z różnymi sposobami interakcji z usługami chmurowymi wybranego dostawcy, a także pełną konfiguracją i wszechstronnym użytkowaniem tych usług.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	Student rozumie ideę chmury obliczeniowej oraz mechanizmów w ramach niej działających.	I1_W05
EW1	Student rozumie ideę chmury obliczeniowej oraz mechanizmów w ramach niej działających.	I1_W10
EW1	Student rozumie ideę chmury obliczeniowej oraz mechanizmów w ramach niej działających.	I1_W12
Umiejętności Absolwent potrafi:		
EU1	Student potrafi kompleksowo konfigurować usługi chmurowe wybranego dostawcy.	I1_U12
EU1	Student potrafi kompleksowo konfigurować usługi chmurowe wybranego dostawcy.	I1_U13
EU1	Student potrafi kompleksowo konfigurować usługi chmurowe wybranego dostawcy.	I1_U15
EU1	Student potrafi kompleksowo konfigurować usługi chmurowe wybranego dostawcy.	I1_U16
EU1	Student potrafi kompleksowo konfigurować usługi chmurowe wybranego dostawcy.	I1_U20
EU1	Student potrafi kompleksowo konfigurować usługi chmurowe wybranego dostawcy.	I1_U23
EU2	Student potrafi wykorzystywać w szerokim zakresie usługi chmurowe wybranego dostawcy.	I1_U18
EU2	Student potrafi wykorzystywać w szerokim zakresie usługi chmurowe wybranego dostawcy.	I1_U20
EU2	Student potrafi wykorzystywać w szerokim zakresie usługi chmurowe wybranego dostawcy.	I1_U22
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Student jest gotów do jasnego tłumaczenia wpływu usług chmurowych na obraz współczesnego społeczeństwa informacyjnego.	I1_K05
EK1	Student jest gotów do jasnego tłumaczenia wpływu usług chmurowych na obraz współczesnego społeczeństwa informacyjnego.	I1_K06

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
7	4	E	18	0	0	18	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	Podstawy przetwarzania w chmurze: Czym jest chmura i dlaczego zmienia ona oblicze technologii i biznesu.	3
2	W	Modele chmur obliczeniowych oraz sposobów dostarczania usług. Idea wirtualizacji maszyn. Wykorzystywanie mocy obliczeniowej chmur. Procesy zachodzące w chmurze - przetwarzanie zadań obliczeniowych, szeregowanie, równoważenie obciążenia. Idea wirtualizacji sieci. Przestrzeń dyskowa w chmurze. Kontenery. Aplikacje w chmurze. Projektowanie, wdrażanie i monitorowanie w chmurze. Big Data oraz uczenie maszynowe w chmurze.	15
3	LK	Sposoby interakcji użytkownika z wybranym dostawcą usług. Realizacja usług obliczeń w ramach usług chmurowych opartych o wirtualizację maszyn. Pamięć masowa: Wdrażanie różnych modeli strukturalnej i niestukturalnej pamięci masowej. Usługi zarządzane: Różne opcje usług zarządzania aplikacjami w chmurze. Bezpieczeństwo: Zarządzanie bezpieczeństwem w chmurze. Sieci: Budowa bezpiecznej sieci w chmurze. Automatyzacja: Określanie narzędzi do automatyzacji i zarządzania chmurą.	13
4	LK	Big data: Zastosowanie różnorodnych zarządzanych usług Big Data w chmurze. Uczenie maszynowe: Zastosowanie usług opartych o uczenie maszynowe.	5

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Konsultacje przedmiotowe i egzamin w bezpośrednim kontakcie z prowadzącym	6
2	Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury; realizacja zadań i projektu; opracowanie wyników	60

Metody dydaktyczne:

projekt, wykład, laboratorium komputerowe, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, egzamin pisemny, kolokwium, projekt

Kryteria oceny:

1. EW1: Student na ocenę 5.0 definiuje wszystkie poznane na zajęciach pojęcia z zakresu chmur obliczeniowych: Modele chmur obliczeniowych oraz sposobów dostarczania usług, idea wirtualizacji maszyn, wykorzystywanie mocy obliczeniowej chmur, procesy zachodzące w chmurze - przetwarzanie zadań obliczeniowych, szeregowanie, równoważenie obciążenia, idea wirtualizacji sieci, przestrzeń dyskowa w chmurze, kontenery, aplikacje w chmurze, projektowanie, wdrażanie i monitorowanie w chmurze, Big Data oraz uczenie maszynowe w chmurze. Poszczególne oceny wyznaczane są na podstawie uzyskanych punktów w procesie weryfikacji efektów uczenia się. Na ocenę 2.0 Student nie spełnia warunków określonych dla oceny 3.0 Na ocenę 3.0 Uzyskanie co najmniej 51% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 3.5 Uzyskanie co najmniej 61% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 4.0 Uzyskanie co najmniej 71% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 4.5 Uzyskanie co najmniej 81% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 5.0 Uzyskanie co najmniej 91% punktów w ramach sposobów weryfikacji nabytych kompetencji.
2. EU1: Student na ocenę 5.0 swobodnie konfiguruje usługi chmurowe wybranego dostawcy. Student ponadto potrafi wykorzystać sposoby interakcji użytkownika z wybranym dostawcą usług, realizuje usługi obliczeń w ramach usług chmurowych opartych o wirtualizację maszyn, wdraża różne modele strukturalnej i niestrukturalnej pamięci masowej, wykorzystuje usługi zarządzane (różne opcje usług zarządzania aplikacjami w chmurze), dba o bezpieczeństwo (Zarządzanie bezpieczeństwem w chmurze), buduje sieci wirtualne, wykorzystuje narzędzia do automatyzacji i zarządzania chmurą. Poszczególne oceny wyznaczane są na podstawie uzyskanych punktów w procesie weryfikacji efektów uczenia się. Na ocenę 2.0 Student nie spełnia warunków określonych dla oceny 3.0 Na ocenę 3.0 Uzyskanie co najmniej 51% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 3.5 Uzyskanie co najmniej 61% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 4.0 Uzyskanie co najmniej 71% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 4.5 Uzyskanie co najmniej 81% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 5.0 Uzyskanie co najmniej 91% punktów w ramach sposobów weryfikacji nabytych kompetencji.
3. EU2: Student na ocenę 5.0 bezbłędnie wykorzystuje usługi dostawcy w zadanym problemie, w szczególności stosując paradygmaty Big Data oraz metody uczenia maszynowego. Poszczególne oceny wyznaczane są na podstawie uzyskanych punktów w procesie weryfikacji efektów uczenia się. Na ocenę 2.0 Student nie spełnia warunków określonych dla oceny 3.0 Na ocenę 3.0 Uzyskanie co najmniej 51% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 3.5 Uzyskanie co najmniej 61% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 4.0 Uzyskanie co najmniej 71% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 4.5 Uzyskanie co najmniej 81% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 5.0 Uzyskanie co najmniej 91% punktów w ramach sposobów weryfikacji nabytych kompetencji.
4. EK1: Student na ocenę 5.0 tłumaczy wpływ usług chmurowych na obraz współczesnego społeczeństwa informacyjnego. Poszczególne oceny wyznaczane są na podstawie uzyskanych punktów w procesie weryfikacji efektów uczenia się. Na ocenę 2.0 Student nie spełnia warunków określonych dla oceny 3.0 Na ocenę 3.0 Uzyskanie co najmniej 51% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 3.5 Uzyskanie co najmniej 61% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 4.0 Uzyskanie co najmniej 71% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 4.5 Uzyskanie co najmniej 81% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 5.0 Uzyskanie co najmniej 91% punktów w ramach sposobów weryfikacji nabytych kompetencji.

Literatura:

obowiązkowa:

1. Elton Stoneman - Nauka Kubernetesa w miesiąc, Gliwice, 2022, Helion
2. Michael Hausenblas, Stefan Schimanski - Kubernetes. Tworzenie natywnych aplikacji działających w chmurze, Gliwice, 2020, Helion
3. Microsoft: koordynator Jacek Orłowski - Chmura obliczeniowa. Korzyści społeczne, szanse gospodarcze, wyzwania regulacyjne, Wrocław, 2018, Presscom Sp.z o.o.

zalecana/fakultatywna:

1. Morris Kief - Infrastruktura jako kod. Dynamiczne systemy w epoce chmury, Warszawa, 2021, APN Promise

Podstawy sieci komputerowych nazwa przedmiotu
Computer networking fundamentals nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Znajomość podstaw elektroniki i techniki cyfrowej
2. Umiejętność konwersji pomiędzy systemami liczbowymi

Cele przedmiotu:

1. Zaznajomienie z podstawowymi aspektami dotyczącymi sieci komputerowych, zaznajomienie z podstawami działania i budowy sieci.

2. Poznanie modelu referencyjnego ISO/OSI w odniesieniu do mechanizmów dostępu do medium transmisyjnego dla wybranych technologii sieciowych.
3. Poznanie technologii i protokołów transmisji danych, adresowania urządzeń sieciowych oraz metod transmisji w sieciach rozległych.
4. Nabycie umiejętności konfiguracji specjalistycznego sprzętu sieciowego.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza		
Absolwent zna i rozumie:		
EW1	Wiedza na temat podstawowych aspektów dotyczących sieci komputerowych, podstaw działania i budowy sieci. Wiedza na temat modelu referencyjnego ISO/OSI w odniesieniu do mechanizmów dostępu do medium transmisyjnego dla różnych technologii sieciowych.	I1_W03
EW1	Wiedza na temat podstawowych aspektów dotyczących sieci komputerowych, podstaw działania i budowy sieci. Wiedza na temat modelu referencyjnego ISO/OSI w odniesieniu do mechanizmów dostępu do medium transmisyjnego dla różnych technologii sieciowych.	I1_W05
EW1	Wiedza na temat podstawowych aspektów dotyczących sieci komputerowych, podstaw działania i budowy sieci. Wiedza na temat modelu referencyjnego ISO/OSI w odniesieniu do mechanizmów dostępu do medium transmisyjnego dla różnych technologii sieciowych.	I1_W12
Umiejętności		
Absolwent potrafi:		
EU1	Umiejętność adresowania urządzeń sieciowych, projektowania podstawowych rozwiązań w lokalnych sieciach komputerowych, konfiguracji profesjonalnego sprzętu sieciowego w zakresie podstawowym.	I1_U01b
EU1	Umiejętność adresowania urządzeń sieciowych, projektowania podstawowych rozwiązań w lokalnych sieciach komputerowych, konfiguracji profesjonalnego sprzętu sieciowego w zakresie podstawowym.	I1_U03
EU1	Umiejętność adresowania urządzeń sieciowych, projektowania podstawowych rozwiązań w lokalnych sieciach komputerowych, konfiguracji profesjonalnego sprzętu sieciowego w zakresie podstawowym.	I1_U15

EU1	Umiejętność adresowania urządzeń sieciowych, projektowania podstawowych rozwiązań w lokalnych sieciach komputerowych, konfiguracji profesjonalnego sprzętu sieciowego w zakresie podstawowym.	I1_U23
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Umiejętność projektowania i konfiguracji sieci komputerowych w ramach pracy indywidualnej oraz zespołowej w sposób terminowy. Realizacja powierzonych zadań w sposób staranny i sumienny. Świadomość znaczenia realizowanych zagadnień w odniesieniu do pojęcia cyberbezpieczeństwa.	I1_K03
EK1	Umiejętność projektowania i konfiguracji sieci komputerowych w ramach pracy indywidualnej oraz zespołowej w sposób terminowy. Realizacja powierzonych zadań w sposób staranny i sumienny. Świadomość znaczenia realizowanych zagadnień w odniesieniu do pojęcia cyberbezpieczeństwa.	I1_K04

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
5	4	E	18	0	0	18	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
-----	-------------	----------------	---------------

1	W	W1 Wprowadzenie do sieci komputerowych, ewolucja, standardy, organizacje ustanawiające standardy, urządzenia sieciowe. W2 Model referencyjny ISO/OSI, zastosowania modelu. W3 Komunikacja i sieci komputerowe, typy sieci: sieci równorzędne, oparte na serwerach, sieci mieszane, topologie sieci: magistrali, pierścienia, gwiazdy, przełączana; topologie fizyczne i logiczne. W4 Topologie złożone: łańcuchy, hierarchie; przyłączanie stacji, serwera, przyłączanie do sieci WAN, przyłączanie do szkieletu. W5 Charakterystyka warstwy fizycznej: funkcje, znaczenie odległości, tłumienie, nośniki transmisji fizycznej, okablowanie strukturalne. W6 Warstwa łączy danych, składniki typowej ramki, ewolucja struktur ramek, sterowanie łączem logicznym. W7 Adresowanie, klasy adresów, zarezerwowane zakresy adresów, adresy specjalne, maski niestandardowe, utracona przestrzeń adresowa. W8 Technologie udostępniania informacji w sieciach komputerowych, mechanizmy dostępu do nośnika (rywalizacja, priorytet zadań, na zasadzie pierścienia). W9 Sieć lokalna Ethernet, funkcje warstwy łączy danych i warstwy fizycznej, omówienie sieci Fast Ethernet i Gigabit Ethernet. W10 Sieci pierścieniowe Token Ring, FDDI porównanie. W11 Wydajność sieci, zasięg, wydajność sieci Ethernet, Token Ring i FDDI porównanie wydajności i przepustowości sieci. W12 Podstawy sieci ATM, połączenia wirtualne, typy połączeń, szybkości przesyłania danych. W13 Sieci bezprzewodowe, przykład sieci bezprzewodowej WiFi. W14 Porównanie technologii lokalnych sieciowych komputerowych W15 Brama translacyjna, sieci VPN.	18
2	LK	Kodowanie transmisji. Wpływ zakłóceń na poprawność transmisji sygnału: transmisja jedнопrzewodowa, jedнопrzewodowa z bitem parzystości i różnicowa. Adresowanie urządzeń sieci lokalnych wg kryteriów: najlepszego wykorzystania przydzielonej puli adresów IP, minimalnego zużycia adresów, podziału na podsieci o określonej wielkości. Wyznaczanie adresów: sieciowego i rozgłoszeniowego. Identyfikacja klas sieci. Badanie wydajności lokalnej sieci komputerowej Ethernet w oparciu o topologie typu magistrala, gwiazda oraz siatka. Wyznaczanie rodzin charakterystyk sprawności sieci z uwzględnieniem występowania kolizji dla różnych obciążeń. Wprowadzenie do podstawowych narzędzi sieciowych systemu Windows: ipconfig, ping, tracert, route, netstat, nslookup, arp. Komunikacja w ramach jednej sieci - LAN switching, segmentacja sieci. Komunikacja pomiędzy sieciami - routing statyczny. Komunikacja pomiędzy sieciami - routing dynamiczny (RIPv1, RIPv2, EIGRP). Analiza tablic routingu. Konfiguracja specjalistycznego sprzętu sieciowego (przełączniki, routery).	18

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Konsultacje przedmiotowe (w bezpośrednim kontakcie z prowadzącym)	6
2	Egzaminy i zaliczenia (w bezpośrednim kontakcie z prowadzącym)	4
3	Praca własna - przygotowanie do zajęć i przygotowanie sprawozdań	74

Metody dydaktyczne:

konsultacje, wykład, laboratorium komputerowe, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, egzamin pisemny, odpowiedź ustna, test, sprawozdanie

Kryteria oceny:

1. Na ocenę 5,0 student nabył wiedzę treści wykładowych w stopniu co najmniej 90%
2. Na ocenę 5,0 student zalicza ćwiczenia laboratoryjne uzyskując średnią arytmetyczną ocen co najmniej 4,75
3. Na ocenę 5,0 student wykonuje zadania i oddaje je bez opóźnienia. W pracy zespołowej swoją część pracy wykonuje na czas.

Literatura:

obowiązkowa:

1. Mark Sportack - Sieci komputerowe, Warszawa, Helion
2. Barrie Sosinsky - Sieci komputerowe. Biblia. Wiedza obiecana, Gliwice, Helion
3. Mark Sportack - Podstawy adresowania IP, Warszawa, Mikom

zalecana/fakultatywna:

1. Douglas E. Comer - Sieci komputerowe TCPIP. Zasady, protokoły i architektura. Tom 1, Warszawa, BTC
2. Andrew S. Tanenbaum - Sieci Komputerowe, Warszawa, Helion
3. Adam Józefiok - W drodze do CCNA. Cz. 1, Gliwice, Helion

Podstawy transmisji danych nazwa przedmiotu
Introduction to data transmission nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Znajomość podstaw elektroniki i techniki cyfrowej

Cele przedmiotu:

1. Wprowadzenie do zagadnień przetwarzania sygnałów analogowych i cyfrowych pod kątem transmisji. Elementy metod częstotliwościowej i czasowo-częstotliwościowej analizy sygnałów cyfrowych.
2. Zapoznanie z podstawowymi elementami toru transmisyjnego, problemem zakłóceń transmisji, zagadnieniami sygnalizacji w powiązaniu z metodami detekcji sygnału po stronie odbiornika.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	Student zna różne typy sygnałów, rozumie istotę sygnału cyfrowego i potrafi uzasadnić przewagę takiej postaci sygnału w zastosowaniach praktycznych. Wie co to jest reprezentacja sygnału. Student zna metody analizy częstotliwościowej i czasowo-częstotliwościowej sygnałów cyfrowych, rozumie wpływ zakłóceń oraz zna metody eliminacji wpływu zakłóceń na proces detekcji	I1_W02
EW1	Student zna różne typy sygnałów, rozumie istotę sygnału cyfrowego i potrafi uzasadnić przewagę takiej postaci sygnału w zastosowaniach praktycznych. Wie co to jest reprezentacja sygnału. Student zna metody analizy częstotliwościowej i czasowo-częstotliwościowej sygnałów cyfrowych, rozumie wpływ zakłóceń oraz zna metody eliminacji wpływu zakłóceń na proces detekcji	I1_W09
Umiejętności Absolwent potrafi:		
EU1	Student potrafi analizować proces transmisji danych w warunkach zakłóceń. Posiada umiejętność doboru rodzaju oraz wyznaczenia parametrów filtrów, Student potrafi dobrać podstawowe elementy, parametry toru transmisyjnego. Umie ocenić wpływ parametrów poszczególnych elementów toru transmisji danych na jej jakość.	I1_U06b

EU1	Student potrafi analizować proces transmisji danych w warunkach zakłóceń. Posiada umiejętność doboru rodzaju oraz wyznaczenia parametrów filtrów, Student potrafi dobrać podstawowe elementy, parametry toru transmisyjnego. Umie ocenić wpływ parametrów poszczególnych elementów toru transmisji danych na jej jakość.	I1_U22
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Umiejętność przeprowadzania symulacji pracy toru transmisyjnego i przesyłania sygnałów cyfrowych	I1_K03

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
6	3	Z	9	0	0	18	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	Przegląd typów sygnałów i parametrów je opisujących, określenie sygnału cyfrowego, własności matematyczne przestrzeni sygnałów., iloczyn skalarny, norma, metryka w przestrzeni sygnałów, transmisja sygnału jednym przewodem sygnałowym, zakłócenia, błędy transmisji, transmisja sygnału przewodami różnicowymi, wpływ zakłóceń, redukcja sygnału zakłócającego po stronie odbiorczej, model przepływu danych, modelowanie transmisji sygnału w przewodach miedzianych za pomocą elementów czwórnikowych, transmisja sygnałów w światłowodach, okno optyczne, światłowody wielomodowe i jednomodowe, światłowody gradientowe, tłumienność, transmisja sygnału w skrętce miedzianej, rodzaje i klasy przewodów, skrętki wielożyłowe, pasmo przenoszenia, częstotliwość graniczna, filtry częstotliwościowe, akwizycja sygnałów dyskretnych - próbkowanie sygnałów ciągłych, twierdzenie o próbkowaniu, błąd aliasingu, układy próbkujące pamiętające, charakterystyki układów, parametry, modulacja sygnałów analogowych, metody modulacji, charakterystyki czasowe i częstotliwościowe, analiza porównawcza systemów modulacji.	9

2	LK	Badanie transmisji jedнопrzewodowej, zakłócenia, błędy transmisji, Badanie transmisji różnicowej, zakłócenia, błędy transmisji, wpływ siły i odległości sygnału zakłócającego, znaczenie parametrów okablowania dwużyłowego, modelowanie medium transmisyjnego przewodu dwużyłowego w oparciu o czwórnik impedancyjne, wyznaczenie odpowiedzi impulsowej układów liniowych, analiza wpływu wybranej metody modulacji na błąd detekcji sygnału po stronie odbiornika, kodowanie sygnałów, analiza metod weryfikacji poprawności przesyłanych danych, kody korekcyjne.	18
---	----	--	----

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Konsultacje przedmiotowe (w bezpośrednim kontakcie z prowadzącym)	6
2	Kolokwium i zaliczenia (w bezpośrednim kontakcie z prowadzącym)	4
3	Praca własna - przygotowanie do zajęć i przygotowanie sprawozdań	53

Metody dydaktyczne:

konsultacje, wykład, laboratorium komputerowe, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, kolokwium, odpowiedź ustna, test, sprawozdanie

Kryteria oceny:

1. Na ocenę 5,0 student nabył wiedzę treści wykładowych w stopniu co najmniej 90%
2. Na ocenę 5,0 student zalicza ćwiczenia laboratoryjne uzyskując średnią arytmetyczną ocen co najmniej 4,75
3. Na ocenę 5,0 student wykonuje zadania i oddaje je bez opóźnienia. W pracy zespołowej swoją część pracy wykonuje na czas.

Literatura:

obowiązkowa:

1. Tomasz P. Zieliński - Cyfrowe przetwarzanie sygnałów, Warszawa, Wydawnictwo Komunikacji i Łączności
2. Ulrich Tietze, Christoph Schenk - Układy półprzewodnikowe, Warszawa, WNT
3. Jerzy Szabatin - Podstawy przetwarzania sygnałów, Warszawa

zalecana/fakultatywna:

1. T. Chmaj, M. Lankosz - Akwizycja i przetwarzanie sygnałów cyfrowych, Kraków

Praktyka

nazwa przedmiotu

Student internship nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Zaliczenie co najmniej trzeciego semestru studiów.
2. Złożenie dokumentów zgodnych z procedurą organizacji studenckich praktyk zawodowych.

Cele przedmiotu:

1. Praktyczne zapoznanie się z procesami technologicznymi, metodami, modelami i narzędziami informatycznymi w branżach właściwych dla kierunku Informatyka.
2. Nabycie umiejętności samodzielnego projektowania, obsługi i utrzymania narzędzi systemów w branżach właściwych dla kierunku Informatyka oraz opracowywania dokumentacji i sprawozdań z osiągniętych rezultatów.
3. Nabycie umiejętności pracy w zespole.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		

EW1	Student zna i rozumie podstawy tworzenia i funkcjonowanie modeli, systemów i oprogramowania używanych w branżach związanych z kierunkiem Informatyka. Zna najczęstsze problemy związane z ich użytkowaniem.	I1_W10
EW1	Student zna i rozumie podstawy tworzenia i funkcjonowanie modeli, systemów i oprogramowania używanych w branżach związanych z kierunkiem Informatyka. Zna najczęstsze problemy związane z ich użytkowaniem.	I1_W12
Umiejętności Absolwent potrafi:		
EU1	Student ma podstawowe umiejętności w zakresie tworzenia i posługiwania się narzędziami inżynierskimi właściwymi dla kierunku Informatyka. Potrafi samodzielnie rozwiązywać podstawowe problemy związane z ich użytkowaniem. Student potrafi współpracować w zespole, umie efektywnie zagospodarować swój czas pracy.	I1_U05
EU1	Student ma podstawowe umiejętności w zakresie tworzenia i posługiwania się narzędziami inżynierskimi właściwymi dla kierunku Informatyka. Potrafi samodzielnie rozwiązywać podstawowe problemy związane z ich użytkowaniem. Student potrafi współpracować w zespole, umie efektywnie zagospodarować swój czas pracy.	I1_U23
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Student jest gotów zasięgać opinii doświadczonych współpracowników przy rozwiązywaniu napotkanych problemów. Zna zasady etyki zawodowej i tradycji panujących w firmie i jest gotów do ich przestrzegania.	I1_K02
EK1	Student jest gotów zasięgać opinii doświadczonych współpracowników przy rozwiązywaniu napotkanych problemów. Zna zasady etyki zawodowej i tradycji panujących w firmie i jest gotów do ich przestrzegania.	I1_K03
EK1	Student jest gotów zasięgać opinii doświadczonych współpracowników przy rozwiązywaniu napotkanych problemów. Zna zasady etyki zawodowej i tradycji panujących w firmie i jest gotów do ich przestrzegania.	I1_K06

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
6	6	Z	0	0	0	0	0	0	160

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	PZ	Szkolenie BHP oraz zapoznanie z miejscem odbywania praktyki (zasady panujące w Firmie, zakres obowiązków, szkolenie stanowiskowe) prowadzone przez Firmę przyjmującą na praktykę.	15
2	PZ	Praca pod nadzorem Opiekuna praktyki na wyznaczonym stanowisku: zapoznanie się z używanymi w firmie systemami i technologiami, nabieranie umiejętności praktycznych w zakresie ich użytkowania, poznawanie zasad pracy w danym zespole. Realizacja zadań ustalonych z Opiekunem.	140
3	PZ	Opracowanie, w uzgodnieniu z Opiekunem praktyki, pisemnego sprawozdania z odbytej praktyki.	5

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
-----	--------------------	---------------

Metody dydaktyczne:

praktyka zawodowa

Sposoby weryfikacji i oceny efektów uczenia się:

sprawozdanie

Kryteria oceny:

1. Złożenie w terminie dokumentów wymaganych do zaliczenia praktyki. Pozytywna opinia Opiekuna praktyki podkreślająca bardzo dobre opanowanie umiejętności Studenta w zakresie efektywnego zagospodarowania czasu pracy i współpracy w zespole. Dodatkowo na ocenę 5.0: Ocena przebiegu praktyki przez Opiekuna Praktykanta na minimum 5.0. Ocena realizacji efektu uczenia się na podstawie sprawozdania z odbytej praktyki na 5.0.
2. Złożenie w terminie dokumentów wymaganych do zaliczenia praktyki. Pozytywna opinia Opiekuna praktyki podkreślająca bardzo dobre opanowanie umiejętności Studenta w zakresie tworzenia i posługiwania się narzędziami inżynierskimi właściwymi dla studiowanego kierunku. Dodatkowo na ocenę 5.0: Ocena przebiegu praktyki przez Opiekuna Praktykanta na minimum 5.0. Ocena realizacji efektu uczenia się na podstawie sprawozdania z odbytej praktyki na 5.0.
3. Złożenie w terminie dokumentów wymaganych do zaliczenia praktyki. Pozytywna opinia Opiekuna praktyki podkreślająca bardzo dobre zrozumienie podstawy tworzenia i funkcjonowania modeli, systemów i oprogramowania używanych w branżach związanych z kierunkiem studiów. Dodatkowo na ocenę 5.0: Ocena przebiegu praktyki przez Opiekuna Praktykanta na minimum 5.0. Ocena realizacji efektu uczenia się na podstawie sprawozdania z odbytej praktyki na 5.0

Literatura:

obowiązkowa:

1. Odpowiednia do konkretnego tematu realizowanego w trakcie praktyki

zalecana/fakultatywna:

1. Zaproponowana przez opiekuna praktyk, odpowiednia do konkretnego tematu realizowanego w trakcie praktyki

Prawne aspekty działalności w branży IT nazwa przedmiotu
Legal aspects of activities in the IT industry nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty ogólne kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. brak

Cele przedmiotu:

1. Pogłębienie wiedzy z zakresu prawnych i ekonomicznych uwarunkowań aktywności zawodowej (w tym prowadzenia działalności gospodarczej oraz zasad prawnej ochrony pracy)
2. Wykształcenie umiejętności analizy prawnej i ekonomicznej podejmowanych działań w zakresie aspektów związanych z działalnością zawodową
3. Uświadomienie, że podejmowanie decyzji zawodowych, wymaga brania pod uwagę uwarunkowań prawnych i ekonomicznych w kontekście etyki i odpowiedzialności za życie innych ludzi i środowisko

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza		
Absolwent zna i rozumie:		
EW1	podstawowe zagadnienia z zakresu kodeksów etycznych dotyczących informatyki, rozumie zagrożenia związane z przestępczością elektroniczną, nt. patentów, ustawy prawo autorskie i prawa pokrewne oraz ustawy o ochronie danych osobowych.; podstawowe pojęcia z zakresu ekonomii, zarządzania i prowadzenia działalności gospodarczej w sektorze informatycznym.	I1_W13
EW1	podstawowe zagadnienia z zakresu kodeksów etycznych dotyczących informatyki, rozumie zagrożenia związane z przestępczością elektroniczną, nt. patentów, ustawy prawo autorskie i prawa pokrewne oraz ustawy o ochronie danych osobowych.; podstawowe pojęcia z zakresu ekonomii, zarządzania i prowadzenia działalności gospodarczej w sektorze informatycznym.	I1_W14
Umiejętności		
Absolwent potrafi:		
EU1	dostrzegać aspekty społeczne i prawne przy formułowaniu i rozwiązywaniu zadań informatycznych.	I1_U08
Kompetencje społeczne		
Absolwent jest gotów do:		
EK1	podjęcia roli społecznej absolwenta uczelni technicznej i zrozumienia potrzeby przekazywania opinii społecznej informacji dotyczących aktualnego stanu wiedzy w reprezentowanej dyscyplinie na ogólny rozwój techniki i komunikacji społecznej.	I1_K06

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
6	1	Z	0	9	0	0	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	C	ASPEKTY PRAWNE DZIAŁALNOŚCI ZAWODOWEJ (Rodzaje aktywności zawodowej w świetle prawa. Działalność gospodarcza i jej formy organizacyjno-prawne, zasady prowadzenia i obowiązki, w tym: przedsiębiorstwa typu start-up. Ochrona własności intelektualnej wynalazki i innowacje)	3
2	C	ASPEKTY PRAWNE Z OCHRONY OCHRONY PRAWNEJ RÓŻNYCH FORM DZIAŁALNOŚCI ZAWODOWEJ (Rodzaje aktywności zawodowej i ich prawne modele ochrony - Prawo pracy i ochrona uprawnień pracowniczych)	3
3	C	ASPEKTY EKONOMICZNE - FINANSOWE DZIAŁALNOŚCI ZAWODOWEJ (Źródła finansowania działalności zawodowej, aspekty podatkowe prowadzenia różnych form działalności zawodowej, modele organizacyjne, biznes plan i planowanie rozwoju firmy)	3

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Przygotowanie się do zajęć.	15
2	Konsultacje przedmiotowe i zaliczenie w bezpośrednim kontakcie z prowadzącym	6

Metody dydaktyczne: wykład, konsultacje, dyskusja, analiza przypadku

Sposoby weryfikacji i oceny efektów uczenia się: uczestnictwo w zajęciach, odpowiedź ustna, test

Kryteria oceny:

1. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji wiedzy.
2. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji umiejętności.
3. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji kompetencji społecznych.

Literatura:

obowiązkowa:

1. Konstytucja RP
2. ustawa Kodeks Cywilny

zalecana/fakultatywna:

1. ustawa O prawach autorskich i prawach pokrewnych
2. ustawa Prawo przedsiębiorców, Kodeks Spółek Handlowych

Wprowadzenie do programowania gier nazwa przedmiotu
Introduction to game programming nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Techniki programowania
2. Grafika komputerowa
3. Podstawy fizyki

Cele przedmiotu:

1. Zapoznanie studentów z metodami programowania grafiki uwzględniającymi nowe trendy w dziedzinie
2. Zapoznanie studentów z podstawami programowania gier (warstwa mechaniki gry, modelowanie postaci i otoczenia, warstwa obliczeniowa)
3. Rozwój kompetencji związanych pracą nad projektem gry

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
------------------------	-------------------------	-------------------------------------

Wiedza Absolwent zna i rozumie:		
EW3	Zna technologie wykorzystywane programowaniu grafiki komputerowej.	I1_W01
EW3	Zna technologie wykorzystywane programowaniu grafiki komputerowej.	I1_W02
EW3	Zna technologie wykorzystywane programowaniu grafiki komputerowej.	I1_W06
EW3	Zna technologie wykorzystywane programowaniu grafiki komputerowej.	I1_W07
EW4	Zna narzędzia wykorzystywanych w tworzeniu gier.	I1_W01
EW4	Zna narzędzia wykorzystywanych w tworzeniu gier.	I1_W02
EW4	Zna narzędzia wykorzystywanych w tworzeniu gier.	I1_W06
EW4	Zna narzędzia wykorzystywanych w tworzeniu gier.	I1_W07
Umiejętności Absolwent potrafi:		
EU2	Umie zastosować zagadnienia związane z fizyką, obróbką graficzną, technikami obliczeniowymi w programowaniu gier	I1_U12
Kompetencje społeczne Absolwent jest gotów do:		
EK2	Umie pracować indywidualnie i w zespole, współpracować ze specjalistami z różnych dziedzin	I1_K03

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
8	6	E	18	0	0	18	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	LK	Przygotowanie gry w środowisku PyGame - (środowisko, postać i animacja, symulacje, interfejs użytkownika, integracja)	6

2	LK	Przygotowanie gry w środowisku Unity/MonoGame - (środowisko, postać i animacja, symulacje, interfejs użytkownika, integracja)	6
4	W	Wprowadzenie do tematyki gier. Technologie i systematyka gier	1
5	W	Środowiska programowania grafiki	2
6	W	Metody wizualizacji i programowania grafiki	2
7	W	Przegląd środowisk implementacyjnych dla gier komputerowych	1
8	W	Biblioteki graficzne w Pythonie. PyGame.	1
9	W	Programowanie gry 2D w środowisku Python	1
10	W	Programowanie w środowisku Unity/Monogame	2
16	LK	Projekt i implementacja gry 2D w wybranym środowisku w zespołach.	6
17	W	Projektowanie gier - tworzenie koncepcji gry, przygotowanie scenariuszy, fabuły, budowa świata gry, projektowanie postaci, narracja w grze, projektowanie interakcji, budowa interfejsu użytkownika.	2
18	W	Matematyczne i fizyczne podstawy programowania gier. Reprezentacje danych i algorytmy geometryczne w grach. Ruch i dynamika w grach.	2
19	W	Wykorzystanie kontrolerów w grach.	2
20	W	Trendy i technologie w nowoczesnych grach komputerowych.	2

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Konsultacje przedmiotowe (w bezpośrednim kontakcie z prowadzącym)	25
2	Egzaminy i zaliczenia (w bezpośrednim kontakcie z prowadzącym)	14
3	Przygotowanie treści teoretycznych do egzaminu	20
4	Praca własna w zakresie realizowanych ćwiczeń i projektów	55

Metody dydaktyczne:

konsultacje, projekt, wykład, laboratorium komputerowe

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, egzamin pisemny, egzamin ustny, prezentacja, projekt, sprawozdanie

Kryteria oceny:

1. Student na ocenę 5.0 zna różne środowiska technologii programowania grafiki, zna różnice, wady i zalety, potrafi wybrać odpowiednia technologie dla charakterystyki gry.
2. Student na ocenę 5.0 zna związki pomiędzy narzędziami a technologiami programowania gier i potrafi wykorzystać pełne środowisko tworzenia gry.
3. Student na ocenę 5.0 potrafi wybrać odpowiednie algorytmy, metody, techniki i je wykorzystać w programowaniu gier.
4. Student na ocenę 5.0 realizuje projekt gry w zespole interdyscyplinarnym, wykazuje aktywność w realizacji projektu.

Literatura:

obowiązkowa:

1. E. Adams, Projektowanie gier - podstawy, Gliwice, 2011, Helion
2. I. Kalb, Python zorientowany obiektowo, programowanie gier i graficznych interfejsów użytkownika, 2022, Helion.
3. E. Ross, J. Ross, Unity i C# : podstawy programowania gier, Gliwice, 2018, Helion

zalecana/fakultatywna:

1. Bruno Miguel, Teixeira de Sousa, Programowanie gier - kompendium, 2003, Helion

Programowanie interfejsów graficznych nazwa przedmiotu
Programming user interfaces nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Podstawy grafiki komputerowej
2. Podstawowy kurs programowania

Cele przedmiotu:

1. Zdobyć wiedzy o rodzajach interakcji i typach interfejsów graficznych użytkownika.
2. Umiejętność posługiwania się narzędziami do projektowania interfejsów graficznych.
3. Nabycie umiejętności programowania interfejsu w środowiskach graficznych.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW2	Zna podstawowe pojęcia z zakresu projektowania interfejsów użytkownika oraz sposoby interakcji z użytkownikiem.	I1_W07
EW3	Zna zasady projektowania aplikacji zawierających graficzny interfejs użytkownika oraz mechanizmy generowania zdarzeń i sposoby ich obsługi uwzględniające reguły projektowania skoncentrowanego na użytkowniku.	I1_W07
EW4	Zna popularne komponenty graficznego interfejsu użytkownika.	I1_W07
Umiejętności Absolwent potrafi:		
EU2	Posługuje się narzędziami i technikami wspomagającymi projektowanie interfejsu użytkownika.	I1_U12
EU3	Umie zaprojektować graficzny interfejs użytkownika.	I1_U12
EU4	Umie zaprogramować elementy graficznego interfejsu użytkownika w wybranych środowiskach.	I1_U12
Kompetencje społeczne Absolwent jest gotów do:		
EK2	Wykazuje kreatywność przy rozwiązywaniu problemów i potrafi pracować w zespole.	I1_K03

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
6	4	Z	9	0	0	18	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	Definicja systemu informacyjnego oraz proces tworzenia GUI dla systemu informacyjnego.	1
2	W	Środowiska i elementy graficznego interfejsu użytkownika	1

3	W	Projektowanie GUI: zasady, wymagania, etapy projektu, wzorce projektowe, narzędzia projektowe.	2
4	W	Zagadnienia funkcjonalności GUI, wskazówki projektowe dla GUI.	1
5	W	Elementy i funkcje środowiska Swing.	2
7	LK	Budowa GUI w środowisku Swing - podstawowe kontrolki.	1
8	LK	Budowa GUI w środowisku Swing - obsługa zdarzeń.	1
9	LK	Budowa GUI w środowisku Swing - zaawansowane komponenty GUI.	1
10	LK	Budowa GUI w środowisku Swing - integracja.	2
11	LK	Budowa GUI w środowisku QT - wprowadzenie, podstawowe kontrolki.	1
12	LK	Budowa GUI w środowisku QT - obsługa zdarzeń.	1
13	LK	Budowa GUI w środowisku QT - zaawansowane funkcje.	1
14	LK	Budowa GUI w środowisku QT - integracja.	2
15	LK	Projekt GUI - Figma	2
16	LK	Implementacja GUI w wybranym środowisku.	6
17	W	Elementy i funkcje środowiska QT.	2

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Konsultacje przedmiotowe (w bezpośrednim kontakcie z prowadzącym)	10
2	Zaliczenia (w bezpośrednim kontakcie z prowadzącym)	5
3	Praca własna w zakresie realizowanych ćwiczeń	53
4	Przygotowanie do sprawdzianu z wiedzy teoretycznej	10

Metody dydaktyczne:

projekt, wykład, laboratorium komputerowe

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, kolokwium, projekt, sprawozdanie z pracy indywidualnej

Kryteria oceny:

1. Na ocenę 5.0 wykazuje bardzo dużą kreatywność przy rozwiązywaniu problemów.
2. Na ocenę 5.0 z łatwością posługuje się wybranymi narzędziami i technikami wspomagającymi projektowanie interfejsu użytkownika.
3. Na ocenę 5.0 umie bardzo dobrze zaprojektować graficzny interfejs użytkownika.
4. Na ocenę 5.0 umie bardzo dobrze zaprogramować elementy graficznego interfejsu użytkownika w wybranych środowiskach.
5. Na ocenę 5.0 zna bardzo dobrze podstawowe pojęcia z zakresu projektowania interfejsów użytkownika oraz sposoby interakcji z użytkownikiem.
6. Na ocenę 5.0 zna bardzo dobrze zasady projektowania aplikacji zawierających graficzny interfejs użytkownika oraz mechanizmy generowania zdarzeń i sposoby ich obsługi uwzględniające reguły projektowania skoncentrowanego na użytkowniku.
7. Na ocenę 5.0 zna dobrze większość komponentów graficznego interfejsu użytkownika.

Literatura:

obowiązkowa:

1. Joel Spolsky: Projektowanie interfejsu użytkownika. Poradnik dla programisty, 2001, Mikom
2. Pieta, Ścibosz, Wiśniewski: Podstawy tworzenia interfejsu graficznego aplikacji desktopowych w języku Java, Warszawa, 2020, Wydawnictwo Politechniki Warszawskiej
3. M. Sommerfield: Biblioteki Qt : zaawansowane programowanie przy użyciu C++, Gliwice, 2014, Helion

zalecana/fakultatywna:

1. Jenifer Tidwell: Projektowanie interfejsów - sprawdzone wzorce projektowe. 2012, Helion

Programowanie mobilne nazwa przedmiotu
Mobile programming nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Znajomość co najmniej jednego języka programowania obiektowego.
2. Podstawowa wiedza z zakresu technologii sieciowej, systemów operacyjnych i baz danych.

Cele przedmiotu:

1. Zapoznanie się z metodami i zasadami projektowania i implementowania aplikacji mobilnych.
2. Zapoznanie się z dostępnymi środowiskami implementowania aplikacji mobilnych.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	Zna zasady projektowania i tworzenia aplikacji mobilnych. Zna różnice pomiędzy dostępnymi platformami mobilnymi.	I1_W05
EW1	Zna zasady projektowania i tworzenia aplikacji mobilnych. Zna różnice pomiędzy dostępnymi platformami mobilnymi.	I1_W06
EW1	Zna zasady projektowania i tworzenia aplikacji mobilnych. Zna różnice pomiędzy dostępnymi platformami mobilnymi.	I1_W07
EW1	Zna zasady projektowania i tworzenia aplikacji mobilnych. Zna różnice pomiędzy dostępnymi platformami mobilnymi.	I1_W09
Umiejętności Absolwent potrafi:		
EU1	Potrafi implementować natywne aplikacje mobilne na systemy Android i iOS.	I1_U01b
EU1	Potrafi implementować natywne aplikacje mobilne na systemy Android i iOS.	I1_U11
EU1	Potrafi implementować natywne aplikacje mobilne na systemy Android i iOS.	I1_U12
EU1	Potrafi implementować natywne aplikacje mobilne na systemy Android i iOS.	I1_U20
EU1	Potrafi implementować natywne aplikacje mobilne na systemy Android i iOS.	I1_U21
EU1	Potrafi implementować natywne aplikacje mobilne na systemy Android i iOS.	I1_U22
EU1	Potrafi implementować natywne aplikacje mobilne na systemy Android i iOS.	I1_U23
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Umie pracować indywidualnie i w grupie oraz przekazywać uzyskane rezultaty pracy w zrozumiały sposób.	I1_K03
EK1	Umie pracować indywidualnie i w grupie oraz przekazywać uzyskane rezultaty pracy w zrozumiały sposób.	I1_K04

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
---------	-------------	------------------------	-------------	---------------	-----------------	------------------------------	--------------	---------------	------------------------

8	6	E	18	0	0	18	0	0	BRAK DANYCH
---	---	---	----	---	---	----	---	---	-------------

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	Wprowadzenie do systemów mobilnych.	2
2	W	Responsive Web Design - tworzenie mobilnych aplikacji webowych.	1
3	W	Wprowadzenie do języka Kotlin.	2
4	W	Wstęp do programowania w systemie Android.	2
5	W	Programowanie w systemie Android.	3
6	W	Wprowadzenie do języka Swift.	2
7	W	Wstęp do programowania w systemie iOS.	1
8	W	Programowanie w systemie iOS.	3
9	W	Tworzenie natywnie renderowanych aplikacji na iOS i Androida za pomocą React Native.	2
10	LK	Programowanie w Kotlinie.	2
11	LK	Programowanie w systemie Android.	6
12	LK	Programowanie w Swiftie.	2
13	LK	Programowanie w systemie iOS.	6
14	LK	Programowanie w React Native.	2

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Przygotowanie raportu, projektu, prezentacji, dyskusji	30
2	Egzaminy i zaliczenia w sesji (w bezpośrednim kontakcie z prowadzącym)	4
3	Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury	30
4	Opracowanie wyników	40
5	Konsultacje przedmiotowe (w bezpośrednim kontakcie z prowadzącym)	10

Metody dydaktyczne:

konsultacje, wykład, laboratorium komputerowe, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, dyskusja, egzamin pisemny, sprawozdanie z pracy indywidualnej, sprawdzian wiadomości z wykładów

Kryteria oceny:

1. Na ocenę 5.0 student umie pracować indywidualnie i w grupie oraz przekazywać uzyskane rezultaty pracy w zrozumiały sposób.

2. Na ocenę 5.0 student potrafi implementować natywne aplikacje mobilne na systemy Android i iOS.
3. Na ocenę 5.0 student zna zasady projektowania i tworzenia aplikacji mobilnych. Zna różnice pomiędzy dostępnymi platformami mobilnymi.

Literatura:

obowiązkowa:

1. Charlie Collins, Michael Galpin, Matthias Kaepler, Android w praktyce, 2012, Helion
2. Vandan Nahavandipoor, iOS 5. Programowanie. Receptury, 2012, Helion
3. Nader Dabit, React Native w akcji. Tworzenie aplikacji na iOS i Android w JavaScriptcie, 2020, Wydawnictwo Naukowe PWN

zalecana/fakultatywna:

1. Josh Skeen, David Greenhalgh, Programowanie w języku Kotlin. The Big Nerd Ranch Guide, 2019, Helion
2. Mark A. Lasso (Author), Tom Stachowitz (Contributor), Podstawy języka Swift. Programowanie aplikacji dla platformy iOS, 2016, Helion
3. Frahaan Hussain, Responsive Web Design. Nowoczesne strony WWW na przykładach, 2019, Helion
4. Jakub Nielson, Raluca Budiu, Funkcjonalność aplikacji mobilnych. Nowoczesne standardy UX i UI, 2013, Helion

Programowanie na platformie .NET nazwa przedmiotu
Programming on the .NET platform nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Podstawowa znajomość systemów operacyjnych.
2. Umiejętność programowania w dowolnym języku programowania.

Cele przedmiotu:

1. Poznanie składni języka C# oraz technik programowania na platformie .NET.
2. Nabycie umiejętności programowania interfejsów graficznych z wykorzystaniem WPF oraz webserwisów typu REST.
3. Poznanie podstawowych wzorców projektowych, architektonicznych oraz technik konteneryzacji aplikacji w technologii .NET.
4. Nabycie umiejętności programowania usług typu windows.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	Student zna architekturę platformy .NET. Zna podstawy składni C#. Ma wiedzę dotyczącą obiektowego paradygmatu programowania.	I1_W08
Umiejętności Absolwent potrafi:		
EU1	Student potrafi zaprojektować i zaimplementować aplikację w języku C#.	I1_U23
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Student potrafi pracować w zespole.	I1_K03

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
5	5	Z	18	0	0	18	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	Podstawy składni języka C#. Pętle, instrukcje warunkowe, typy danych, funkcje, struktury, rekordy.	2
2	W	Programowanie obiektowe w C#. Klasy, obiekty, interfejsy, dziedziczenie, polimorfizm statyczny i dynamiczny. Obsługa wyjątków.	2
3	W	Delegaty. Wyrażenia lambda. Zdarzenia.	2
4	W	Typy generyczne. Technologia LINQ.	2
5	W	Programowanie równoległe i asynchroniczne na platformie .NET.	2
6	W	Podstawowe wzorce projektowe i architektoniczne.	2
7	W	Programowanie interfejsów graficznych z wykorzystaniem WPF.	2
8	W	Mapowanie obiektowo relacyjne z wykorzystaniem Entity Framework.	2
9	W	Programowanie usług windowsowych.	2
10	LK	Podstawy składni języka C#.	2
11	LK	Programowanie obiektowe w C#.	3
12	LK	Implementacja podstawowych wzorów projektowych i architektonicznych.	2
13	LK	Programowanie interfejsów graficznych z wykorzystaniem WPF.	2
14	LK	Programowanie webserwisów typu REST.	3
15	LK	Mapowanie obiektowo relacyjne z wykorzystaniem Entity Framework.	2
16	LK	Programowanie usług windowsowych.	2
17	LK	Prezentacje projektów.	2

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury	40
2	Przygotowanie raportu, projektu, prezentacji, dyskusji	34
3	Zaliczenia (w bezpośrednim kontakcie z prowadzącym)	5
4	Konsultacje przedmiotowe (w bezpośrednim kontakcie z prowadzącym)	10

Metody dydaktyczne:

konsultacje, projekt, wykład, laboratorium komputerowe

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, projekt, test, sprawozdanie

Kryteria oceny:

1. Na ocenę 5.0 student potrafi pracować w zespole.
2. Na ocenę 5.0 student potrafi implementować aplikacje na platformie .NET w języku C#.
3. Na ocenę 5.0 student ma wiedzę dotyczącą platformy .NET i języka C#.

Literatura:

obowiązkowa:

1. Mark J. Price, C# 10 i .NET 6 dla programistów aplikacji wieloplatformowych : twórz aplikacje, witryny www oraz serwisy sieciowe za pomocą ASP.NET Core 6, Blazor i EF Core 6 w Visual Studio 2022 i Visual Studio Code, Gliwice, 2023, Helion

zalecana/fakultatywna:

1. Mark Michaelis, Eric Lippert, C# 7.0 : kompletny przewodnik dla praktyków, Miejsowość, 2020, Helion
2. Adam Freeman, ASP.NET Core 3 : zaawansowane programowanie, Gliwice, 2022, Helion

Programowanie robotów nazwa przedmiotu
Robot programming nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Podstawowa wiedza z zakresu elektroniki i programowania.
2. Znajomość języka angielskiego w stopniu umożliwiającym korzystanie z dokumentacji, studiowanie literatury i uczestnictwo w zajęciach.

Cele przedmiotu:

1. Zapoznanie z zasadami projektowania i implementacji algorytmów sterowania robotów z użyciem wybranych języków programowania.
2. Zapoznanie z metodami generowania i programowania trajektorii ruchu robotów mobilnych.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	Zna podstawowe pojęcia związane z robotyką.	I1_W11
Umiejętności Absolwent potrafi:		
EU1	Potrafi wykorzystać języki programowania do programowania i sterowania robotami mobilnymi oraz pozyskiwać informacje z dokumentacji i wykorzystywać je do realizacji zadań inżynierskich.	I1_U23
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Potrafi współpracować w grupie i realizować projekt zespołowy.	I1_K03

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
7	4	E	18	0	0	18	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	Podstawowe pojęcia związane z robotyką. Akwizycja danych wykorzystywane w robotach. Układy regulacji i sterowanie wykorzystywane w robotach. Rodzaje efektorów wykorzystywanych w robotach kołowych oraz gąsienicowych. Sensory wykorzystywane w robotach mobilnych. Kinematyka różnych rodzajów robotów mobilnych. Języki programowania wykorzystywane w programowaniu i sterowaniu robotami mobilnymi. Symulatory robotów mobilnych i ich wykorzystanie w testowaniu kinematyki robotów mobilnych. Zastosowanie robotów w przemyśle ciężkim. Systemy lokalizacji i pozycjonowania robotów mobilnych.	18

2	LK	Badanie symulacyjne efektorów wykorzystywanych w robotach kołowych. Programowanie robota kołowego mBot2. Badanie sensorów wykorzystanych w robotach kołowych mBot2. Układ akwizycji danych i wykorzystanie pozyskanych danych w sterowaniu robotem mobilnych. Chwytnik i jego wykorzystanie w robotach kołowych. Programowanie robota humanoidalnego BIOLOID. Koordynacja ruch robotów mobilnych mBot2. Lokalizacja robota kołowego w przestrzeni kartezjańskiej.	14
3	LK	Realizacja w grupie projektu z programowania robotów.	4

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Przygotowanie się do zajęć i dyskusji, w tym studiowanie zalecanej literatury, opracowanie wyników przeprowadzonych badań, przygotowanie raportu, projektu, prezentacji itp.	39
2	Konsultacje przedmiotowe (w bezpośrednim kontakcie z prowadzącym)	20
3	Egzaminy i zaliczenia (w bezpośrednim kontakcie z prowadzącym)	5

Metody dydaktyczne:

dyskusja, konsultacje, praca w grupach, projekt, wykład, laboratorium komputerowe, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, dyskusja, prezentacja, projekt, sprawozdanie

Kryteria oceny:

1. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji wiedzy.
2. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji umiejętności.
3. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji kompetencji społecznych.

Literatura:

obowiązkowa:

1. Mordechai Ben-Ari, Francesco Mondada. Elementy robotyki dla początkujących, Warszawa, 2022, Helion
2. Cameron Hughes, Tracey Hughes. Programowanie robotów. Sterowanie pracą robotów autonomicznych, Warszawa, 2017, Helion

zalecana/fakultatywna:

1. David Cook. Budowa robotów dla początkujących. Wydanie III (ebook)

Programowanie usług sieciowych nazwa przedmiotu
Programming of Network Services nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Konfiguracja i utrzymanie sieci komputerowych.
2. Administrowanie systemami Linux i Windows.
3. Programowanie w języku programowania wspierającym komunikację na poziomie TCP/IP i szyfrowanie SSL (np. Python, C/C++, Java).
4. Programowanie na poziomie systemu operacyjnego.
5. Znajomość języka angielskiego w stopniu umożliwiającym studiowanie literatury.

Cele przedmiotu:

1. Poznanie konstrukcji protokołów i aplikacji sieciowych ze szczególnym uwzględnieniem 3. oraz 4. warstwy ISO/OSI.
2. Nauka projektowania protokołów i aplikacji sieciowych z naciskiem na obsługę błędów transmisji, stan połączenia i bezpieczeństwo komunikacji.
3. Praktyka implementacji protokołów i aplikacji sieciowych z wykorzystaniem niskopoziomowych gniazd Unix.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	podstawowe metody, techniki i narzędzia stosowane przy rozwiązywaniu prostych zadań informatycznych z zakresu budowy systemów komputerowych, sieci komputerowych i technologii sieciowych oraz systemów wbudowanych.	I1_W12
Umiejętności Absolwent potrafi:		
EU1	sformułować specyfikację prostych systemów informatycznych w odniesieniu do sprzętu, oprogramowania systemowego i cech funkcjonalnych aplikacji.	I1_U21
EU2	zaprojektować oraz zrealizować system informatyczny, używając właściwych metod, technik i narzędzi.	I1_U23
Kompetencje społeczne Absolwent jest gotów do:		
EK1	pracy indywidualnie i w zespole informatyków, w tym także potrafi zarządzać swoim czasem oraz podejmować zobowiązania i dotrzymywać terminów.	I1_K03

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
7	4	Z	18	0	0	0	18	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	Wprowadzenie do tematyki, podstawowe wiadomości wstępne. Gniazda sieciowe, programowanie na warstwie transportowej, protokół TCP. Programowanie na warstwie transportowej, protokół UDP.	3

2	W	Programowanie w warstwie IP - opcje IP i gniazda surowe. Wprowadzenie do kryptografii, podstawy teoretyczne. Bezpieczna komunikacja oparta o techniki programistyczne.	3
3	W	Projektowanie protokołów sieciowych. Projektowanie aplikacji sieciowych. Architektura klient-serwer, struktura adresowa.	3
4	W	Protokoły i ich wykorzystanie w bezpiecznej komunikacji sieciowej. Programowanie komunikacji lokalnej. Programowe przekształcanie nazw domenowych, protokół DNS.	3
5	W	Firewall systemu operacyjnego.	3
6	W	Demonizacja procesów, logowanie przez system operacyjny. Programowanie na protokole QUIC.	3
7	P	Gniazda TCP, UDP i gniazda surowe, komunikacja IPv4 i IPv6.	3
8	P	Bezpieczna komunikacja w OpenSSL.	3
9	P	Zaprojektowanie autorskiego protokołu sieciowego uwzględniającego bezpieczną komunikację i obsługę błędów połączenia.	3
10	P	Zaprojektowanie aplikacji klient-serwer lub aplikacji P2P wykorzystującej protokół z P3.	4
11	P	Implementacja aplikacji z P4.	5

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Konsultacje przedmiotowe (w bezpośrednim kontakcie z prowadzącym)	18
2	Realizacja projektów	44
3	Zaliczenie (w bezpośrednim kontakcie z prowadzącym)	2

Metody dydaktyczne:

projekt, wykład, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

odpowiedź ustna, projekt

Kryteria oceny:

- Wymaganie na ocenę 5,0: student wykonał projekt w terminie. Projekt spełniał wszystkie wymagane założenia, był dobrze zaimplementowany, przetestowany i udokumentowany.
Wymaganie na ocenę 4,5: student opanował mniej niż 91% ww. zagadnień, ale więcej niż 80%.
Wymaganie na ocenę 4,0: student opanował mniej niż 81% ww. zagadnień, ale więcej niż 70%.
Wymaganie na ocenę 3,5: student opanował mniej niż 71% ww. zagadnień, ale więcej niż 60%.
Wymaganie na ocenę 3,0: student opanował mniej niż 61% ww. zagadnień, ale więcej niż 50%.
- Wymaganie na ocenę 5,0: student poprawnie rozumiał polecenie zadania, opracował specyfikację i dokonał poprawnego doboru technologii informacyjnych do ich realizacji.
Wymaganie na ocenę 4,5: student opanował mniej niż 91% ww. zagadnień, ale więcej niż 80%.
Wymaganie na ocenę 4,0: student opanował mniej niż 81% ww. zagadnień, ale więcej niż 70%.
Wymaganie na ocenę 3,5: student opanował mniej niż 71% ww. zagadnień, ale więcej niż 60%.
Wymaganie na ocenę 3,0: student opanował mniej niż 61% ww. zagadnień, ale więcej niż 50%.

3. Wymaganie na ocenę 5,0: student poprawnie zrealizował projekty od strony technicznej.
Wymaganie na ocenę 4,5: student opanował mniej niż 91% ww. zagadnień, ale więcej niż 80%.
Wymaganie na ocenę 4,0: student opanował mniej niż 81% ww. zagadnień, ale więcej niż 70%.
Wymaganie na ocenę 3,5: student opanował mniej niż 71% ww. zagadnień, ale więcej niż 60%.
Wymaganie na ocenę 3,0: student opanował mniej niż 61% ww. zagadnień, ale więcej niż 50%.
4. Wymaganie na ocenę 5,0: student zna stos ISO/OSI oraz obsługę komunikacji na poziomie bibliotek systemu operacyjnego począwszy od warstwy trzeciej tego stosu. Umie odpowiednio zaprojektować i zaimplementować aplikację sieciową. Potrafi zaprojektować protokół komunikacji, za pomocą którego jego aplikacje będą się komunikować ze sobą.
Wymaganie na ocenę 4,5: student opanował mniej niż 91% ww. zagadnień, ale więcej niż 80%.
Wymaganie na ocenę 4,0: student opanował mniej niż 81% ww. zagadnień, ale więcej niż 70%.
Wymaganie na ocenę 3,5: student opanował mniej niż 71% ww. zagadnień, ale więcej niż 60%.
Wymaganie na ocenę 3,0: student opanował mniej niż 61% ww. zagadnień, ale więcej niż 50%.

Literatura:

obowiązkowa:

1. W. Richard Stevens, Programowanie Usług Sieciowych API: gniazda i XTI, Warszawa, 2002, WNT
2. Internet Society (ISOC), Dokumenty RFC, www.ietf.org, 2024, IETF

zalecana/fakultatywna:

1. Rochkind Marc J., Programowanie w systemie UNIX dla zaawansowanych, Warszawa 2010, WNT
2. Greg Perry, Dean Miller, Język C. Programowanie dla początkujących. Wydanie III, Gliwice, 2016, Helion
3. Robert Love, Linux. Programowanie systemowe. Wydanie II, Gliwice, 2014, Helion

Programowanie w języku Java nazwa przedmiotu
Programming of Java nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Umiejętność budowania algorytmów.
2. Znajomość jednego języka programowania (strukturalnego lub obiektowego).

Cele przedmiotu:

1. Poznanie elementów leksykalnych, składniowych i podstaw semantyki języka Java Standard Edition, jej wieloparadygmatowości i sposobów rozszerzania języka Java za pomocą refleksji.
2. Opanowanie podstawowych narzędzi wykorzystywanych w środowisku wytwórczym typowym dla języka Java Standard Edition: IntelliJ IDEA, Maven, Gradle, Git, JUnit5.
3. Opanowanie wzorców projektowych ogólnego przeznaczenia (GoF) najbardziej przydatnych w programowaniu w Java Standard Edition oraz dobrych praktyk w ich stosowaniu. Wzorce strukturalne: Dekorator, wzorce czynnościowe: Iterator, wzorce kreacyjne: Metoda Wytwórcza, Budowniczy. Poznanie podstaw języka modelowania UML w zakresie potrzebnym do modelowania wzorców projektowych.
4. Poznanie mechanizmu osłabiania zależności i możliwych do osiągnięcia granic w ramach Java Standard Edition a także sposobów ich przekraczania.
5. Poznanie biblioteki kontenerów Java 17 Standard Edition.
6. Poznanie biblioteki wejścia/wyjścia oraz strumieni Java 17 Standard Edition.
7. Opanowanie trwałości w Java z wykorzystaniem JDBC i JPA.
8. Poznanie podstaw i zasad korzystania z biblioteki Swing dla GUI aplikacji desktopowych.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	Instrukcje i struktury danych w Javie. Zasady obiektowości. Pojęcia: klasa, obiekt, enkapsulacja, abstrakcja, dziedziczenie, polimorfizm, klasa abstrakcyjna, interfejs, klasa wewnętrzna, anonimowa klasa wewnętrzna, wyrażenie lambda, wyjątki, typy generyczne, adnotacje, refleksja.	I1_W06
EW2	Znajomość standardowych bibliotek języka Java, w tym bibliotek wejścia/wyjścia, kontenerów, JDBC, JPA, Swing.	I1_W12

Umiejętności Absolwent potrafi:		
EU1	Implementowanie wzorców projektowych dla wybranych przez studentów problemów z wykorzystaniem wiedzy o języku Java, o wybranych wzorcach projektowych i o narzędziach deweloperskich. Umiejętność dokonywania wyboru właściwych wzorców projektowych oraz składania wzorców projektowych.	I1_U09
EU2	Praktyczne wykorzystanie wiedzy o języku, wzorcach projektowych, bibliotekach standardowych Java oraz o dobrych praktykach w laboratoryjnych projektach zaliczeniowych.	I1_U09
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Praca w zespole, zrozumienie znaczenia procesów twórczych oprogramowania, w tym z dyscyplinami zarządzania konfiguracją i testowania, wspólny projekt i implementacja oprogramowania. Zebranie doświadczeń w ramach projektów zespołowych.	I1_K03

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
4	4	E	18	0	0	18	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	Wprowadzenie do języka Java oraz do przedmiotu. Prezentacja zalecanej literatury. Prezentacja przykładowego programu w Java ilustrującego polimorfizm.	2
2	W	Omówienie podstawowych elementów leksykalnych języka Java.	1
3	W	Typy generyczne oraz biblioteka kontenerów w Java 17.	2
4	W	Omówienie bibliotek wejścia/wyjścia oraz strumieni w Java 17.	2
5	W	Omówienie potrzeby stosowania, mechanizmu wyjątków oraz dobrych praktyk w ich stosowaniu.	2
6	W	Wstęp do zagadnień trwałości w Java. Omówienie JDBC ilustrowane licznymi przykładami. Dobre praktyki związane z wykorzystywaniem JDBC.	2
7	W	Omówienie własności oraz sposobów konstruowania GUI z wykorzystaniem biblioteki Swing.	2

8	W	Kontynuacja zagadnień związanych z trwałością. Omówienie standardy JPA i frameworku Hibernate.	1
9	W	Wprowadzenie do wielowątkowości w Java 17.	2
10	W	Wprowadzenie do zasad funkcjonowania JVM.	1
11	W	Charakterystyka technologii związanych z JVM, w tym alternatywnych języków programowania.	1
12	LK	Prezentacja sposobu wykorzystywania środowiska deweloperskiego Java wybranego przez prowadzącego zajęcia (domyślnie IntelliJ IDEA) na przykładzie z wykładu ilustrującym polimorfizm. Samodzielna implementacja tego programu przez studentów w wersji zmodyfikowanej.	2
13	LK	Omówienie narzędzia Maven zastosowanie go przez studentów do zrealizowania przykładu z polimorfizmem	1
14	LK	Omówienie narzędzia JUnit5 i wykorzystanie go przez studentów do zrealizowania przykładu z polimorfizmem.	1
15	LK	Omówienie narzędzia Git i wykorzystanie go przez studentów do zrealizowania przykładu z polimorfizmem	1
16	LK	Samodzielna implementacja programu wykorzystującego narzędzia Git, Maven i JUnit5 pozwalającego na zastosowanie właściwych wzorców projektowych do wybranego przez studentów problemu. Tylko warstwa logiki biznesowej. Praca zespołowa.	4
17	LK	Realizacja większego projektu pozwalającego wykorzystać wszystkie znane narzędzia, konstrukcje językowe, biblioteki i dobre praktyki poznawane stopniowo na wykładach. Wraz z warstwą danych i warstwą prezentacji. Praca zespołowa.	9

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Egzaminy i zaliczenia (w bezpośrednim kontakcie z prowadzącym)	5
2	Samodzielne zapoznawanie się z literaturą związaną z przedmiotem.	25
3	Samodzielna implementacja programów w Javie pozwalająca na usystematyzowanie wiedzy zdobywanej na wykładzie.	20
4	Konsultacje przedmiotowe (w bezpośrednim kontakcie z prowadzącym)	14

Metody dydaktyczne:

ćwiczenia laboratoryjne, wykład

Sposoby weryfikacji i oceny efektów uczenia się:

egzamin pisemny, kolokwium, odpowiedź ustna, projekt, sprawozdanie z pracy indywidualnej, sprawozdanie z pracy zespołowej

Kryteria oceny:

1. Na ocenę 5.0 student musi znać składnię języka Java w zakresie omawianym na wykładzie.

2. Na ocenę 5.0 student musi umieć wykorzystać biblioteki standardowej języka Java, kontenerów oraz strumieni.
3. Na ocenę 5.0 student musi umieć tworzyć programy w języku Java.
4. Na ocenę 5.0 student musi znać wzorce projektowe i umieć je zastosować w tworzonych programach.
5. Na ocenę 5.0 student musi umieć zaprojektować i zaimplementować złożony program komputerowy. Na ocenę 5.0 student musi umieć współpracować podczas tworzenia programu.

Literatura:

obowiązkowa:

1. Horstman C.S., Cornell G "Java: podstawy"
2. Horstman C.S., Cornell G "Java: techniki zaawansowane"
3. Gamma E., Helm R., Johnson R., Vlissides J.M. "Wzorce projektowe. Elementy oprogramowania obiektowego wielokrotnego użytku"

zalecana/fakultatywna:

1. Smart J.F. "Java. Praktyczne narzędzia"

Programowanie w języku Python nazwa przedmiotu
Programming in Python nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Znajomość podstaw obsługi komputera.

Cele przedmiotu:

1. Nauka programowania w języku Python.
2. Nauka użycia podstawowych bibliotek dostępnych w języku Python.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	Student zna i rozumie podstawowe konstrukcje języka Python.	I1_W06
EW1	Student zna i rozumie podstawowe konstrukcje języka Python.	I1_W08
EW1	Student zna i rozumie podstawowe konstrukcje języka Python.	I1_W09
EW1	Student zna i rozumie podstawowe konstrukcje języka Python.	I1_W10
EW1	Student zna i rozumie podstawowe konstrukcje języka Python.	I1_W11
EW2	Student zna i rozumie użycie podstawowych bibliotek języka Python.	I1_W06
EW2	Student zna i rozumie użycie podstawowych bibliotek języka Python.	I1_W09
EW2	Student zna i rozumie użycie podstawowych bibliotek języka Python.	I1_W10
EW2	Student zna i rozumie użycie podstawowych bibliotek języka Python.	I1_W11
EW2	Student zna i rozumie użycie podstawowych bibliotek języka Python.	I1_W08
Umiejętności Absolwent potrafi:		
EU1	Student potrafi wykorzystać podstawowe konstrukcje programistyczne w języku Python.	I1_U02
EU1	Student potrafi wykorzystać podstawowe konstrukcje programistyczne w języku Python.	I1_U03
EU1	Student potrafi wykorzystać podstawowe konstrukcje programistyczne w języku Python.	I1_U07b
EU1	Student potrafi wykorzystać podstawowe konstrukcje programistyczne w języku Python.	I1_U08

EU1	Student potrafi wykorzystać podstawowe konstrukcje programistyczne w języku Python.	I1_U10
EU1	Student potrafi wykorzystać podstawowe konstrukcje programistyczne w języku Python.	I1_U22
EU1	Student potrafi wykorzystać podstawowe konstrukcje programistyczne w języku Python.	I1_U23
EU2	Student potrafi wykorzystać podstawowe biblioteki języka Python.	I1_U02
EU2	Student potrafi wykorzystać podstawowe biblioteki języka Python.	I1_U03
EU2	Student potrafi wykorzystać podstawowe biblioteki języka Python.	I1_U07b
EU2	Student potrafi wykorzystać podstawowe biblioteki języka Python.	I1_U08
EU2	Student potrafi wykorzystać podstawowe biblioteki języka Python.	I1_U10
EU2	Student potrafi wykorzystać podstawowe biblioteki języka Python.	I1_U22
EU2	Student potrafi wykorzystać podstawowe biblioteki języka Python.	I1_U23
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Student zna i docenia rolę języka Python we współczesnym świecie IT.	I1_K01
EK1	Student zna i docenia rolę języka Python we współczesnym świecie IT.	I1_K02
EK1	Student zna i docenia rolę języka Python we współczesnym świecie IT.	I1_K03
EK1	Student zna i docenia rolę języka Python we współczesnym świecie IT.	I1_K04
EK1	Student zna i docenia rolę języka Python we współczesnym świecie IT.	I1_K05
EK1	Student zna i docenia rolę języka Python we współczesnym świecie IT.	I1_K06

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
2	4	Z	9	0	0	18	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	Wprowadzenie do języka Python.	5
2	W	Biblioteka numpy i wybrane obliczenia algebraiczne.	1
3	W	Biblioteka matplotlib i wybrane typy wykresów.	1
4	W	Biblioteka scipy i wybrane obliczenia naukowe.	1
5	W	Biblioteka pandas i podstawowe struktury do analizy danych.	1
6	LK	Wprowadzenie do języka Python.	10
7	LK	Biblioteka numpy i wybrane obliczenia algebraiczne.	2
8	LK	Biblioteka matplotlib i wybrane typy wykresów.	2
9	LK	Biblioteka scipy i wybrane obliczenia naukowe.	2
10	LK	Biblioteka pandas i podstawowe struktury do analizy danych.	1
11	LK	Prezentacja projektu.	1

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Konsultacje przedmiotowe (w bezpośrednim kontakcie z prowadzącym)	10
2	Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury	25
3	Opracowanie wyników (w bezpośrednim kontakcie z prowadzącym)	18
4	Przygotowanie raportu, projektu, prezentacji, dyskusji	18
5	Zaliczenie (w bezpośrednim kontakcie z prowadzącym)	2

Metody dydaktyczne:

ćwiczenia laboratoryjne, dyskusja, konsultacje, praca w grupach, praca z tekstem/materiałem audio i wideo, wykład, laboratorium komputerowe, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, dyskusja, kolokwium, odpowiedź ustna, prezentacja, projekt, referat, rozwiązanie zadania problemowego, sprawozdanie z pracy indywidualnej, sprawozdanie z pracy zespołowej, test, zaliczenie pisemne, zaliczenie ustne, sprawdzian wiadomości z wykładów, kolokwium z zadań, sprawozdanie

Kryteria oceny:

1. Na ocenę 5.0 Student zna i rozumie podstawowe konstrukcje języka Python na poziomie 90% wykładanego materiału.
2. Na ocenę 5.0 Student zna i rozumie użycie podstawowych bibliotek języka Python na poziomie więcej niż 90% wykładanego materiału.
3. Na ocenę 5.0 Student potrafi wykorzystać podstawowe konstrukcje programistyczne w języku Python na poziomie większym niż 90% materiału.
4. Na ocenę 5.0 Student potrafi wykorzystać podstawowe biblioteki języka Python na poziomie powyżej 90% materiału.
5. Na ocenę 5.0 Student zna i docenia rolę języka Python we współczesnym świecie IT.

Literatura:

obowiązkowa:

1. Mark Lutz, 'Python: wprowadzenie', 2022, Helion
2. Hans Fangohr, 'Computational Science and Engineering in Python', 2018, Zenodo
3. Redakcja zbiorowa, 'Scipy Lecture Notes', 2022,

zalecana/fakultatywna:

1. James O. Knowlton, 'Python : projekty do wykorzystania', 2010, Helion
2. David M. Beazley, Brian K. Jones, 'Python : receptury', 2014, Helion

Projekt zespołowy II nazwa przedmiotu
Team project II nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Umiejętność pracy w zespole, kompetencje programistyczne, znajomość podstawowych zasad inżynierii oprogramowania.
2. Znajomość języka angielskiego w stopniu umożliwiającym korzystanie z dokumentacji, studiowanie literatury i uczestnictwo w zajęciach.

Cele przedmiotu:

1. Nabycie i/lub doskonalenie umiejętności przygotowania praktycznego projektu biznesowego o charakterze wdrożeniowym, zgodnie z wymaganiami i oczekiwaniami rynku IT.
2. Przygotowanie dokumentacji, wykorzystanie odpowiednich metodologii wytwarzania oprogramowania i analiz rynku oraz prezentacja projektu na forum grupy.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	Student posiada wiedzę o procesach biznesowych w organizacji i jej otoczeniu, potrafi dokonać ich analizy i oceny. Posiada wiedzę o narzędziach i metodach wykorzystywanych w różnych obszarach zarządzania. Rozumie istotę przedsiębiorczości oraz zasady funkcjonowania rynku w sektorze IT.	I1_W10
Umiejętności Absolwent potrafi:		
EU1	Student potrafi zdefiniować problem biznesowy, przygotować propozycję rozwiązania problemu i dobrać odpowiednie metody i narzędzia do jego realizacji. Student potrafi formułować wnioski i rekomendować działania o walorach implementacyjnych/wdrożeniowych.	I1_U05
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Student potrafi współpracować w grupie realizując wspólny projekt. Cechuje go twórcze myślenie, kreatywność i innowacyjność w realizowanych zadaniach.	I1_K03

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
7	3	Z	0	0	0	0	35	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	P	Przedsiębiorczość na rynku pracy. Twórcze metody znajdowania rozwiązań. Innowacyjne przedsięwzięcia (nisza rynkowa, identyfikacja potrzeb, ocena własnych możliwości, analiza ryzyka). Praca w zespole. Autoprezentacja.	9
2	P	Kontakt z klientem. Specyfikacja wymagań, scenariusze zadań. Strategie i metodyki zarządzanie projektem. Planowanie budżetu. Szacowanie czasu trwania projektu/zadań, delegowanie zadań, zarządzanie zespołem. Efektywne sposoby testowania i wdrażania oprogramowania. Dobre praktyki. Zasady rozliczenia poszczególnych zadań i całego projektu. Spotkania z przedstawicielami instytucji współpracujących z wydziałem. Ustalenie celu projektu biznesowego, określenie zadań, powołanie zespołów projektowych. Ustalenie harmonogramu spotkań i zasad mentoringu. Realizacja projektu biznesowego pod kierunkiem mentora z firmy i opiekuna przedmiotu. Spotkania projektowe.	19
3	P	Końcowa prezentacja osiągniętych rezultatów. Końcowa wizyta studyjna w firmie, prezentacja projektów, dyskusja, wzajemna wymiana wiedzy i doświadczenia.	8

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Przygotowanie się do zajęć i dyskusji, w tym studiowanie zalecanej literatury, opracowanie wyników przeprowadzonych badań, przygotowanie raportu, projektu, prezentacji itp. Realizacja projektu na wybrany temat.	33
2	Konsultacje przedmiotowe i zaliczenie (w bezpośrednim kontakcie z prowadzącym)	7

Metody dydaktyczne:

dyskusja, konsultacje, praca w grupach, projekt, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, dyskusja, prezentacja, projekt

Kryteria oceny:

1. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji kompetencji społecznych.
2. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji umiejętności.
3. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji wiedzy.

Literatura:

obowiązkowa:

1. Piotr Wróblewski. Zwinnie do przodu. Poradnik kierownika projektów informatycznych, 2020, Helion
2. Mariot Tsitoara. Git i GitHub. Kontrola wersji, zarządzanie projektami i zasady pracy zespołowej, 2022, Helion

zalecana/fakultatywna:

1. Adam Koszlajda. Od pomysłu do przemysłu. Projekty IT w praktyce (ebook)

Projekt zespołowy I nazwa przedmiotu
Team project I nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Znajomość języków programowania i inżynierii oprogramowania na podstawowym poziomie.
2. Znajomość języka angielskiego w stopniu umożliwiającym korzystanie z dokumentacji, studiowanie literatury i uczestnictwo w zajęciach.

Cele przedmiotu:

1. Zdobyć umiejętności wyszukiwania/pozyskiwania informacji o bieżących zagadnieniach IT
2. Doskonalenie kompetencji w zakresie realizacji indywidualnych i zespołowych projektów dotyczących bieżących zagadnień IT
3. Nabycie umiejętności przygotowania raportu i zwięzłej prezentacji uzyskanych wyników

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	Student umie pozyskiwać informacje z literatury, baz danych oraz innych źródeł, integrować je, dokonywać ich interpretacji oraz wyciągać wnioski i formułować opinie.	I1_W10
Umiejętności Absolwent potrafi:		
EU1	Student potrafi pracować w zespole przy realizacji projektów. Potrafi formułować zadania/problemy z dziedziny informatyki.	I1_U05
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Student potrafi kierować zespołem, delegować zadania i zarządzać czasem.	I1_K03

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
6	3	Z	0	0	0	0	35	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	P	Zaproponowane tematu projektu, uformowanie się grupy projektowej, określenie celów prac. Zdefiniowanie wymagań i wyzwań technologicznych.	9
2	P	Analiza dostępnych rozwiązań Realizacja projektu. Zespół projektowy realizuje założone cele i regularnie przedstawia wyniki prac. Przedstawienie finalnych wyników oraz opracowanie raportu końcowego.	18
3	P	Praca w grupie, delegowanie zadań i zarządzanie projektem. Przedstawienie finalnych wyników oraz opracowanie raportu końcowego.	8

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Przygotowanie się do zajęć i dyskusji, w tym studiowanie zalecanej literatury, opracowanie wyników przeprowadzonych badań, przygotowanie raportu, projektu, prezentacji itp. Realizacja projektu zgodnie z wybranym tematem.	18
2	Konsultacje przedmiotowe i zaliczenie (w bezpośrednim kontakcie z prowadzącym)	22

Metody dydaktyczne:

dyskusja, konsultacje, praca w grupach, projekt, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, dyskusja, prezentacja, projekt

Kryteria oceny:

1. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji kompetencji społecznych.
2. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji umiejętności.
3. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji wiedzy.

Literatura:

obowiązkowa:

1. Klaudyna Maciąg. Pisz. Publikuj. Działaj. Jak tworzyć skuteczne treści w internecie. MT Biznes.2023.
2. Tim Clark, Alexander Osterwalder, Yves Pigneur. Model biznesowy. TY (ebook)

zalecana/fakultatywna:

1. Adam Koszlajda. Od pomysłu do przemysłu. Projekty IT w praktyce (ebook)

Projektowanie sieci komputerowych nazwa przedmiotu
Computer network design nazwa przedmiotu w języku angielskim
polski język wykładowy

przedmioty kierunkowe kategoria przedmiotu/grupa zajęć
--

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Podstawy sieci komputerowych

Cele przedmiotu:

1. Umiejętność projektowania wymaganych elementów lokalnych sieci komputerowych

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	Teoretyczna wiedza na temat poszczególnych elementów sieci komputerowych i etapów projektowania	I1_W05
Umiejętności Absolwent potrafi:		
EU1	Umiejętność przeprowadzenia procesu projektowania poszczególnych elementów sieci komputerowej	I1_U01b
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Wykonanie indywidualnego lub w zespole projektu sieci komputerowej wg określonego harmonogramu	I1_K03

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
7	4	Z	18	0	0	18	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	Składowe elementy sieci komputerowych, projekt hierarchiczny, projekt fizyczny, urządzenia sieciowe, adresowanie i konfiguracja urządzeń, technologie, protokoły, przepustowość, sprawność sieci, metody projektowania.	18
2	LK	Wykonanie projektu wg określonych założeń i ustalonego harmonogramu, konsultowanie i zdawanie poszczególnych etapów projektowych, prezentowanie projektu.	18

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Konsultacje przedmiotowe (w bezpośrednim kontakcie z prowadzącym)	15
2	zaliczanie wykonanego projektu (w bezpośrednim kontakcie z prowadzącym)	15
3	Praca własna - przygotowanie do zajęć i wykonanie opisu projektu	54

Metody dydaktyczne:

dyskusja, konsultacje, praca w grupach, projekt, wykład, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, dyskusja, projekt, zaliczenie ustne

Kryteria oceny:

1. Na ocenę 5,0 student nabył wiedzę treści wykładowych w stopniu co najmniej 90%
2. Na ocenę 5,0 - student zaliczy etapy wykonania projektu terminowo a całość projektu zostanie oceniona na 5,0
3. Na ocenę 5,0 student wykonuje etapy projektu i oddaje je bez opóźnienia. W pracy zespołowej swoją część pracy wykonuje na czas.

Literatura:

obowiązkowa:

1. Paweł Zaręba - Projekty i rozwiązania sieciowe w praktyce, Helion, 2022
2. Russ White, Ethan Banks - Sieci komputerowe. Najczęstsze problemy i ich rozwiązania, Helion, 2019

zalecana/fakultatywna:

1. J. Scott Haugdahl - Diagnostowanie i utrzymanie sieci. Księga eksperta, Helion, 2000

Projektowanie systemów informatycznych dla osób z niepełnosprawnościami nazwa przedmiotu
Project IT systems for people with disabilities nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Podstawy programowania w stopniu umożliwiającym stworzenie prostej aplikacji okienkowej i aplikacji mobilnej.
2. Podstawy w tworzeniu stron internetowych.
3. Obsługa systemu operacyjnego Windows i Android.
4. Znajomość języka angielskiego w stopniu umożliwiającym studiowanie literatury.

Cele przedmiotu:

1. Poznanie sposobów obsługi komputera przez osoby z różnymi rodzajami niepełnosprawności.
2. Poznanie aktualnych wytycznych jak tworzyć systemy informatyczne dostępne dla osób z niepełnosprawnościami.
3. Praktyka implementacji aplikacji na komputery i urządzenia mobilne oraz stron internetowych spełniających wytyczne dostępności.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	Zna różne rodzaje niepełnosprawności i aktualne wytyczne jak tworzyć systemy informatyczne dostępne dla osób z ograniczeniami.	I1_W13
EW2	Wie w jaki sposób projektować sposób obsługi swoich aplikacji i stron internetowych aby były dostępne dla osób z ograniczeniami.	I1_W12
Umiejętności Absolwent potrafi:		
EU1	Potrafi zaimplementować aplikację okienkową i mobilną oraz zaprojektować stronę internetową spełniającą wytyczne dostępności dla osób z niepełnosprawnościami.	I1_U23
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Zna potrzeby osób z niepełnosprawnościami w kwestii obsługi komputera i ma świadomość o potrzebie projektowania systemów informatycznych w taki sposób aby było dostępne.	I1_K06

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
5	3	Z	18	0	0	18	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	Niepełnosprawności wzroku, słuchu oraz niepełnosprawności ruchowe osób korzystających z komputerów. Funkcje dostępności (FD) wbudowane w systemy operacyjne na komputery i urządzenia mobilne. Technologie asystujące (TA) - urządzenia i oprogramowanie firm trzecich wspomagające obsługę komputera przez osoby z ograniczeniami.	3

2	W	Web Content Accessibility Guidelines (WCAG) - wytyczne World Wide Web Consortium (W3C) dotyczące dostępności treści internetowych.	3
3	W	Przedstawienie informacji i komponentów interfejsu użytkownika w sposób dostrzegalny dla zmysłów z ograniczeniami wzrokowymi, obsługiwany przez FD i TA	3
4	W	Projektowanie komponentów interfejsu użytkownika w sposób umożliwiający interakcję z nimi za pomocą FD i TA.	3
5	W	Projektowanie całych aplikacji i serwisów internetowych w sposób przejrzysty i zrozumiały, redukując konieczność niepotrzebnych interakcji, które dla zdrowych osób nie stanowią problemu, a dla osób z niepełnosprawnościami niepotrzebnie utrudniają korzystanie.	3
6	W	Sprawdzanie aplikacji i serwisów internetowych pod kątem dostępności. Wymagania prawne zgodnie z którymi strony internetowe i aplikacje mobilne podmiotów publicznych muszą być dostępne cyfrowo.	3
7	LK	Ćwiczenia uświadamiające: próba obsługi komputera z symulowanymi ograniczeniami (zamkniętymi oczami, zatkanymi uszami, bez myszki lub tylko myszką, możliwość naciśnięcia tylko jednego klawisza w jednym czasie).	3
8	LK	Funkcje dostępności wbudowane w system Windows i system Android. Obsługa komputera programem eViacam. Powtórka podstaw HTML i tworzenia stron internetowych.	3
9	LK	Accessible Rich Internet Applications (ARIA), ćwiczenie z obsługi strony internetowej z symulowanymi ograniczeniami. Układ semantyczny strony HTML i projektowanie dostępnej nawigacji w serwisie internetowym. Formatowanie tekstu i układu strony w sposób przejrzysty i intuicyjny.	3
10	LK	Stosowanie treści alternatywnych dla multimediów (tekst do grafik, napisy do filmów). Projektowanie dostępnych formularzy na stronach internetowych. Powtórka z C# i tworzenia aplikacji okienkowych w Visual C#	3
11	LK	Implementacja treści audio-wizualnych kompatybilnych z FD i TA w aplikacjach okienkowych. Implementacja formularzy kompatybilnych z FD i TA w aplikacjach okienkowych.	3
12	LK	Powtórka z Java/Kotlin i tworzenia aplikacji okienkowych na Android. Implementacja treści audio-wizualnych kompatybilnych z FD i TA w aplikacjach mobilnych. Implementacja formularzy kompatybilnych z FD i TA w aplikacjach mobilnych.	3

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Konsultacje przedmiotowe (w bezpośrednim kontakcie z prowadzącym)	18
2	Przygotowanie do zajęć	34
3	Zaliczenie (w bezpośrednim kontakcie z prowadzącym)	2

Metody dydaktyczne:

konsultacje, wykład, laboratorium komputerowe, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, kolokwium, sprawozdanie

Kryteria oceny:

1. Na ocenę 5,0 Student opanował więcej niż 90% materiału. Na ocenę 4,5 Student opanował więcej niż 80% materiału. Na ocenę 4,0 Student opanował więcej niż 70% materiału. Na ocenę 3,5 Student opanował więcej niż 60% materiału. Na ocenę 3,0 Student opanował więcej niż 50% materiału.
2. Na ocenę 5,0 Student opanował więcej niż 90% materiału. Na ocenę 4,5 Student opanował więcej niż 80% materiału. Na ocenę 4,0 Student opanował więcej niż 70% materiału. Na ocenę 3,5 Student opanował więcej niż 60% materiału. Na ocenę 3,0 Student opanował więcej niż 50% materiału.
3. Na ocenę 5,0 Student opanował więcej niż 90% materiału. Na ocenę 4,5 Student opanował więcej niż 80% materiału. Na ocenę 4,0 Student opanował więcej niż 70% materiału. Na ocenę 3,5 Student opanował więcej niż 60% materiału. Na ocenę 3,0 Student opanował więcej niż 50% materiału.
4. Na ocenę 5,0 Student opanował więcej niż 90% materiału. Na ocenę 4,5 Student opanował więcej niż 80% materiału. Na ocenę 4,0 Student opanował więcej niż 70% materiału. Na ocenę 3,5 Student opanował więcej niż 60% materiału. Na ocenę 3,0 Student opanował więcej niż 50% materiału.

Literatura:

obowiązkowa:

1. W3C, Web Content Accessibility Guidelines (WCAG) 2.1, Cambridge, Massachusetts, United States, 2021, <https://www.w3.org/Translations/WCAG21-pl/>
2. Maksymilian Gutowski, HTML5. Nieoficjalny podręcznik. Wydanie II, Gliwice, 2014, Helion
3. Anna Kempa, Wprowadzenie do WPF. Tworzenie aplikacji w WPF przy użyciu XAML i C#, Gliwice, 2017, Helion
4. Bill Phillips, Programowanie aplikacji dla Androida : Big nerd ranch guide, Gliwice, 2018, Helion

zalecana/fakultatywna:

1. Dabit, Nader, React Native w akcji : twórz aplikacje na iOS i Android w JavaScriptcie, Warszawa, 2020, PWN
2. W3C, ARIA in HTML, Cambridge, Massachusetts, United States, 2023, W3C

Przedsiębiorczość innowacyjna nazwa przedmiotu
Innovative Entrepreneurship nazwa przedmiotu w języku angielskim
polski język wykładowy

przedmioty ogólne

kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Brak wymagań wstępnych.

Cele przedmiotu:

1. Zapoznanie studentów z istotą przedsiębiorczości i podstawowym aparatem pojęciowym z zakresu zarządzania i prowadzenia działalności gospodarczej w sektorze informatycznym.
2. Wykształcenie umiejętności pozyskiwania (z różnych źródeł) i syntezy informacji, planowania i organizowania. przedsięwzięć biznesowych.
3. Wykształcenie umiejętności współdziałania i kierowania zespołem projektowym.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	Student swobodnie posługuje się aparatem pojęciowym z zakresu przedsiębiorczości.	I1_W14
Umiejętności Absolwent potrafi:		
EU1	Student w oparciu o kluczowe modele analityczne identyfikuje potrzeby i luki rynkowe, planuje przedsięwzięcie biznesowe.	I1_U01b
EU2	Student przygotowuje dokument opisujący planowane przedsięwzięcie w formie teasera informacyjnego/inwestycyjnego.	I1_U01b

Kompetencje społeczne

Absolwent jest gotów do:

EK1	Student myśli i działa w sposób przedsiębiorczy.	I1_K04
-----	--	--------

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
7	2	Z	9	9	0	0	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	WPROWADZENIE (Przedsiębiorczość i przedsiębiorca. Przedsiębiorstwa typu start-up. Wynałazki i innowacje. Komercjalizacja wynalazków i innowacyjnych pomysłów.)	1
2	W	ASPEKTY EKONOMICZNO - ZARZĄDCZE W DZIAŁALNOŚCI GOSPODARCZEJ (Zarządzanie przedsięwzięciem, strategia, struktura i operacje. Planowanie, harmonogramowanie i budżetowanie zadań. Praca w zespole. Źródła finansowania działalności zawodowej, w tym działalności B+R. Analiza opłacalności przedsięwzięć - próg rentowności. Wycena technologii, innowacji i przedsiębiorstw).	3
3	W	FINANSOWANIE PRZEDSIĘWZIĘĆ (Źródła finansowania na różnych etapach rozwoju, próg rentowności, szacunkowy rachunek wyników planowanej działalności, szacunkowy rachunek przepływów gotówkowych, szacunkowy bilans przedsiębiorstwa, kalkulacja ryzyka).	2
4	W	DOKUMENT PREZENTUJĄCY PRZEDSIĘWZIĘCIE (Cel, pojęcie, funkcje, struktura teasera/pitch-decka/biznes-planu - analiza zewnętrzną i wewnętrzną, strategia, plan organizacyjny, techniczny, marketingowy i finansowy).	2
5	W	PREZENTACJA I SPRZEDAŻ POMYSŁU (Techniki budowania efektywnych procesów komunikowania, schematy skutecznej komunikacji.)	1
6	C	Od pomysłu do uruchomienia biznesu - źródła pomysłów na własny biznes, identyfikowanie potrzeb i luk rynkowych, badanie rynku. Kluczowe modele analityczne w obszarze przedsiębiorczości: SWOT, 5 sił Portera, model biznesu, model analizy okazji biznesowych, metody portfelowe.	3
7	C	Operacjonalizacja przedsięwzięcia w formie teasera/pitch-decka/biznesplanu.	3
8	C	Prezentacja operacjonalizacji własnych pomysłów.	3

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Konsultacje przedmiotowe i zaliczenie w bezpośrednim kontakcie z prowadzącym	7
2	Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury	10
3	Przygotowanie raportu, projektu, prezentacji	15

Metody dydaktyczne:

analiza przypadku, ćwiczenia, dyskusja, praca w grupach, projekt, symulacja, wykład, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

projekt, test

Kryteria oceny:

1. Na ocenę 5.0 student poprawnie wyjaśnia procesy związane z zarządzaniem przedsiębiorstwem, innowacją, biznes planem posługując się w poprawny sposób językiem dyscypliny
2. Na ocenę 5.0 student przeprowadza analizę trendów otoczenia, określa konsekwencje ich wpływów dla wariantu optymistycznego, pesymistycznego, najbardziej prawdopodobnego dla zaplanowanego przedsięwzięcia biznesowego
3. Na ocenę 5.0 student przygotowuje opis planowanego przedsięwzięcia uwzględniający planowaną strukturę kosztów i strumienie przychodów
4. Student na ocenę 5.0 akceptuje elastyczność i konieczność dostosowywania swoich postaw, decyzji, działań do szans i atutów oraz eliminacji zagrożeń i słabości towarzyszących aktywności zawodowej/gospodarczej

Literatura:

obowiązkowa:

1. J.Cieślik - Przedsiębiorczość dla ambitnych, Warszawa, 2010, Wydawnictwa Akademickie i Profesjonalne
2. B. Tracy - Przedsiębiorczość. Jak założyć i rozwijać własną firmę, Gliwice, 2021, Helion

zalecana/fakultatywna:

1. red. M.Strużycki - Przedsiębiorczość w teorii i praktyce, Warszawa, 2006, SGH
2. E. Ries - Droga Startupu. Metoda sukcesu, Gliwice, 2018, Helion
3. Z.Pierścioneł - Strategie konkurencji i rozwoju przedsiębiorstwa, Warszawa, 2010, PWN

Przetwarzanie języka naturalnego

nazwa przedmiotu

Processing of natural language nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Algorytmy i struktury danych.
2. Podstawy programowania w języku skryptowym Python.

Cele przedmiotu:

1. Praktyczne poznanie struktury oraz symbolicznych właściwości języka naturalnego. Wykorzystanie w praktyce poznanych metod przetwarzania dla języka naturalnego. Praktyczne wykorzystanie istniejących rozwiązań dla NLP.
2. Poznanie struktury oraz symbolicznych właściwości języka naturalnego. Algorytmy NLP.
3. Wykorzystanie w praktyce poznanych metod przetwarzania dla języka naturalnego.
4. Projektowanie rozwiązań dla zagadnień związanych z NLP.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	Student zna i rozumie podstawowe metody, techniki i narzędzia stosowane przy rozwiązywaniu prostych zadań informatycznych z zakresu przetwarzania języka naturalnego.	I1_W04

EW1	Student zna i rozumie podstawowe metody, techniki i narzędzia stosowane przy rozwiązywaniu prostych zadań informatycznychz zakresu przetwarzania języka naturalnego.	I1_W06
EW1	Student zna i rozumie podstawowe metody, techniki i narzędzia stosowane przy rozwiązywaniu prostych zadań informatycznychz zakresu przetwarzania języka naturalnego.	I1_W08
EW1	Student zna i rozumie podstawowe metody, techniki i narzędzia stosowane przy rozwiązywaniu prostych zadań informatycznychz zakresu przetwarzania języka naturalnego.	I1_W09
EW1	Student zna i rozumie podstawowe metody, techniki i narzędzia stosowane przy rozwiązywaniu prostych zadań informatycznychz zakresu przetwarzania języka naturalnego.	I1_W10
EW1	Student zna i rozumie podstawowe metody, techniki i narzędzia stosowane przy rozwiązywaniu prostych zadań informatycznychz zakresu przetwarzania języka naturalnego.	I1_W11
EW2	Student ma wiedzę na temat metod analizy składniowej, semantycznej, analizy sentymentu oraz zastosowania sieci neuronowych do zagadnień przetwarzania języka naturalnego.	I1_W04
EW2	Student ma wiedzę na temat metod analizy składniowej, semantycznej, analizy sentymentu oraz zastosowania sieci neuronowych do zagadnień przetwarzania języka naturalnego.	I1_W06
EW2	Student ma wiedzę na temat metod analizy składniowej, semantycznej, analizy sentymentu oraz zastosowania sieci neuronowych do zagadnień przetwarzania języka naturalnego.	I1_W08
EW2	Student ma wiedzę na temat metod analizy składniowej, semantycznej, analizy sentymentu oraz zastosowania sieci neuronowych do zagadnień przetwarzania języka naturalnego.	I1_W09
EW2	Student ma wiedzę na temat metod analizy składniowej, semantycznej, analizy sentymentu oraz zastosowania sieci neuronowych do zagadnień przetwarzania języka naturalnego.	I1_W10
EW2	Student ma wiedzę na temat metod analizy składniowej, semantycznej, analizy sentymentu oraz zastosowania sieci neuronowych do zagadnień przetwarzania języka naturalnego.	I1_W11
Umiejętności Absolwent potrafi:		
EU1	Student umie - zgodnie z zadaną specyfikacją - zaprojektować oraz zrealizować system informatyczny (aplikację) dla zagadnień związanych z NLP używając właściwych metod, technik i narzędzi.	I1_U01b
EU1	Student umie - zgodnie z zadaną specyfikacją - zaprojektować oraz zrealizować system informatyczny (aplikację) dla zagadnień związanych z NLP używając właściwych metod, technik i narzędzi.	I1_U02

EU1	Student umie - zgodnie z zadaną specyfikacją - zaprojektować oraz zrealizować system informatyczny (aplikację) dla zagadnień związanych z NLP używając właściwych metod, technik i narzędzi.	I1_U03
EU1	Student umie - zgodnie z zadaną specyfikacją - zaprojektować oraz zrealizować system informatyczny (aplikację) dla zagadnień związanych z NLP używając właściwych metod, technik i narzędzi.	I1_U04
EU1	Student umie - zgodnie z zadaną specyfikacją - zaprojektować oraz zrealizować system informatyczny (aplikację) dla zagadnień związanych z NLP używając właściwych metod, technik i narzędzi.	I1_U05
EU1	Student umie - zgodnie z zadaną specyfikacją - zaprojektować oraz zrealizować system informatyczny (aplikację) dla zagadnień związanych z NLP używając właściwych metod, technik i narzędzi.	I1_U07b
EU1	Student umie - zgodnie z zadaną specyfikacją - zaprojektować oraz zrealizować system informatyczny (aplikację) dla zagadnień związanych z NLP używając właściwych metod, technik i narzędzi.	I1_U08
EU1	Student umie - zgodnie z zadaną specyfikacją - zaprojektować oraz zrealizować system informatyczny (aplikację) dla zagadnień związanych z NLP używając właściwych metod, technik i narzędzi.	I1_U10
EU1	Student umie - zgodnie z zadaną specyfikacją - zaprojektować oraz zrealizować system informatyczny (aplikację) dla zagadnień związanych z NLP używając właściwych metod, technik i narzędzi.	I1_U12
EU1	Student umie - zgodnie z zadaną specyfikacją - zaprojektować oraz zrealizować system informatyczny (aplikację) dla zagadnień związanych z NLP używając właściwych metod, technik i narzędzi.	I1_U18
EU1	Student umie - zgodnie z zadaną specyfikacją - zaprojektować oraz zrealizować system informatyczny (aplikację) dla zagadnień związanych z NLP używając właściwych metod, technik i narzędzi.	I1_U20
EU1	Student umie - zgodnie z zadaną specyfikacją - zaprojektować oraz zrealizować system informatyczny (aplikację) dla zagadnień związanych z NLP używając właściwych metod, technik i narzędzi.	I1_U22
EU1	Student umie - zgodnie z zadaną specyfikacją - zaprojektować oraz zrealizować system informatyczny (aplikację) dla zagadnień związanych z NLP używając właściwych metod, technik i narzędzi.	I1_U23
Kompetencje społeczne Absolwent jest gotów do:		

EK1	Student zna i docenia rolę NLP we współczesnym świecie IT.	I1_K01
EK1	Student zna i docenia rolę NLP we współczesnym świecie IT.	I1_K02
EK1	Student zna i docenia rolę NLP we współczesnym świecie IT.	I1_K03
EK1	Student zna i docenia rolę NLP we współczesnym świecie IT.	I1_K04
EK1	Student zna i docenia rolę NLP we współczesnym świecie IT.	I1_K05
EK1	Student zna i docenia rolę NLP we współczesnym świecie IT.	I1_K06

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
8	6	E	18	0	0	18	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	Wprowadzenie do tematyki, podstawowe wiadomości wstępne.	1
2	W	Wyrażenia regularne i ich zastosowanie do ekstrakcji informacji.	1
3	W	Budowa słownika - tokenizator.	2
4	W	Wektory TD-IDF.	2
5	W	Analiza semantyczna.	2
6	W	Teoria sieci neuronowych od perceptronu do wstecznej propagacji.	1
7	W	Wektor słów Word2vec.	2
8	W	Konwolucyjne sieci neuronowe i kolejność słów.	1
9	W	Rekurencyjne sieci neuronowe.	1
10	W	Sieci LSTM (Long Short-Term Memory).	1
11	W	Sieci ciąg-ciąg, tłumaczenie tekstu i chatbot.	1
12	W	Ekstrakcja informacji i odpowiadanie na pytania.	1
13	W	Tagowanie częściami mowy (POS tagging).	1
14	W	Narzędzia specyficzne dla języka polskiego.	1
15	LK	Przypomnienie Python + Jupyter + biblioteki, uczenie maszynowe.	1
16	LK	Wyrażenia regularne.	1
17	LK	Budowa słownika tokenizator.	1
18	LK	Wektory TD-IDF.	2
19	LK	Analiza semantyczna.	2
20	LK	Podstawy sieci neuronowych - biblioteka Keras i TensorFlow.	1
21	LK	Wektor słów Word2vec.	2
22	LK	Konwolucyjne sieci neuronowe i kolejność słów.	1
23	LK	Rekurencyjne sieci neuronowe.	1
24	LK	Sieci LSTM (Long Short-Term Memory).	1

25	LK	Sieci ciąg-ciąg, tłumaczenie tekstu i chatbot.	1
26	LK	Ekstrakcja informacji i odpowiadanie na pytania.	1
27	LK	Tagowanie częściami mowy (POS tagging).	1
28	LK	Narzędzia specyficzne dla języka polskiego.	1
29	LK	Projekt i implementacja systemu przetwarzania tekstu. Testowanie systemu przetwarzania tekstu.	1

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Konsultacje przedmiotowe (w bezpośrednim kontakcie z prowadzącym)	10
2	Egzaminy i zaliczenia (w bezpośrednim kontakcie z prowadzącym)	10
3	Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury	30
4	Opracowanie wyników (w bezpośrednim kontakcie z prowadzącym)	20
5	Przygotowanie raportu, projektu, prezentacji, dyskusji	20
6	Realizacja zadań programistycznych (w bezpośrednim kontakcie z prowadzącym)	24

Metody dydaktyczne:

analiza przypadku, ćwiczenia laboratoryjne, dyskusja, konsultacje, pokaz, praca w grupach, praca z tekstem/materiałem audio i wideo, projekt, symulacja, wykład, laboratorium komputerowe, prezentacja, ćwiczenia projektowe z wykorzystaniem oprogramowania

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, dyskusja, egzamin pisemny, kolokwium, odpowiedź ustna, prezentacja, projekt, referat, rozwiązanie zadania problemowego, sprawozdanie z pracy indywidualnej, sprawozdanie z pracy zespołowej, test, zaliczenie pisemne, zaliczenie ustne, sprawdzian wiadomości z wykładów, kolokwium z zadań, praca własna-zadanie, quiz, test, rysunek, sprawozdanie

Kryteria oceny:

1. Na ocenę 5.0 Student zna i rozumie podstawowe metody, techniki i narzędzia stosowane przy rozwiązywaniu prostych zadań informatycznych z zakresu przetwarzania języka naturalnego powyżej 90% materiału omawianego na wykładzie.
2. Na ocenę 5.0 Student ma wiedzę na temat metod analizy składniowej, semantycznej, analizy sentymentu oraz zastosowania sieci neuronowych do zagadnień przetwarzania języka naturalnego powyżej 90% materiału omawianego na wykładzie.
3. Na ocenę 5.0 Student umie - zgodnie z zadaną specyfikacją - zaprojektować oraz zrealizować system informatyczny (aplikację) dla zagadnień związanych z NLP używając właściwych metod, technik i narzędzi na poziomie powyżej 90% materiału omawianego na laboratoriach.
4. Na ocenę 5.0 Student zna i docenia rolę NLP we współczesnym świecie IT.

Literatura:

obowiązkowa:

1. H. Lane, C. Howard, H.M. Hapke, 'Przetwarzanie języka naturalnego w akcji', Warszawa, 2021, PWN
2. praca zbiorowa, 'NLTK Documentation', 2022

zalecana/fakultatywna:

Przetwarzanie współbieżne nazwa przedmiotu
Concurrent processing nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Umiejętność programowania w języku C lub C++, oraz Java.
2. Ukończenie kursów z programowania proceduralnego i obiektowego, architektur systemów komputerowych, systemów operacyjnych oraz sieci komputerowych.

Cele przedmiotu:

1. Przedstawienie problematyki obliczeń współbieżnych, równoległych i rozproszonych oraz zapoznanie ze sposobami tworzenia programów równoległych i rozproszonych w różnych środowiskach sprzętowych i programowych
2. Nauczenie programowania z wykorzystaniem podstawowych narzędzi przetwarzania współbieżnego, równoległego i rozproszonego: bibliotek wątków, środowisk OpenMP i MPI.
3. Zapoznanie ze sposobami analizy poprawności i wydajności programów współbieżnych, równoległych i rozproszonych oraz z metodami rozwiązywania podstawowych problemów

projektowych w przetwarzaniu współbieżnym, równoległym i rozproszonym

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	Student zna podstawowe idee i pojęcia związane z przetwarzaniem współbieżnym, równoległym i rozproszonym	I1_W05
EW1	Student zna podstawowe idee i pojęcia związane z przetwarzaniem współbieżnym, równoległym i rozproszonym	I1_W06
EW2	Student zna podstawowe narzędzia tworzenia programów współbieżnych, równoległych i rozproszonych w ramach środowisk: wątków Pthreads, wątków Javy, OpenMP, MPI, gniazd oraz zdalnego wywołania procedur w systemie Linux i języku Java	I1_W06
EW2	Student zna podstawowe narzędzia tworzenia programów współbieżnych, równoległych i rozproszonych w ramach środowisk: wątków Pthreads, wątków Javy, OpenMP, MPI, gniazd oraz zdalnego wywołania procedur w systemie Linux i języku Java	I1_W11
EW3	Student potrafi identyfikować podstawowe przyczyny niepoprawności wykonania programów współbieżnych i równoległych, takie jak : zależności między instrukcjami w kodzie, sytuacje wyścigu oraz zakleszczenia lub zagłodzenia wątków i procesów	I1_W04
EW4	Student potrafi wskazać podstawowe czynniki wpływające na wydajność programów równoległych	I1_W04
Umiejętności Absolwent potrafi:		
EU1	Student potrafi modyfikować, uruchamiać i śledzić wykonanie prostych programów wykorzystujących: bibliotekę wątków Pthreads, środowisko OpenMP oraz specyfikację MPI.	I1_U01b
EU1	Student potrafi modyfikować, uruchamiać i śledzić wykonanie prostych programów wykorzystujących: bibliotekę wątków Pthreads, środowisko OpenMP oraz specyfikację MPI.	I1_U10
EU1	Student potrafi modyfikować, uruchamiać i śledzić wykonanie prostych programów wykorzystujących: bibliotekę wątków Pthreads, środowisko OpenMP oraz specyfikację MPI.	I1_U20
EU1	Student potrafi modyfikować, uruchamiać i śledzić wykonanie prostych programów wykorzystujących: bibliotekę wątków Pthreads, środowisko OpenMP oraz specyfikację MPI.	I1_U21

EU1	Student potrafi modyfikować, uruchamiać i śledzić wykonanie prostych programów wykorzystujących: bibliotekę wątków Pthreads, środowisko OpenMP oraz specyfikację MPI.	I1_U22
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Student potrafi pracować w zespole, realizować zadania w projektach interdyscyplinarnych.	I1_K03

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
7	4	E	18	0	0	18	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	Podstawowe idee i pojęcia związane z przetwarzaniem współbieżnym, równoległym i rozproszonym. Procesy i wątki, Biblioteka POSIX threads.	3
2	W	Problemy współbieżności. Zależności.	3
3	W	Programowanie w modelu pamięci wspólnej - OpenMP. Pule wątków, wektoryzacja	3
4	W	Metodologia tworzenia programów współbieżnych	3
5	W	Programowanie w modelu przesyłania komunikatów - MPI	3
6	W	Współbieżność w środowiskach obiektowych	3
7	LK	Przeprowadzenie pomiaru czasu CPU i zegarowego wykonania operacji. Organizacja środowiska tworzenia oprogramowania w systemie Linux.	3
8	LK	Tworzenie procesów i wątków w systemie Linux. Podstawy biblioteki POSIX threads, atrybuty wątków.	3
9	LK	Synchronizacja wątków, mutexy i zmienne warunku	3
10	LK	Nabycie umiejętności tworzenia i implementacji programów równoległych z wykorzystaniem OpenMP. Doskonalenie znajomości OpenMP. Pule wątków i wektoryzacja w OpenMP	3
11	LK	Programowanie w środowisku rozproszonym - MPI. Konfiguracja połączenia. Podstawowe programy. Zaawansowane metody komunikacji.	3
12	LK	Wątki w środowisku obiektowym Java	3

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
-----	--------------------	---------------

1	Konsultacje przedmiotowe (w bezpośrednim kontakcie z prowadzącym)	29
2	Egzaminy i zaliczenia (w bezpośrednim kontakcie z prowadzącym)	10
3	Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury	10
4	Opracowanie wyników	15

Metody dydaktyczne:

wykład, laboratorium komputerowe

Sposoby weryfikacji i oceny efektów uczenia się:

egzamin pisemny, sprawozdanie z pracy indywidualnej

Kryteria oceny:

1. Na ocenę 5.0 student uzyskuje powyżej 89% maksymalnej liczby punktów z części dotyczącej efektów kształcenia EW1-4 wydzielonej z testu końcowego
2. Na ocenę 5,0 student zalicza wszystkie ćwiczenia laboratoryjne związane z efektem kształcenia EU1 oraz otrzymuje ocenę średnią z przygotowania do ćwiczeń laboratoryjnych i ze sprawności realizacji ćwiczeń laboratoryjnych powyżej 4,75
3. Na ocenę 5.0 student uzyskuje powyżej 89% maksymalnej liczby punktów z części dotyczącej efektów kształcenia EK1 wydzielonej z testu końcowego

Literatura:

obowiązkowa:

1. L. Ridgeway Scott, Terry Clark, Babak Bagheri - Scientific Parallel Computing, , 2005, Princeton University Press

zalecana/fakultatywna:

1. A. Karbowski , E. Niewiadomska-Szynkiewicz - Obliczenia równoległe i rozproszone
2. A. Grama et al - Introduction to Parallel Computing, 2003, Addison-Wesley

Przygotowanie dyplomowe	
nazwa przedmiotu	
Diploma preparation	
nazwa przedmiotu w języku angielskim	
polski	
język wykładowy	

przedmioty kierunkowe

kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Znajomość treści programowych z dotychczasowego przebiegu studiów.
2. Znajomość języka angielskiego w stopniu umożliwiającym korzystanie z dokumentacji, studiowanie literatury i uczestnictwo w zajęciach.

Cele przedmiotu:

1. Przygotowanie studentów do zaplanowania i napisania pracy inżynierskiej. Wybór tematu pracy dyplomowej, której tematyka związana jest z kierunkiem studiów.
2. Przygotowanie deklaracji wyboru tematu pracy, określenie celu i zakresu pracy.
3. Zapoznanie się z procedurami dyplomowania na kierunku, niezbędnymi dokumentami oraz systemem antyplagiatowym.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	Student ma ogólną wiedzę o aktualnych kierunkach rozwoju informatyki i najnowszych osiągnięciach w kontekście problematyki związanej z pracą inżynierską.	I1_W13
Umiejętności Absolwent potrafi:		

EU1	Student potrafi określić cel pracy dyplomowej oraz dobrać narzędzia do realizacji wyznaczonych zadań. Potrafi pozyskiwać informacje z literatury i innych źródeł w zakresie informatyki, także w języku angielskim. Potrafi integrować uzyskane informacje, dokonywać ich interpretacji i krytycznej oceny, wyciągać wnioski oraz formułować i wyczerpująco uzasadniać opinie.	I1_U02
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Student widzi potrzebę ciągłego samodoskonalenia i zdobywania nowej wiedzy i kompetencji w perspektywie kariery zawodowej.	I1_K06

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
7	1	Z	0	0	0	0	0	15	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	S	Przedstawienie wymagań formalnych, procedury dyplomowania, dokumentów. System antyplagiatowy. Przedstawienie zasad powstawania pracy dyplomowej w aspektach planowania pracy, wyznaczania celów i zakresu pracy, definiowania wymagań oraz doboru używanych narzędzi. Omówienie zasad edycji i redakcji tekstu.	6
2	S	Sztuka prezentacji, czyli jak dobrze przygotować prezentację, która będzie wartościowa merytorycznie, czytelna i interesująca dla odbiorcy. Wartość pracy w grupie oraz ciągłego samodoskonalenia i zdobywania nowej wiedzy i kompetencji w perspektywie kariery zawodowej.	6
3	S	Omówienie sposobów wyszukiwania informacji ze szczególnym uwzględnieniem tematyki pracy. Poprawne zamieszczanie w tekście odwołań bibliograficznych. Jak pozyskiwać informacje z literatury i innych źródeł w zakresie informatyki, także w języku angielskim. Integracja uzyskanych informacji, dokonywanie ich interpretacji i krytycznej oceny, a także wyciągać wnioski oraz formułować i wyczerpująco uzasadniać opinie.	3

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
-----	--------------------	---------------

1	Przygotowanie się do zajęć i dyskusji, w tym studiowanie zalecanej literatury, opracowanie wyników przeprowadzonych badań, przygotowanie raportu, projektu, prezentacji itp.	7
2	Konsultacje przedmiotowe i zaliczenie (w bezpośrednim kontakcie z prowadzącym)	8

Metody dydaktyczne:

dyskusja, konsultacje, seminarium, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, dyskusja, prezentacja

Kryteria oceny:

1. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji kompetencji społecznych.
2. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji umiejętności.
3. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji wiedzy.

Literatura:

obowiązkowa:

1. Renata Wojciechowska. Przewodnik metodyczny pisania pracy dyplomowej. Difin. 2010.
2. Radosław Zenderowski. Praca magisterska. Licencjat. Przewodnik po metodologii pisania i obrony pracy dyplomowej. 2022

zalecana/fakultatywna:

1. Joanna Wrycza-Bekier. Magia słów. Jak pisać teksty, które porwą tłumy. OnePress. 2021.

Przygotowanie pracy dyplomowej nazwa przedmiotu
Diploma Project Preparation nazwa przedmiotu w języku angielskim
polski język wykładowy

przedmioty kierunkowe

kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Znajomość treści programowych z dotychczasowego przebiegu studiów.
2. Znajomość języka angielskiego w stopniu umożliwiającym korzystanie z dokumentacji, studiowanie literatury i uczestnictwo w zajęciach.

Cele przedmiotu:

1. Przygotowanie studentów do zaplanowania i napisania pracy inżynierskiej. Wybór tematu pracy dyplomowej, której tematyka związana jest z kierunkiem studiów.
2. Przygotowanie deklaracji wyboru tematu pracy, określenie celu i zakresu pracy. Zapoznanie się z procedurami dyplomowania na kierunku, niezbędnymi dokumentami oraz systemem antyplagiatowym.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	Student zna pojęcia związane z tematyką realizowanej pracy dyplomowej.	I1_W13
Umiejętności Absolwent potrafi:		
EU1	Student potrafi sformułować, rozwiązać oraz przedstawić zagadnienie/problem omawiany w realizowanej pracy dyplomowej. Student potrafi wykorzystać zdobytą w czasie studiów wiedzę i umiejętności do rozwiązania zdefiniowanego zadania.	I1_U02

Kompetencje społeczne		
Absolwent jest gotów do:		
EK1	Student potrafi pracować samodzielnie i rozwiązywać napotkane problemy, np. poprzez korzystanie ze specjalistycznej literatury lub konsultacje z promotorem/ekspertami.	I1_K06

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
8	8	Z	0	0	0	0	5	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	P	Konsultacje związane z przygotowaniem pracy dyplomowej.	2
2	P	Konsultacje związane z przygotowaniem pracy dyplomowej.	2
3	P	Konsultacje związane z przygotowaniem pracy dyplomowej.	1

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Przygotowanie się do zajęć i dyskusji, w tym studiowanie zalecanej literatury, opracowanie wyników przeprowadzonych badań, przygotowanie raportu, projektu, prezentacji itp. Przygotowanie pracy dyplomowej.	170
2	Konsultacje przedmiotowe i zaliczenie (w bezpośrednim kontakcie z prowadzącym)	25

Metody dydaktyczne:

dyskusja, konsultacje, projekt, seminarium

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, dyskusja, prezentacja, projekt

Kryteria oceny:

1. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji kompetencji społecznych.
2. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji wiedzy.

3. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji umiejętności.

Literatura:

obowiązkowa:

1. Joanna Wrycza-Bekier. Magia słów. Jak pisać teksty, które porwą tłumy. OnePress. 2021.
2. Renata Wojciechowska. Przewodnik metodyczny pisanie pracy dyplomowej. Difin. 2010.

zalecana/fakultatywna:

1. Radosław Zenderowski. Praca magisterska. Licencjat. Przewodnik po metodologii pisanie i obrony pracy dyplomowej. 2022.

Rachunek prawdopodobieństwa i statystyka nazwa przedmiotu
The calculus of probability and the statistician nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Wymagana jest znajomość analizy matematycznej w zakresie funkcji jednej i wielu zmiennych.

Cele przedmiotu:

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	Student zna podstawowe definicje i twierdzenia rachunku prawdopodobieństwa oraz podstawy wnioskowania statystycznego.	I1_W01
Umiejętności Absolwent potrafi:		
EU1	Student umie stosować wzór włącz-wyłącz, wzór na prawdopodobieństwo całkowite i wzór Bayesa oraz podstawowe pojęcia kombinatoryczne (permutacje, kombinacje, wariacje z powtórzeniami i bez powtórzeń) do obliczania prawdopodobieństw.	I1_U06b
EU2	Student potrafi wyznaczyć parametry rozkładów zmiennych i wektorów losowych oraz wykorzystać twierdzenia graniczne i prawa wielkich liczb do szacowania prawdopodobieństw.	I1_U06b
EU3	Student umie prowadzić proste wnioskowania statystyczne.	I1_U06b
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Student dostrzega ograniczenia własnej wiedzy, rozumie potrzebę dalszego kształcenia oraz potrzebę dzielenia się wiedzą.	I1_K06

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
3	4	Z	18	0	0	18	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
-----	-------------	----------------	---------------

1	W	Aksjomatyczna definicja prawdopodobieństwa, własności, wzór włącz-wyłącz. Twierdzenie o ciągłości prawdopodobieństwa, przykłady przestrzeni probabilistycznych, paradoks Bertranda. Schemat Bernoullego, prawdopodobieństwo warunkowe, wzór na prawdopodobieństwo całkowite, wzór Bayesa.	3
2	W	Zmienna losowa, rozkład prawdopodobieństwa, rozkład dyskretny, absolutnie ciągły, dystrybuanta. Parametry rozkładów: wartość oczekiwana, wariancja.	3
3	W	Wektory losowe, niezależność zmiennych losowych, korelacja, kowariancja, wariancja sumy zmiennych losowych. Przegląd rozkładów prawdopodobieństwa.	2
4	W	Prawa wielkich liczb, centralne twierdzenie graniczne.	2
5	W	Elementy statystyki opisowej. Estymacja punktowa, metoda największej wiarygodności. Estymacja przedziałowa.	3
6	W	Testowanie hipotez statystycznych, testy parametryczne, testy dotyczące wartości średniej. Testy dotyczące wariancji, testy dotyczące wskaźnika struktury. Testy nieparametryczne, test chi kwadrat, test chi kwadrat Fischera.	3
7	W	Informacja o procesach stochastycznych, analizy wydajności prostych układów sprzętowo-programowych z wykorzystaniem procesów stochastycznych, proces Wienera.	2
8	LK	Wykorzystanie schematów kombinatorycznych do rozwiązywania zadań z rachunku prawdopodobieństwa, prawdopodobieństwo geometryczne.	3
9	LK	Zastosowanie schematu Bernoullego, prawdopodobieństwo warunkowe, niezależność zdarzeń, wzór na prawdopodobieństwo całkowite, wzór Bayesa, obliczanie niezawodności prostych układów sprzętowych.	3
10	LK	Badanie zmiennych losowych i ich rozkładów, wyznaczanie dystrybuanty, odczytywanie rozkładu z dystrybuanty, wyznaczanie parametrów i ich interpretacja.	3
11	LK	Zastosowanie centralnych twierdzeń granicznych: tw. Moivre'a-Laplace'a, tw. Lindeberga-Levy'ego, obliczanie niezawodności i wydajności prostych systemów programowych.	4
12	LK	Wyznaczanie przedziału ufności, testowanie hipotez statystycznych - zadania.	3
13	LK	Sprawdzian wiadomości.	2

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury.	70
2	Konsultacje przedmiotowe i zaliczenie w bezpośrednim kontakcie z prowadzącym	14

Metody dydaktyczne:

ćwiczenia, konsultacje, wykład, ćwiczenia tablicowe, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, kolokwium, odpowiedź ustna, sprawdzian wiadomości z wykładów, kolokwium z zadań

Kryteria oceny:

1. W zakresie EW1 Na ocenę 5.0: Student zna podstawowe definicje i twierdzenia rachunku prawdopodobieństwa w stopniu bardzo dobrym (uzyskał od 90% do 100% punktów z testu).
2. W zakresie EU3 Na ocenę 5.0: Student umie prowadzić proste wnioskowania statystyczne w stopniu bardzo dobrym (uzyskał od 90% do 100% punktów z kolokwium).
3. W zakresie EU2 Na ocenę 5.0: Student potrafi wyznaczyć parametry rozkładów zmiennych losowych oraz wykorzystać twierdzenia graniczne i prawa wielkich liczb do szacowania prawdopodobieństw w stopniu bardzo dobrym (uzyskał od 90% do 100% punktów z kolokwium).
4. W zakresie EU1 Na ocenę 5.0: Student umie stosować wzór włącz-wyłącz, wzór na prawdopodobieństwo całkowite i wzór Bayesa oraz podstawowe pojęcia kombinatoryczne w stopniu bardzo dobrym (uzyskał od 90% do 100% punktów z kolokwium).
5. W zakresie EK1 Student rozwiązuje zadania domowe, korzysta z materiałów dostępnych na platformie e-learningowej, na zajęciach lub podczas konsultacji, zadaje pytania o charakterze merytorycznym, chętnie prezentuje rozwiązania zadań i problemów na ćwiczeniach. Poszukuje odpowiedzi na pytania w literaturze przedmiotu, potrafi prowadzić nieformalną rozmowę o zagadnieniach merytorycznych związanych z przedmiotem.

Literatura:

obowiązkowa:

1. A. Plucińska, E. Pluciński - Probabilistyka, Warszawa, 2000, WNT
2. W. Krysicki, J. Bartos, W. Dyczka, K. Królikowska, M. Wasilewski - Rachunek prawdopodobieństwa i statystyka matematyczna w zadaniach, cz. I i II, Warszawa, 2004, PWN

zalecana/fakultatywna:

1. J. Greń - Statystyka matematyczna. Modele i zadania, Warszawa, 1982, PWN
2. M. Wiciak - Elementy probabilistyki w zadaniach, Kraków, 2008, Wydawnictwo PK
3. J. Jakubowski, R. Sztencel - Wstęp do teorii prawdopodobieństwa, Warszawa, 2001, SCRIPT

Seminarium dyplomowe nazwa przedmiotu
Diploma seminar nazwa przedmiotu w języku angielskim
polski język wykładowy

przedmioty kierunkowe

kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Umiejętność tworzenia prezentacji multimedialnych - posługiwanie się podstawowymi narzędziami informatycznymi w tym zakresie (Latex, MS Power Point, etc.).
2. Wybrany temat pracy dyplomowej zaakceptowany przez promotora pracy i opiekuna kierunku.

Cele przedmiotu:

1. Zapoznanie studentów z formalnymi zasadami dotyczącymi struktury i warunkami pisania i obrony pracy dyplomowej, przebiegu egzaminu dyplomowego i procedury antyplagiatowej.
2. Student zapoznaje się z formalnymi zasadami i warunkami realizacji, pisania i obrony pracy inżynierskiej w tym przebiegu egzaminu dyplomowego.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	Znajomość prawa autorskiego i zagadnień etycznych związanych z tworzeniem pracy dyplomowej inżynierskiej. Analiza przydatności narzędzi informatycznych do tworzenia prac dyplomowych, oprogramowania w ramach pracy dyplomowej i prezentacji uzyskanych wyników.	I1_W13
Umiejętności Absolwent potrafi:		

EU1	Umiejętność posługiwania się niezbędnymi narzędziami do tworzenia i składu tekstów prac technicznych (MS Word, Latex), znajomość podstawowych elementów struktury takich prac, umiejętność wykonania prezentacji multimedialnej pokazującej wyniki opisane w pracy inżynierskiej.	I1_U23
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Umiejętność prezentowania (prezentacja indywidualna) efektów pracy inżynierskiej na forum grupy, uczestnictwo w dyskusji merytorycznej, sztuka argumentacji.	I1_K03

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
8	2	Z	0	0	0	0	0	30	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	S	Wprowadzenie do tematyki seminarium- omówienie i krótka prezentacja dostępnych narzędzi do tworzenia prac dyplomowych, tekstów technicznych i prezentacji multimedialnych. Omówienie zasad pisania pracy dyplomowej, przykładowa struktura pracy, wymagania dotyczące poszczególnych elementów pracy dyplomowej. Omówienie i prezentacja działania systemu plagiat.pl. Omówienie zasad unikania zapożyczeń, właściwe tworzenie referencji w pracy, umiejętność korzystania ze źródeł zewnętrznych.	10
2	S	Przygotowanie i prezentacja tematyki i zakresu merytorycznego prac dyplomowych studentów - wstępne prezentacje indywidualne (limit czasowy - 5 minut).	10
3	S	Przygotowanie finalnych prezentacji multimedialnych oraz wygłoszenie indywidualnie przez studentów na forum grupy prezentacji wyników pracy składającej się na pracę dyplomową inżynierską - wystąpienia indywidualne (limit czasowy- 10 min na każda prezentacje + 5 min na dyskusje). z profilem studiowanego kierunku.	10

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
-----	--------------------	---------------

1	Przygotowanie się do zajęć i dyskusji, w tym studiowanie zalecanej literatury, opracowanie wyników przeprowadzonych badań, przygotowanie raportu, projektu, prezentacji itp. Przygotowanie pracy dyplomowej.	14
2	Konsultacje przedmiotowe i zaliczenie (w bezpośrednim kontakcie z prowadzącym)	16

Metody dydaktyczne:

dyskusja, konsultacje, seminarium, wykład, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, dyskusja, prezentacja

Kryteria oceny:

1. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji kompetencji społecznych.
2. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji umiejętności.
3. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji wiedzy.

Literatura:

obowiązkowa:

1. Krystek Jacek. Poradnik pisania pracy dyplomowej 2021. Wydawnictwo Politechniki Łódzkiej
2. Radosław Zenderowski. Technika pisania prac magisterskich i licencjackich 2020. CeDeWu

zalecana/fakultatywna:

1. Renata Wojciechowska. Przewodnik metodyczny pisania pracy dyplomowej. Difin. 2010.

Systemy informatyczne w zarządzaniu nazwa przedmiotu
Information systems in management nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Znajomość języka angielskiego w stopniu komunikatywnym umożliwiającym uczestnictwo w zajęciach oraz korzystanie z literatury fachowej.

Cele przedmiotu:

1. Zapoznanie z modelami biznesowymi stosowanymi w e-biznesie
2. Wskazanie możliwości zastosowania nowoczesnych technologii IT w innowacyjnym biznesie
3. Zapoznanie studentów z narzędziami Business Intelligence

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	Student zna podstawowe pojęcia z zakresu ekonomii, zarządzania i prowadzenia działalności gospodarczej w sektorze informatycznym.	I1_W14
Umiejętności Absolwent potrafi:		
EU1	Student potrafi pozyskiwać informacje z literatury, baz danych oraz innych źródeł, integrować je, dokonywać ich interpretacji oraz wyciągać wnioski i formułować opinie.	I1_U01b
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Student potrafi pracować indywidualnie i w zespole informatyków, w tym zarządzać swoim czasem oraz podejmować zobowiązania i dotrzymywać terminów.	I1_K03

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
7	4	Z	18	0	0	0	18	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	Tworzenie modeli biznesowych. Design thinking. Industry 4.0. Sztuczna inteligencja, wirtualna i rozszerzona rzeczywistość, blockchain i inne zagadnienia w biznesie. Business Intelligence. Power BI. Sustainability a IT.	18
2	P	Zapoznanie z narzędziem Power BI. Opracowanie własnego modelu biznesowego związanego z branżą IT. Warsztaty z design thinking. Zarządzanie projektem. Konsultacje z ekspertem/mentorem. Praca w grupach. Realizacja projektu zgodnie z ustalonym harmonogramem.	18

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Konsultacje przedmiotowe w bezpośrednim kontakcie z prowadzącym	15
2	Opracowanie wyników z przeprowadzonych działań, przygotowanie raportu, projektu, prezentacji itp.	15
3	Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury	25
4	Zaliczenia w bezpośrednim kontakcie z prowadzącym	9

Metody dydaktyczne:

konsultacje, praca w grupach, projekt, wykład, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, dyskusja, prezentacja, projekt

Kryteria oceny:

1. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji kompetencji społecznych.
2. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji umiejętności.
3. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji wiedzy.

Literatura:

obowiązkowa:

1. Alexander Osterwalder, Yves Pigneur. TWORZENIE MODELI BIZNESOWYCH PODRĘCZNIK WIZJONERA. 2022.
2. Jerzy Surma. Business Intelligence. Wydawnictwo Naukowe PWN. 2012.

zalecana/fakultatywna:

1. Greg Deckler. Mastering Microsoft Power BI (ebook)

Systemy operacyjne nazwa przedmiotu
Operating systems nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Znajomość struktur danych
2. Umiejętność programowania w języku C/C++

Cele przedmiotu:

1. Zapoznanie studentów z budową i strukturą systemu operacyjnego oraz funkcjonalnością wszystkich jego modułów
2. Zaznajomienie studentów z zasadami działania systemów operacyjnych: Unix, Linux i Windows
3. Zaznajomienie studentów z wybranymi funkcjami systemowymi systemu linux i praktyczne ich wykorzystanie.
4. Implementacja problemów synchronizacji i komunikacji procesów z wykorzystaniem IPC

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	Student zna zagadnienia z zakresu systemów operacyjnych - budowy, funkcjonalności wszystkich podsystemów	I1_W05
Umiejętności Absolwent potrafi:		
EU1	Student potrafi zastosować polecenia i funkcje systemowe systemu linux	I1_U11
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Student potrafi realizować projekty w zadanym terminie	I1_K03

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
3	5	E	18	0	0	18	9	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	Budowa i funkcjonalność systemów operacyjnych i wszystkich jego podsystemów,	18
2	LK	System linux - polecenia systemowe, skrypty powłoki bash, AWK, funkcje systemowe związane z czasem, plikami, procesami, wątkami, mechanizmy synchronizacji procesów - IPC	18
3	P	Realizacja projektów związanych z synchronizacją procesów	9

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
-----	--------------------	---------------

1	Przygotowanie do zajęć	30
2	Opracowanie projektów	40
3	Konsultacje przedmiotowe (w bezpośrednim kontakcie z prowadzącym)	25
4	Opracowanie wyników	5
5	Egzaminy i zaliczenia (w bezpośrednim kontakcie z prowadzącym)	5

Metody dydaktyczne:

konsultacje, projekt, wykład, laboratorium komputerowe, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, egzamin pisemny, kolokwium, odpowiedź ustna, projekt, sprawozdanie z pracy indywidualnej, test, zaliczenie pisemne, sprawdzian wiadomości z wykładów, sprawozdanie

Kryteria oceny:

1. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji wiedzy.
2. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji umiejętności.
3. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji kompetencji społecznych.

Literatura:

obowiązkowa:

1. Abraham Silberschatz, James Peterson, Peter Galvin, PODSTAWY SYSTEMÓW OPERACYJNYCH, Warszawa, 2005, WNT
2. A. Jasinska-Suwada, S. Plichta, PRZEWODNIK DO CWICZEN Z PRZEDMIOTU: SYSTEMY OPERACYJNE, Kraków, 2001, Wydawnictwo PK
3. A. Jasinska-Suwada, S. Plichta, PRZEWODNIK DO CWICZEN Z PRZEDMIOTU: SYSTEMY OPERACYJNE cz II, Kraków, 2010, Wydawnictwo PK
4. M. Mitchell, J. Oldham, A. Samuel, LINUX Programowanie dla zaawansowanych, Warszawa, 2002, Wydawnictwo RM
5. N. Matthew R. Stones, Linux. Programowanie, Warszawa, 1999, Read Me
6. P. Yosifovich, A. Ionescu, „Windows od środka”, Helion 2023

zalecana/fakultatywna:

1. W. Richard Stevens, UNIX Programowanie usług sieciowych, Warszawa, 2001, WNT
2. K. Stencel, Systemy operacyjne. Struktura i zasady budowy, Warszawa, 2006, Wydawnictwo PJWSTK

Systemy wbudowane i mikrokontrolery

nazwa przedmiotu

Microcontrollers and embedded systems nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Jest zalecane, by słuchacze tego wykładu znali podstawy budowy i programowania mikrokontrolerów.

Cele przedmiotu:

1. Celem przedmiotu jest zdobycie wiedzy i umiejętności projektowania i implementacji aplikacji wbudowanych.
2. Główne części wykładu obejmują tematykę wzorców projektowych dla aplikacji wbudowanych oraz obsługi układów peryferyjnych.
3. Część wykładu poświęcona jest zagadnieniu sterowania układami zewnętrznymi przy użyciu mikrokontrolerów.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW2	Zna wzorce projektowe stosowane w projektowaniu systemów wbudowanych.	I1_W05

Umiejętności		
Absolwent potrafi:		
EU2	Budować proste systemy wbudowane. Potrafi zastosować wzorce projektowe dla systemów wbudowanych w implementacji takich systemów. Wykorzystać protokoły komunikacyjne do implementacji komunikacji w rozproszonych systemach wbudowanych oraz algorytm sterowania cyfrowego w projektowaniu aplikacji wbudowanych.	I1_U19
Kompetencje społeczne		
Absolwent jest gotów do:		
EK1	Student potrafi pracować w zespole, realizować zadania w projektach interdyscyplinarnych.	I1_K03

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
5	5	Z	18	0	0	18	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	LK	Implementacja wzorców projektowych dla układów wbudowanych w aplikacjach wbudowanych. Projektowanie rozproszonych aplikacji wbudowanych wykorzystujących protokoły komunikacyjne. Programowanie na platformie Raspberry Pi. Programowanie na platformie STM32Discovery	18
2	W	Magistrale komunikacyjne: współpraca ze sprzętem. Wzorce projektowe dla układów wbudowanych: Super Loop. Wzorce projektowe dla układów wbudowanych: aplikacje sterowane zdarzeniowo, wykorzystanie automatów skończonych w modelowaniu i implementacji aplikacji wbudowanych. Wzorce projektowe dla układów wbudowanych: model planisty co-operative. Wzorce projektowe dla układów wbudowanych: model planisty pre-emptive, systemy operacyjne dla układów wbudowanych 6. Układy rozproszone - metody komunikacji, obsługa błędów. Sterowanie układami zewnętrznymi	18

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Konsultacje przedmiotowe w bezpośrednim kontakcie z prowadzącym	15

2	Zaliczenie w bezpośrednim kontakcie z prowadzącym	10
3	Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury	30
4	Przygotowanie raportu, projektu, prezentacji, dyskusji	34

Metody dydaktyczne:

ćwiczenia laboratoryjne, konsultacje, praca w grupach, projekt, wykład, laboratorium komputerowe

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, dyskusja, kolokwium, odpowiedź ustna, projekt, referat, rozwiązanie zadania problemowego, sprawozdanie z pracy indywidualnej, sprawozdanie z pracy zespołowej, test, zaliczenie pisemne, zaliczenie ustne, sprawozdanie

Kryteria oceny:

1. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji wiedzy.
2. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji umiejętności.
3. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji kompetencji społecznych.

Literatura:

obowiązkowa:

1. Labrosse, Jean J. Embedded software, 2008

zalecana/fakultatywna:

1. R.Williams Real-Time Systems Development
2. M.J.Pont Patterns for Time-Triggered Embedded Systems
3. Jim Ledin Embedded Control Systems in C/C++

Techniki multimedialne nazwa przedmiotu
Multimedia techniques nazwa przedmiotu w języku angielskim
polski język wykładowy

przedmioty kierunkowe

kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Znajomość analizy matematycznej
2. Znajomość podstaw grafiki komputerowej
3. Znajomość obsługi programów do edycji grafiki rastrowej i wektorowej

Cele przedmiotu:

1. Wprowadzenie pojęć dotyczących różnego typu mediów: obrazów, dźwięku i video, ich charakterystyka i właściwości.
2. Zapoznanie studentów z narzędziami do przetwarzania danych graficznych, dźwiękowych i video, obsługa formatów plików, metod kompresji oraz sposób ich integracji.
3. Zaznajomienie z reprezentacjami obiektów trójwymiarowych w modelowaniu cyfrowym
4. Zaznajomienie z modelami powierzchni i algorytmami renderingu

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza		
Absolwent zna i rozumie:		
EW4	pojęcia z zakresu fizyki niezbędne do rozwiązywania problemów informatycznych związanych przede wszystkim z modelowaniem procesów w czasie rzeczywistym.	I1_W02
EW5	zagadnienia w zakresie algorytmiki i złożoności obliczeniowej algorytmów. Zna metody, technologie i narzędzia projektowania i implementacji algorytmów.	I1_W04
EW6	w sposób uporządkowany ogólne pojęcia z zakresu grafiki komputerowej, komunikacji człowiek-komputer, budowania prostych interfejsów graficznych.	I1_W07

Umiejętności Absolwent potrafi:		
EU5	pozyskiwać informacje z literatury, baz danych oraz innych źródeł, integrować je, dokonywać ich interpretacji oraz wyciągać wnioski i formułować opinie.	I1_U01b
EU6	ocenić złożoność obliczeniową algorytmów i problemów	I1_U10
EU7	wykonać analizę sposobu funkcjonowania systemu informatycznego i ocenić istniejące rozwiązania informatyczne, przynajmniej w odniesieniu do ich cech funkcjonalnych.	I1_U20
EU8	ocenić, na podstawowym poziomie, przydatność rutynowych metod i narzędzi informatycznych oraz wybrać i zastosować właściwą metodę i narzędzia do typowych zadań informatycznych.	I1_U22
Kompetencje społeczne Absolwent jest gotów do:		
EK2	przekazywania informacji o osiągnięciach informatyki i wielu aspektach zawodu informatyka w sposób powszechnie zrozumiały.	I1_K05

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
6	3	Z	9	0	0	18	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	LK	Projektowanie i realizacja grafiki w reprezentacji cyfrowej i proceduralnej. Tworzenie obrazów rastrowych w standardowych środowiskach graficznych. Tworzenie tekstur do wykorzystania w scenach 3D.	6
2	LK	Modelowanie 3D, rendering i animacja w standardowym środowisku DCC	7
3	LK	Efekty specjalne, montaż i postprodukcja materiałów multimedialnych	5
4	W	Techniki multimedialne - zastosowanie, korzyści i perspektywy rozwoju, etapy dążenia do realizmu	1
5	W	Reprezentacje obiektów 3D	1
6	W	Modelowanie powierzchni w przestrzeni wielowymiarowej	1
7	W	Algorytmy widoczności i eliminacji obiektów zasłoniętych	1

8	W	Modele oświetlenia	1
9	W	Funkcja BRDF i modele fizyki powierzchni	1
10	W	Model śledzenia promienia i jego rozwinięcia	1
11	W	Modelowanie zjawisk przestrzennych w scenach 3D	1

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Konsultacje przedmiotowe w bezpośrednim kontakcie z prowadzącym	15
2	Zaliczenie w bezpośrednim kontakcie z prowadzącym	5
3	Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury	15
4	Przygotowanie raportu, projektu, prezentacji, dyskusji	13

Metody dydaktyczne:

projekt, wykład, laboratorium komputerowe

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, kolokwium, projekt, sprawozdanie

Kryteria oceny:

1. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji kompetencji społecznych.
2. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji umiejętności.
3. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji umiejętności.
4. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji umiejętności.
5. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji umiejętności.
6. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji wiedzy.
7. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji wiedzy.
8. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji wiedzy.

Literatura:

obowiązkowa:

1. James D. Foley i inni "Wprowadzenie do grafiki komputerowej"
2. Pazdur Wojciech "3ds Max. Leksykon"

zalecana/fakultatywna:

1. David Dabner, Sandra Stewart, Abbie Vickress "Graphic design school : a foundation course for graphic designers working in print, moving image and digital media"

Technologie obiektowe nazwa przedmiotu
Object Oriented Methods nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Znajomość programowania co najmniej w jednym w obiektowym języku programowania.
2. Znajomość programowania strukturalnego.

Cele przedmiotu:

1. Zaznajomienie studentów z zasadami modelowania obiektowego.
2. Implementacja w obiektowych językach programowania.
3. Wdrożenie pojęć wzorców projektowych.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		

EW1	Ma wiedzę ogólną w zakresie języków i paradygmatów programowania, programowania obiektowego oraz inżynierii oprogramowania.	I1_W06
EW2	Ma szczegółową wiedzę nt. algorytmiki, projektowania i programowania obiektowego, baz danych i sztucznej inteligencji.	I1_W08
Umiejętności Absolwent potrafi:		
EU1	Umie stworzyć model obiektowy prostego systemu (np. w języku UML).	I1_U09
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Potrafi pracować indywidualnie i w zespole informatyków, w tym także potrafi zarządzać swoim czasem oraz podejmować zobowiązania i dotrzymywać terminów.	I1_K03

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
6	3	Z	9	0	0	18	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	LK	Implementacja wzorców projektowych w wybranych językach programowania i zastosowaniach.	18
2	W	Podstawowe pojęcia obiektowości.	1
3	W	Język UML i zastosowane w modelowaniu obiektowym.	1
4	W	Wzorce projektowe GoF.	4
5	W	Zasady programowania obiektowego SOLID.	1
6	W	Testy oprogramowania obiektowego.	1
7	W	Refaktoring.	1

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Konsultacje przedmiotowe w bezpośrednim kontakcie z prowadzącym	15

2	Zaliczenie w bezpośrednim kontakcie z prowadzącym	10
3	Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury	23

Metody dydaktyczne: konsultacje, wykład, laboratorium komputerowe

Sposoby weryfikacji i oceny efektów uczenia się: uczestnictwo w zajęciach, odpowiedź ustna, projekt

Kryteria oceny:

1. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji kompetencji społecznych.
2. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji umiejętności.
3. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji wiedzy.
4. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji wiedzy.

Literatura:

obowiązkowa:

1. Gamma E. et al Design Patterns: Elements of Reusable Object-Oriented Software, 1994, Addison-Wesley

zalecana/fakultatywna:

1. Ockerman Stephanie et al. Mastering Professional Scrum: Coaches Notes for Busting Myths, Solving Challenges, and Growing Agility, Upper Saddle River, 2018, Wydawnictwo
2. Fowler Martin Refactoring: Improving the Design of Existing Code, Upper Saddle River, 2018, Wydawnictwo

Testowanie oprogramowania nazwa przedmiotu
Software testing nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Umiejętność pracy w zespole, kompetencje programistyczne, znajomość zasad inżynierii oprogramowania.
2. Znajomość języka angielskiego w stopniu umożliwiającym korzystanie z dokumentacji, studiowanie literatury i uczestnictwo w zajęciach.

Cele przedmiotu:

1. Zapoznanie się z podstawami w zakresie testowania oprogramowania, rodzajami testów, stosowanymi narzędziami i wykorzystanie zdobytej wiedzy w praktyce, np. podczas realizacji projektu biznesowego.
2. Nabycie kompetencji miękkich, czyli umiejętność rozwiązywania problemów, kreatywnego i analitycznego myślenia, współpracy w zespole.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	Student ma wiedzę w zakresie podstaw testowania i zapewniania jakości oprogramowania, technik projektowania testów. Rozumie potrzebę tworzenia scenariuszy testowych, zarządzania i dokumentacji procesu testowania.	I1_W10
Umiejętności Absolwent potrafi:		
EU1	Student potrafi projektować, implementować i realizować efektywne przypadki testowe, opracować plan testów, zautomatyzować testy, skonfigurować środowisko testowe, dobrać odpowiednie narzędzia.	I1_U22

Kompetencje społeczne		
Absolwent jest gotów do:		
EK1	Student ma potrzebę efektywnej komunikacji z współpracownikami i klientem oraz świadomość ciągłego samodoskonalenia się w zakresie testowania.	I1_K01

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
6	4	E	18	0	0	18	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	Wprowadzenie do testowania oprogramowania. Testowanie w cyklu życia oprogramowania. Podstawowe narzędzia developerskie testera. Rodzaje testów według funkcjonalności i celów, poziomy testów (E2E, integracyjne, jednostkowe). Poznanie v-modelu. Testy manualne, automatyczne. Testowanie w zespołach zwinnych (Agile). Scenariusze testowe, zarządzanie i dokumentacja procesu testowania. Strategie testowania, projektowanie testów, konfiguracja lokalnego środowiska z wykorzystaniem, np. maszyn wirtualnych i kontenerów, tworzenie przypadków testowych. System kontroli wersji GIT, raportowanie błędów.	18
2	LK	Testowanie aplikacji internetowych z wykorzystaniem, np. Selenium. Testowanie API. Testowanie aplikacji mobilnych iOS/Android. Nowe trendy w testowaniu. Kultura DevOps. Dobre praktyki. Planowanie procesu testowania, zautomatyzowanie testów, skonfigurowanie środowiska testowego, dobranie odpowiednich narzędzi.	14
3	LK	Efektywna komunikacja w grupie i klientem oraz świadomość ciągłego doskonalenia się w zakresie testowania. Zasady współpracy, negocjacji i analiza/zarządzanie ryzykiem.	4

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Przygotowanie się do zajęć i dyskusji, w tym studiowanie zalecanej literatury, opracowanie wyników przeprowadzonych badań, przygotowanie raportu, projektu, prezentacji itp.	45
2	Konsultacje przedmiotowe (w bezpośrednim kontakcie z prowadzącym)	14

3	Egzaminy i zaliczenia (w bezpośrednim kontakcie z prowadzącym)	5
---	--	---

Metody dydaktyczne:

konsultacje, praca w grupach, projekt, wykład, laboratorium komputerowe, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, dyskusja, egzamin pisemny, prezentacja, projekt

Kryteria oceny:

1. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji kompetencji społecznych.
2. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji umiejętności.
3. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji wiedzy.

Literatura:

obowiązkowa:

1. Karolina Zmitrowicz, Adam Roman. Testowanie oprogramowania w praktyce. Studium przypadków 2.0 (ebook)
2. Mateusz Boguszewski. Testowanie automatyczne w .NET

zalecana/fakultatywna:

1. Adam Roman. Testowanie i jakość oprogramowania. PWN 2022

Wprowadzenie do analizy danych nazwa przedmiotu
Introduction to data analysis nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Student zna podstawowe pojęcia rachunku prawdopodobieństwa
2. Student zna podstawowe pojęcia statystyki

Cele przedmiotu:

1. Zapoznanie studentów z podstawowymi metodami analizy danych

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	Zna podstawowe metody, techniki i narzędzia matematyczne, statystyczne i komputerowe do opisu i analizy danych oraz zjawisk masowych.	I1_W01
EW1	Zna podstawowe metody, techniki i narzędzia matematyczne, statystyczne i komputerowe do opisu i analizy danych oraz zjawisk masowych.	I1_W09
Umiejętności Absolwent potrafi:		
EU1	Potrafi prowadzić proste wnioskowania statystyczne, także z wykorzystaniem narzędzi komputerowych; umie wykonywać analizy statystyczne danych. Potrafi posługiwać się narzędziem komputerowym (pakiet R) celem wygenerowania i prezentacji podstawowych rezultatów statystycznych. Potrafi prezentować i wyjaśniać wyniki analiz, eksperymentów i symulacji w postaci wzorów, tabel, wykresów, schematów.	I1_U06b

EU1	Potrafi prowadzić proste wnioskowania statystyczne, także z wykorzystaniem narzędzi komputerowych; umie wykonywać analizy statystyczne danych. Potrafi posługiwać się narzędziem komputerowym (pakiet R) celem wygenerowania i prezentacji podstawowych rezultatów statystycznych. Potrafi prezentować i wyjaśniać wyniki analiz, eksperymentów i symulacji w postaci wzorów, tabel, wykresów, schematów.	I1_U07b
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Student potrafi pracować w zespole, realizować zadania w projektach interdyscyplinarnych.	I1_K04

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
4	5	Z	18	0	0	18	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	Statystyka opisowa (powtórka). Miary położenia i rozproszenia. Skale pomiarowe i ich charakterystyka. Metody graficzne prezentacji danych jakościowych (wykresy kołowe i słupkowe) oraz ilościowych skalarnych (diagramy i histogramy).	2
2	W	Graficzna prezentacja rozkładów zmiennych. Statystyki próbkowe i ich własności, gęstość rozkładu obserwowanej cechy, gęstości normalne i ich własności. Transformacja normalizacyjna. Metody Jackknife i Bootstrap	4
3	W	Zastosowania wnioskowania statystycznego. Praktyczne aspekty obliczeniowe estymacji parametrów, podstawowe przedziały ufności i parametryczne testy istotności	6
4	W	Regresja prosta 1. Definicja modelu, estymacja parametrów, dobór zmiennych. Wykresy rozproszenia, współczynnik korelacji próbkowej. Model zależności liniowej, estymatory współczynników teoretycznej prostej regresji wyznaczone metodą najmniejszych kwadratów oraz ich własności. Współczynnik determinacji.	2
5	W	Regresja prosta 2. Wnioskowanie statystyczne dla modelu regresji prostej. Testy istotności i przedziały ufności dla współczynników regresji liniowej. Prognoza przyszłej wartości zmiennej objaśnianej, błąd prognozy, analiza wartości resztowych.	4
6	LK	Wstęp do programowania w R (powtórka) Pakiety i funkcje programu R .	2

7	LK	Metody graficzne prezentacji danych w R. Podsumowanie danych. Funkcje <code>summary()</code> , <code>boxplot()</code> , <code>hist()</code>	2
8	LK	Generowanie danych z wybranych rozkładów. Graficzne metody sprawdzania dopasowania rozkładu dla rozkładu normalnego	2
9	LK	Generowanie dystrybucyj empirycznych	2
10	LK	Projekt 1	2
11	LK	Bootstrap, przedziały ufności	2
12	LK	Testowania hipotez. Funkcja <code>t.test()</code>	2
13	LK	Regresja liniowa w R. Funkcja <code>lm()</code>	2
14	LK	Projekt 2	2

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Opracowanie wyników	25
2	Przygotowanie raportu, projektu, prezentacji, dyskusji	25
3	Egzaminy i zaliczenia w sesji (w bezpośrednim kontakcie z prowadzącym)	5
4	Konsultacje przedmiotowe (w bezpośrednim kontakcie z prowadzącym)	10
5	Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury	24

Metody dydaktyczne:

konsultacje, projekt, wykład, laboratorium komputerowe, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, kolokwium, projekt, sprawozdanie z pracy indywidualnej, praca własna-zadanie, quiz, test, rysunek

Kryteria oceny:

1. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji wiedzy.
2. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji umiejętności.
3. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji kompetencji społecznych.

Literatura:

obowiązkowa:

1. Tibshirani, James, Witten, Hastie, An Introduction to Statistical Learning with Applications in R, 2013, Springer

zalecana/fakultatywna:

1. J.J. Faraway, Linear Models with R, 2005, CRC Press

Wstęp do informatyki nazwa przedmiotu
Introduction to computer science nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Wiedza z matematyki i informatyki na poziomie szkoły średniej.

Cele przedmiotu:

1. Zapoznanie studentów z budową i działaniem komputera.
2. Przedstawienie sposobów reprezentacji liczb w komputerze.
3. Zapoznanie ze sposobami reprezentacji informacji w komputerze.
4. Nabycie wiedzy w zakresie algorytmów i sposobów ich reprezentacji.
5. Przedstawienie systemów operacyjnych.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		

EW1	Student potrafi posługiwać się fachową terminologią; opisuje budowę komputera i objaśnia zasady jego działania; opisuje zadania i elementy systemu operacyjnego.	I1_W03
EW1	Student potrafi posługiwać się fachową terminologią; opisuje budowę komputera i objaśnia zasady jego działania; opisuje zadania i elementy systemu operacyjnego.	I1_W05
EW2	Student posiada ogólną wiedzę w zakresie algorytmów i sposobów ich reprezentacji.	I1_W04
EW3	Student zna sposoby reprezentacji informacji w komputerze.	I1_W03
EW3	Student zna sposoby reprezentacji informacji w komputerze.	I1_W05
Umiejętności Absolwent potrafi:		
EU1	Student potrafi narysować schemat blokowy lub podać pseudokod podanego algorytmu oraz określić złożoność obliczeniową, a także poprawność prostych algorytmów.	I1_U10
EU1	Student potrafi narysować schemat blokowy lub podać pseudokod podanego algorytmu oraz określić złożoność obliczeniową, a także poprawność prostych algorytmów.	I1_U18
EU2	Student poprawie posługuje się różnorodnymi systemami liczbowymi.	I1_U10
EU2	Student poprawie posługuje się różnorodnymi systemami liczbowymi.	I1_U18
EU3	Student umie wykonywać podstawowe polecenia w systemie Linux. Student potrafi wykorzystać system dla potrzeb skompilowania, zainstalowania i wykonania programu.	I1_U11
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Student potrafi współdzielić dostępne zasoby systemu komputerowego.	I1_K03
EK1	Student potrafi współdzielić dostępne zasoby systemu komputerowego.	I1_K04
EK1	Student potrafi współdzielić dostępne zasoby systemu komputerowego.	I1_K05

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
1	6	E	18	0	0	18	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	LK	Linux a DOS - wykonanie programu w różnych środowiskach, etapy powstawania programu. Algorytmy, schematy blokowe. Typy zmiennych.	1
2	LK	Wyrażenia arytmetyczne, logiczne (pole pod wykresem), schematy blokowe: $\max(a,b)$, $\max(a,b,c)$, $\max\{a_n\}$ - bez tablicy, strukturalne petle, różne warunki stopu, algorytm Euklidesa, optymalizacja algorytmów schemat Hornera, obliczanie wartości funkcji na podstawie rozwinięcia w szereg.	1
3	LK	Układy liczbowe (2, 4, 8, 16, 3, 7..), bramki logiczne, półsumator, sumator bitowy, arytmetyka dwójkowa.	1
4	LK	Reprezentacja liczb w komputerze: liczby całkowite: ZM, U1, U2; liczby zmiennoprzecinkowe, reprezentacja znaków KOD ASCII.	1
5	LK	Bitowe operacje logiczne, kod Haminga; reprezentacja tablic, przekazywanie parametrów do funkcji.	1
6	LK	Odwrotna notacja polska, maszyna Turinga. Przykładowy język maszynowy hipotetycznej maszyny.	1
7	LK	Rekurencja.	1
8	LK	Notacja BNF, diagramy składniowe -> drzewa składniowe - wyrażenia.	1
9	LK	Systemu Linux struktura katalogów, ułatwienia bash?a (TAB, strzałki, Ctrl+c, Ctrl+d,), ls l, polecenia na plikach, katalogach, potoki, przekierowania, edytor Nano, Vim/Vi. Sewer Torus. Zmienne powłoki bash: PATH, HOME, PS1, PS2, HISTFILE,. pliki startowe powłoki bash, kompilacja, wykonanie programu w C; różne środowiska programistyczne. polecenia linuxa: date, time, cal, finger, ps, top, who, chmod, umask, cmp, diff, linki (ln, ln-s), system plikowy ext4.	1
10	LK	Zapis wyrażen arytmetycznych i wyrażen logicznych przestrzen pod/nad wykresami, „++?, *=&, ?> - przykłady. Strukturalne petle, testowanie algorytmów z cwiczen, zamiana liczby z systemu (10) na (2) z ułamkiem.	1
11	LK	Funkcje silnia - badanie zakresów dla typów double, float, int, rozwinięcia w szereg wybranych funkcji matematycznych, implementacja schematu Hornera, wykorzystanie go do zamiany liczby z systemu (16) na system (10). Przekazywanie parametrów do funkcji organizacja pamięci, wskaźniki jako argumenty funkcji, rekurencja a iteracje (silnia, wyszukiwanie binarne, zamiana liczb na system (2) i inne).	1
12	LK	Interfejs korzystania z plików dyskowych, zapis sformatowany i binarny, funkcje niskopoziomowe i wysokopoziomowe. Parametry funkcji main	1
13	LK	System plikowy /proc, prawa dostępu do plików i katalogów wymagane do wykonania poleceń linuxowych. Wyszukiwanie w systemie plikowym polecenia grep, find, tworzenie aliasów.	3
14	LK	Polecenia dotyczące procesów, procesy tła, tworzenie archiwum oraz bibliotek programy tar, ar. Program make, pliki Makefile, skrypty powłoki bash.	2
15	LK	System Windows - komendy wykonywane przez interpreter powłoki, pliki wsadowe (bat), WSL (Windows Subsystem for Linux), różne środowiska programistyczne w środowisku Windows.	1
16	W	Historia informatyki - pierwsze algorytmy - przyrządy i urządzenia wspomagające obliczenia - generacje komputerów. Podstawy teorii informacji.	2

17	W	Systemy liczbowe. Arytmetyka komputerowa. Reprezentacja liczb w komputerze: - stałe całkowite i rzeczywiste - reprezentacje stało- i zmiennopozycyjne - kodowanie ZM, U1, U2 - pojęcia zakresu i błędu zaokrągleń.	4
18	W	Budowa i działanie komputera - model komputera von Neumanna - wykonanie programu - rozkazy arytmetyczne i logiczne - rodzaje pamięci. Podstawowe architektury komputerowe. Automat (bez stosu i ze stosem). Maszyna Turinga.	3
19	W	Oprogramowanie - klasyfikacja oprogramowania - ewolucja systemów operacyjnych - elementy systemu operacyjnego.	2
20	W	Języki programowania - klasyfikacja języków programowania - pojęcia składni i semantyki - pojęcia kompilacji i translacji. Gramatyka (kontaktowa i bezkontaktowa).	2
21	W	Algorytmika - pojęcie algorytmu - cechy algorytmu - sposoby reprezentacji. Schematy blokowe przykładowych algorytmów.	2
22	W	Rekursja. Miary złożoności algorytmów. Problem stopu i problemy(N)P zupełne. Poprawność algorytmów.	2
23	W	Podstawy technologii blockchain.	1

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Egzaminy i zaliczenia (w bezpośrednim kontakcie z prowadzącym)	14
2	Konsultacje przedmiotowe (w bezpośrednim kontakcie z prowadzącym)	10
3	Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury	40
4	Opracowanie wyników	30
5	Przygotowanie raportu, projektu, prezentacji, dyskusji	20

Metody dydaktyczne:

ćwiczenia, ćwiczenia laboratoryjne, dyskusja, konsultacje, praca w grupach, wykład, ćwiczenia tablicowe, laboratorium komputerowe

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, dyskusja, egzamin pisemny, kolokwium, odpowiedź ustna, test, zaliczenie pisemne, zaliczenie ustne, kolokwium z zadań

Kryteria oceny:

1. Na ocenę 2.0 Student nie spełnia warunków określonych dla oceny 3.0. Na ocenę 3.0 Uzyskanie co najmniej 51% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 3.5 Uzyskanie co najmniej 61% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 4.0 Uzyskanie co najmniej 71% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 4.5 Uzyskanie co najmniej 81% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 5.0 Uzyskanie co najmniej 91% punktów w ramach sposobów weryfikacji nabytych kompetencji.
2. Na ocenę 2.0 Student nie spełnia warunków określonych dla oceny 3.0. Na ocenę 3.0 Uzyskanie co najmniej 51% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 3.5 Uzyskanie co najmniej 61% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 4.0 Uzyskanie co najmniej 71% punktów w ramach sposobów

- weryfikacji nabytych kompetencji. Na ocenę 4.5 Uzyskanie co najmniej 81% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 5.0 Uzyskanie co najmniej 91% punktów w ramach sposobów weryfikacji nabytych kompetencji.
3. Na ocenę 2.0 Student nie spełnia warunków określonych dla oceny 3.0. Na ocenę 3.0 Uzyskanie co najmniej 51% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 3.5 Uzyskanie co najmniej 61% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 4.0 Uzyskanie co najmniej 71% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 4.5 Uzyskanie co najmniej 81% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 5.0 Uzyskanie co najmniej 91% punktów w ramach sposobów weryfikacji nabytych kompetencji.
 4. Na ocenę 2.0 Student nie spełnia warunków określonych dla oceny 3.0. Na ocenę 3.0 Uzyskanie co najmniej 51% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 3.5 Uzyskanie co najmniej 61% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 4.0 Uzyskanie co najmniej 71% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 4.5 Uzyskanie co najmniej 81% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 5.0 Uzyskanie co najmniej 91% punktów w ramach sposobów weryfikacji nabytych kompetencji.
 5. Na ocenę 2.0 Student nie spełnia warunków określonych dla oceny 3.0. Na ocenę 3.0 Uzyskanie co najmniej 51% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 3.5 Uzyskanie co najmniej 61% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 4.0 Uzyskanie co najmniej 71% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 4.5 Uzyskanie co najmniej 81% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 5.0 Uzyskanie co najmniej 91% punktów w ramach sposobów weryfikacji nabytych kompetencji.
 6. Na ocenę 2.0 Student nie spełnia warunków określonych dla oceny 3.0. Na ocenę 3.0 Uzyskanie co najmniej 51% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 3.5 Uzyskanie co najmniej 61% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 4.0 Uzyskanie co najmniej 71% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 4.5 Uzyskanie co najmniej 81% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 5.0 Uzyskanie co najmniej 91% punktów w ramach sposobów weryfikacji nabytych kompetencji.
 7. Na ocenę 2.0 Student nie spełnia warunków określonych dla oceny 3.0. Na ocenę 3.0 Uzyskanie co najmniej 51% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 3.5 Uzyskanie co najmniej 61% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 4.0 Uzyskanie co najmniej 71% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 4.5 Uzyskanie co najmniej 81% punktów w ramach sposobów weryfikacji nabytych kompetencji. Na ocenę 5.0 Uzyskanie co najmniej 91% punktów w ramach sposobów weryfikacji nabytych kompetencji.

Literatura:

obowiązkowa:

1. J.Glenn Brookshear, Informatyka w ogólnym zarysie, Warszawa, 2003, WNT
2. R. Kawa, J. Lembas, Wstęp do informatyki, 2017, PWN

zalecana/fakultatywna:

1. B.S.Chalk, Organizacja i architektura komputerów, Warszawa, 1998, WNT
2. E. Nemeth, G. Snyder, T. R. Hein, B. Whaley, D. Mackin, Unix i Linux. Przewodnik administratora systemów, Warszawa, 2018, Hellion
3. N.Wirth , Algorytmy + struktury danych = programy, 1989, WNT

4. E.W. Dijkstra. Umiejętność programowania, Warszawa, 1985, WNT
5. R.Tadeusiewicz, P.Moszner, A.Szydelko, Teoretyczne podstawy informatyki, Kraków, 1998, WSP

Wstęp do projektowania aplikacji internetowych nazwa przedmiotu
Introduction to web application design nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Znajomość minimum jednego obiektowego języka programowania.
2. Podstawowa wiedza z zakresu projektowania baz danych, w tym znajomość podstawowych komend języka SQL.
3. Podstawowa wiedza z zakresu użytkowania systemu operacyjnego Linux.
4. Podstawowa wiedza dotycząca systemu kontroli wersji GIT.

Cele przedmiotu:

1. Przedmiot ma na celu pokazać jak powstają aplikacje internetowe, jaka jest ich droga od idei przez implementacje, testy aż po wdrożenie.
2. Celem przedmiotu jest zapoznanie studentów z językiem programowania umożliwiającym tworzenie aplikacji internetowych po stronie serwera takim jak PHP oraz technologiami Web Core pozwalającymi na tworzenie aplikacji po stronie klienta (HTML, CSS, JavaScript) wraz z bazą danych PostgreSQL.
3. Celem przedmiotu jest nauczenie studentów tworzenia użytkowych aplikacji internetowych, w tym zwrócenie szczególnej uwagi na bezpieczeństwo, jak również problemy wynikające z

dostępu do danych przez wielu użytkowników w jednakowym czasie.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	Student rozumie potrzebę ciągłego dokształcania się zawodowego i rozwoju osobistego. Dokonuje samooceny własnych kompetencji, wyznacza kierunki własnego rozwoju i kształcenia. Samodzielnie podejmuje refleksje dotyczące etyki w odniesieniu do wykonywanej pracy.	I1_W10
Umiejętności Absolwent potrafi:		
EU1	Umiejętność doboru technologii do wymagań aplikacji. Projektowanie prototypów aplikacji w oparciu o narzędzia Adobe Xd oraz Figma. Rozwój umiejętności pracy nad projektem z wykorzystaniem repozytorium git. Umiejętność tworzenia od podstaw użytkowych aplikacji internetowych z uwzględnieniem reguł bezpieczeństwa w szczególności dotyczących nieautoryzowanego dostępu do danych, walidacji danych, autentykacji użytkownika, hashowanie hasła.	I1_U01b
EU1	Umiejętność doboru technologii do wymagań aplikacji. Projektowanie prototypów aplikacji w oparciu o narzędzia Adobe Xd oraz Figma. Rozwój umiejętności pracy nad projektem z wykorzystaniem repozytorium git. Umiejętność tworzenia od podstaw użytkowych aplikacji internetowych z uwzględnieniem reguł bezpieczeństwa w szczególności dotyczących nieautoryzowanego dostępu do danych, walidacji danych, autentykacji użytkownika, hashowanie hasła.	I1_U08
EU1	Umiejętność doboru technologii do wymagań aplikacji. Projektowanie prototypów aplikacji w oparciu o narzędzia Adobe Xd oraz Figma. Rozwój umiejętności pracy nad projektem z wykorzystaniem repozytorium git. Umiejętność tworzenia od podstaw użytkowych aplikacji internetowych z uwzględnieniem reguł bezpieczeństwa w szczególności dotyczących nieautoryzowanego dostępu do danych, walidacji danych, autentykacji użytkownika, hashowanie hasła.	I1_U16

EU1	Umiejętność doboru technologii do wymagań aplikacji. Projektowanie prototypów aplikacji w oparciu o narzędzia Adobe Xd oraz Figma. Rozwój umiejętności pracy nad projektem z wykorzystaniem repozytorium git. Umiejętność tworzenia od podstaw użytkowych aplikacji internetowych z uwzględnieniem reguł bezpieczeństwa w szczególności dotyczących nieautoryzowanego dostępu do danych, walidacji danych, autentykacji użytkownika, hashowanie hasła.	I1_U21
EU1	Umiejętność doboru technologii do wymagań aplikacji. Projektowanie prototypów aplikacji w oparciu o narzędzia Adobe Xd oraz Figma. Rozwój umiejętności pracy nad projektem z wykorzystaniem repozytorium git. Umiejętność tworzenia od podstaw użytkowych aplikacji internetowych z uwzględnieniem reguł bezpieczeństwa w szczególności dotyczących nieautoryzowanego dostępu do danych, walidacji danych, autentykacji użytkownika, hashowanie hasła.	I1_U23
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Student rozumie potrzebę ciągłego dokształcania się zawodowego i rozwoju osobistego. Dokonuje samooceny własnych kompetencji, wyznacza kierunki własnego rozwoju i kształcenia. Samodzielnie podejmuje refleksje dotyczące etyki w odniesieniu do wykonywanej pracy.	I1_K03

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
5	3	Z	18	0	0	18	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	Wprowadzenie do przedmiotu. Przedstawienie wybranych w ramach przedmiotu technologii (w tym języków programowania). Zasady zaliczenia przedmiotu. Przedstawienie procesu powstawania aplikacji internetowej. Praca z repozytorium git w trakcie rozwoju projektu. Wirtualizacja na poziomie systemu operacyjnego w celu utworzenia serwera. Jak zaprojektować aplikację internetową. Wprowadzenie do procesu prototypowania. Określenie wymagań funkcjonalnych aplikacji. Wykorzystanie elementów języka HTML5 w projektowaniu aplikacji internetowych.	3

2	W	Kaskadowe Arkusze Stylów (CSS). Omówienie najważniejszych elementów CSS (w szczególności wszystkich rodzajów selektorów, tj. m.in.: klas, relacji rodzic-dziecko-sąsiedzi pomiędzy elementami HTML, deklaracji stylów dla urządzeń mobilnych, drukarek, itd.). Używanie bibliotek Google Fonts i Fontawesome. Kaskadowe Arkusze Stylów (CSS). Tworzenie layoutów w oparciu o grid i flex. Budowanie responsywnego layoutu na urządzenia mobilne.	3
3	W	Wstęp do języka PHP. Omówienie najważniejszych elementów języka (tj. warunków, petli, itd.). Wysyłanie i odbieranie formularzy z danymi, omówienie zagrożeń wynikających z przesyłania danych na serwer. Wstęp do klas i tworzenia obiektów. Omówienie protokołu HTTP oraz zagadnień bezpieczeństwa autoryzacji i autentykacji użytkownika. Język PHP. Tworzenie szkieletu własnego MVC. Oprogramowanie wyjątków. Logowanie do własnego systemu, utrzymanie sesji, wylogowanie. Mechanizm sesji i ciasteczek. Bezpieczeństwo logowania, omówienie możliwych ataków (np. brute force, SQL injection, kradzież sesji) i metody ich zapobiegania.	3
4	W	Elementy baz danych w oparciu o bazę PostgreSQL. Przypomnienie najważniejszych informacji dotyczących baz danych (język SQL, postaci normalne, anomalie, relacje, widoki, unie, indeksy, itd.), projektowania poprawnych schematów baz danych na przykładach w odniesieniu do projektowania użytkowych aplikacji internetowych.	3
5	W	Elementy baz danych w oparciu o bazę PostgreSQL. Tworzenie połączenia serwera z bazą danych, pobranie danych. Tworzenie wzorca Data Access Object oraz zapytań do wyciągnięcia danych z bazy na serwer. Wyświetlenie danych po stronie klienta w języku PHP.	3
6	W	Wstęp do języka JavaScript. Omówienie najważniejszych funkcji i elementów języka w aspekcie projektowania użytkowych aplikacji internetowych. Przykłady zastosowania JavaScript Fetch API. Omówienie sposobów wysyłania zapytań za pomocą języka JavaScript. Wprowadzenie do obiektów Promise, notacji JSON. Praca z API.	3
7	LK	Utworzenie wirtualnego kontenera z serwerem NGINX. Stworzenie szkieletu projektu w języku PHP oraz opublikowanie kodu na repozytorium GIT. Prototypowanie aplikacji w Adobe XD.	2
8	LK	Tworzenie pierwszych modułów składowych z wykorzystaniem języków HTML i CSS. Tworzenie layoutu aplikacji w oparciu o grid i flex. Responsywność strony z wykorzystaniem media queries.	3
9	LK	Proste struktury aplikacji wykorzystujące funkcje, pętle, warunki, itd. Tworzenie routingu aplikacji w PHP. Stworzenie struktury projektu w oparciu o filary obiektowości i wzorzec MVC. Autentykacja i autoryzacja użytkownika. Wysyłanie formularzy na serwer.	4
10	LK	Przesyłanie plików na serwer.	2
11	LK	Projekt schematu bazy danych do aplikacji. Ustawienie połączenia z bazą danych z poziomu języka PHP w oparciu o bazę w chmurze. Tworzenie tabel (w tym dobór odpowiednich silników do tabel). Tworzenie referencji z odpowiednimi akcjami na referencjach. Pobieranie danych z bazy w oparciu o wzorzec DAO, repozytoria z zapytaniami, operacje wstawiania, usuwania, odczytu i aktualizacji danych przez zapytania wysłane z serwera.	3

12	LK	Oprogramowanie modułów składowych aplikacji za pomocą języka JavaScript. Manipulacja DOM, dynamiczne tworzenie elementów HTML oraz korzystanie z szablonów. Walidacja formularzy w JavaScript. JavaScript Fetch API. Asynchroniczne wyszukiwanie danych. Oddawanie projektów indywidualnych przez studentów. Uwagi prowadzącego, poprawki studentów. Wystawianie ocen końcowych. (c.d. na konsultacjach).	4
----	----	---	---

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury.	10
2	Przygotowanie raportu, projektu, prezentacji, dyskusji	20
3	Zaliczenie w bezpośrednim kontakcie z prowadzącym	10
4	Konsultacje przedmiotowe w bezpośrednim kontakcie z prowadzącym	14

Metody dydaktyczne:

dyskusja, gra dydaktyczna, konsultacje, praca z tekstem/materiałem audio i wideo, projekt, wykład, laboratorium komputerowe, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, kolokwium, odpowiedź ustna, prezentacja, projekt, sprawozdanie z pracy indywidualnej, praca własna-zadanie, quiz, test, rysunek

Kryteria oceny:

1. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji umiejętności.
2. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji umiejętności.
3. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji umiejętności.
4. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji umiejętności.
5. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji wiedzy.
6. Na ocenę 5.0 Student uzyskuje od 91% do 100% punktów z wybranych form weryfikacji kompetencji społecznych.

Literatura:

obowiązkowa:

1. Duckett J. - HTML i CSS. Zaprojektuj i zbuduj witrynę WWW. Podręcznik Front End Developera, Gliwice, 2014, Helion
2. Duckett J. - JavaScript Oficjalna Dokumentacja, <https://developer.mozilla.org/en-US/docs/Web/JavaScript>
3. Frain B. - Responsive Web Design. Projektowanie elastycznych witryn w HTML5 i CSS3, Gliwice, 2014, Helion
4. Lis M. - PHP 7, Gliwice, 2017, Helion

5. Lavin, Peter - PHP : programowanie obiektowe, Gliwice, 2007, Helion
6. Lemay L., Colburn R., Kyrnin J. - HTML, CSS i JavaScript, Gliwice, 2017, Helion
7. Nicholas C. Z. - ECMAScript 6. Przewodnik po nowym standardzie języka JavaScript, Gliwice, 2017, Helion
8. Robbins J. - Projektowanie stron internetowych. Przewodnik dla początkujących webmasterów po HTML5, CSS3 i grafice, Gliwice, 2014, Helion
9. Worsley, John C; Drake, Joshua D. - PostgreSQL : praktyczny przewodnik, Gliwice, 2002, Helion
10. Dybikowski, Zdzisław. - PostgreSQL, Gliwice, 2012, Helion

zalecana/fakultatywna:

1. Borycki D. - JavaScript i jQuery, Gliwice, 2014, Helion
2. Danowski B. - Wstęp do CSS3 i HTML5, Gliwice, 2011, Helion
3. Henderson C. - Skalowalne witryny internetowe, Gliwice, 2006, O'Reilly
4. Hogan B. - HTML5 i CSS3. Standardy przyszłości, Gliwice, 2011, Helion
5. PHP manual, www.php.net
6. W3S manual, www.w3schools.com

Wstęp do sztucznej inteligencji nazwa przedmiotu
Introduction to Artificial Intelligence nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Podstawowa wiedza z zakresu analizy matematycznej (dotycząca m.in. rachunku różniczkowego).

2. Podstawowa wiedza z zakresu programowania, umiejętność projektowania i implementacji algorytmów i prostych struktur danych.

Cele przedmiotu:

1. Opanowanie przez studentów podstawowych pojęć związanych z wybranymi technikami i systemami inteligencji maszynowej.
2. Zrozumienie przez studentów problemu reprezentacji wiedzy w dziedzinie sztucznej inteligencji oraz zapoznanie się z wybranymi metodami reprezentacji wiedzy i podstawowymi algorytmami wnioskowania.
3. Opanowanie przez studentów podstawowych narzędzi używanych w sztucznej inteligencji, w tym sztucznych sieci neuronowych, algorytmów ewolucyjnych oraz wnioskowania rozmytego, umiejętność pracy z tymi narzędziami i implementacji własnych algorytmów.
4. Przedstawienie studentom współczesnych kierunków rozwoju metod sztucznej inteligencji, umiejętność wyciągania wniosków przez studentów i formułowania własnych tez.

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
EW1	Ugruntowana wiedza w zakresie posługiwania się podstawowymi pojęciami oraz w zakresie problematyki badawczej w dziedzinie sztucznej inteligencji.	I1_W04
EW1	Ugruntowana wiedza w zakresie posługiwania się podstawowymi pojęciami oraz w zakresie problematyki badawczej w dziedzinie sztucznej inteligencji.	I1_W06
EW1	Ugruntowana wiedza w zakresie posługiwania się podstawowymi pojęciami oraz w zakresie problematyki badawczej w dziedzinie sztucznej inteligencji.	I1_W08
EW2	Ugruntowana wiedza w zakresie metod reprezentowania wiedzy oraz podstawowych modeli, technik i algorytmów z dziedziny sztucznej inteligencji.	I1_W04
EW2	Ugruntowana wiedza w zakresie metod reprezentowania wiedzy oraz podstawowych modeli, technik i algorytmów z dziedziny sztucznej inteligencji.	I1_W08
Umiejętności Absolwent potrafi:		
EU1	Umiejętność posługiwania się wybranymi narzędziami i bibliotekami (frameworkami) stosowanymi w implementacji podstawowych metod i algorytmów z dziedziny sztucznej inteligencji.	I1_U01b

EU1	Umiejętność posługiwania się wybranymi narzędziami i bibliotekami (frameworkami) stosowanymi w implementacji podstawowych metod i algorytmów z dziedziny sztucznej inteligencji.	I1_U06b
EU1	Umiejętność posługiwania się wybranymi narzędziami i bibliotekami (frameworkami) stosowanymi w implementacji podstawowych metod i algorytmów z dziedziny sztucznej inteligencji.	I1_U07b
EU1	Umiejętność posługiwania się wybranymi narzędziami i bibliotekami (frameworkami) stosowanymi w implementacji podstawowych metod i algorytmów z dziedziny sztucznej inteligencji.	I1_U18
EU2	Umiejętność walidacji wykonanego oprogramowania na danych testowych, opracowania wyników testów oprogramowania i ich interpretacji.	I1_U01b
EU2	Umiejętność walidacji wykonanego oprogramowania na danych testowych, opracowania wyników testów oprogramowania i ich interpretacji.	I1_U06b
EU2	Umiejętność walidacji wykonanego oprogramowania na danych testowych, opracowania wyników testów oprogramowania i ich interpretacji.	I1_U07b
EU2	Umiejętność walidacji wykonanego oprogramowania na danych testowych, opracowania wyników testów oprogramowania i ich interpretacji.	I1_U10
EU2	Umiejętność walidacji wykonanego oprogramowania na danych testowych, opracowania wyników testów oprogramowania i ich interpretacji.	I1_U14
EU2	Umiejętność walidacji wykonanego oprogramowania na danych testowych, opracowania wyników testów oprogramowania i ich interpretacji.	I1_U18
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Umiejętność pracy w grupie, pracy indywidualnej, samokształcenie, umiejętność komunikacji z nauczycielem i środowiskiem pozauczelnianym w celu popularyzacji przedstawiania uzyskanych rezultatów w zrozumiały sposób.	I1_K02
EK1	Umiejętność pracy w grupie, pracy indywidualnej, samokształcenie, umiejętność komunikacji z nauczycielem i środowiskiem pozauczelnianym w celu popularyzacji przedstawiania uzyskanych rezultatów w zrozumiały sposób.	I1_K03
EK1	Umiejętność pracy w grupie, pracy indywidualnej, samokształcenie, umiejętność komunikacji z nauczycielem i środowiskiem pozauczelnianym w celu popularyzacji przedstawiania uzyskanych rezultatów w zrozumiały sposób.	I1_K04

EK1	Umiejętność pracy w grupie, pracy indywidualnej, samokształcenie, umiejętność komunikacji z nauczycielem i środowiskiem pozauczelnianym w celu popularyzacji przedstawiania uzyskanych rezultatów w zrozumiały sposób.	I1_K05
-----	--	--------

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
5	4	Z	18	0	0	18	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	Wstęp do sztucznej inteligencji - główne obszary AI.	1
2	W	Podstawy wnioskowania bayesowskiego	1
3	W	Metody wnioskowania rozmytego	1
4	W	Algorytmy genetyczne	1
5	W	Optymalizacyjne heurystyki ewolucyjne	4
6	W	Podstawy sieci neuronowych	2
7	W	Metody uczenia sieci neuronowych	3
8	W	Uczenie głębokie, sieci konwolucyjne	2
9	W	Metody uczenia maszynowego	2
10	W	Kolokwium zaliczeniowe	1
11	LK	Wprowadzenie do języka Python. Pakiety: NumPy, Pandas, Matplotlib	2
12	LK	Wnioskowanie w logice klasycznej	2
13	LK	Wnioskowanie w oparciu o wiedzę niepewną, wnioskowanie Bayesowskie.	1
14	LK	Modelowanie i wnioskowanie w oparciu o pojęcia języka naturalnego (modelowanie rozmyte).	1
15	LK	Heurystyczne algorytmy optymalizacji globalnej. Prosty algorytm genetyczny.	4
16	LK	Sztuczne sieci neuronowe: Perceptron.	2
17	LK	Sztuczne sieci neuronowe: Sieć MLP, algorytm wstecznej propagacji błędów.	3
18	LK	Deep Learning: Przykłady zastosowań uczenia głębokiego.	3

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury	10

2	Implementacja wybranych algorytmów	10
3	Opracowanie wyników, „przygotowanie raportu, projektu, prezentacji, dyskusji	20
4	Konsultacje przedmiotowe (w bezpośrednim kontakcie z prowadzącym) poza terminem zajęć	14
5	Przygotowanie się do kolokwium zaliczeniowego, w tym zaliczenie projektów bezpośrednim kontakcie z prowadzącym	10

Metody dydaktyczne:

analiza przypadku, dyskusja, konsultacje, praca w grupach, projekt, wykład, laboratorium komputerowe, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, dyskusja, egzamin pisemny, kolokwium, odpowiedź ustna, prezentacja, projekt, rozwiązanie zadania problemowego, sprawozdanie z pracy indywidualnej, sprawozdanie z pracy zespołowej, test, zaliczenie ustne, sprawozdanie

Kryteria oceny:

1. Na ocenę 5.0 student zna podstawowe metody realizacji projektu grupowego.
2. Na ocenę 5.0 student umie wykorzystać narzędzia i biblioteki do implementacji metod sztucznej inteligencji w opracowaniu zadania laboratoryjnego
3. Na ocenę 5.0 student umie krytycznie opracować wyniki eksperymentów obliczeniowych, poprawnie waliduje modele na danych testowych
4. Na ocenę 5.0 student zna podstawowe metody i techniki sztucznej inteligencji, zna obszary stosowalności podstawowych metod.
5. Na ocenę 5.0 student posiada ugruntowaną wiedzę dotyczącą modeli, algorytmów i technik z obszaru sztucznej inteligencji

Literatura:

obowiązkowa:

1. Michalewicz Z. Algorytmy genetyczne + struktury danych = programy ewolucyjne, Warszawa, 2003, WNT
2. Russel S., Norvig P. Artificial Intelligence: A Modern Approach, Upper Saddle River, New Jersey, 2002, Prentice Hall
3. Rutkowska D., Pilinski M, Rutkowski L. Sieci neuronowe, algorytmy genetyczne i systemy rozmyte, Warszawa, 1997, PWN

zalecana/fakultatywna:

1. Arabas J. Wykłady z algorytmów ewolucyjnych, Warszawa, 2001, WNT
2. Goldberg D.E. Algorytmy genetyczne i ich zastosowania, Warszawa, 1995, WNT
3. Tadeusiewicz R. Elementarne wprowadzenie do techniki sieci neuronowych z przykładowymi programami, Warszawa, 1999, Akademicka Oficyna Wydawnicza PLJ
4. Piegat A. Modelowanie i sterowanie rozmyte, Warszawa, 2003, Akademicka Oficyna Wydawnicza EXIT

Zaawansowane technologie projektowania aplikacji internetowych nazwa przedmiotu
Advanced technologies of web application development nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty kierunkowe kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. Podstawowa wiedza z zakresu projektowania aplikacji internetowych i języków programowania używanych do ich budowy.
2. Umiejętność programowania obiektowego w jednym z tech języków: PHP, Java lub JavaScript
3. Podstawowa wiedza z zakresu sieci komputerowych i baz danych

Cele przedmiotu:

1. Zapoznanie się z zaawansowanymi technologiami tworzenia aplikacji internetowych
2. Zapoznanie się z zasadami projektowania i implementacji nowoczesnych aplikacji webowych w oparciu o różne frameworki i wzorce projektowe
3. Zrozumienie współczesnych problemów związanych z tworzeniem rozbudowanych aplikacji internetowych

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
------------------------	-------------------------	-------------------------------------

Wiedza Absolwent zna i rozumie:		
EW1	Zna i rozumie podstawowe metody, techniki i narzędzia pozwalające budować złożone aplikacje internetowe. Posiada zaawansowaną wiedzę na temat wyboru optymalnego frameworka do budowy aplikacji webowej o konkretnej charakterystyce.	I1_W06
EW1	Zna i rozumie podstawowe metody, techniki i narzędzia pozwalające budować złożone aplikacje internetowe. Posiada zaawansowaną wiedzę na temat wyboru optymalnego frameworka do budowy aplikacji webowej o konkretnej charakterystyce.	I1_W07
EW1	Zna i rozumie podstawowe metody, techniki i narzędzia pozwalające budować złożone aplikacje internetowe. Posiada zaawansowaną wiedzę na temat wyboru optymalnego frameworka do budowy aplikacji webowej o konkretnej charakterystyce.	I1_W08
Umiejętności Absolwent potrafi:		
EU1	Potrafi przeprowadzić szczegółową analizę wdrożeniową w celu wyboru odpowiedniej technologii dla podanego problemu. Potrafi utworzyć w pełni działającą aplikację webową w oparciu o wybrany framework.	I1_U01b
EU1	Potrafi przeprowadzić szczegółową analizę wdrożeniową w celu wyboru odpowiedniej technologii dla podanego problemu. Potrafi utworzyć w pełni działającą aplikację webową w oparciu o wybrany framework.	I1_U04
EU1	Potrafi przeprowadzić szczegółową analizę wdrożeniową w celu wyboru odpowiedniej technologii dla podanego problemu. Potrafi utworzyć w pełni działającą aplikację webową w oparciu o wybrany framework.	I1_U12
EU1	Potrafi przeprowadzić szczegółową analizę wdrożeniową w celu wyboru odpowiedniej technologii dla podanego problemu. Potrafi utworzyć w pełni działającą aplikację webową w oparciu o wybrany framework.	I1_U17
EU1	Potrafi przeprowadzić szczegółową analizę wdrożeniową w celu wyboru odpowiedniej technologii dla podanego problemu. Potrafi utworzyć w pełni działającą aplikację webową w oparciu o wybrany framework.	I1_U20
EU1	Potrafi przeprowadzić szczegółową analizę wdrożeniową w celu wyboru odpowiedniej technologii dla podanego problemu. Potrafi utworzyć w pełni działającą aplikację webową w oparciu o wybrany framework.	I1_U21
EU1	Potrafi przeprowadzić szczegółową analizę wdrożeniową w celu wyboru odpowiedniej technologii dla podanego problemu. Potrafi utworzyć w pełni działającą aplikację webową w oparciu o wybrany framework.	I1_U23

Kompetencje społeczne		
Absolwent jest gotów do:		
EK1	Umie pracować indywidualnie oraz przekazywać uzyskane rezultaty pracy w zrozumiały sposób, wykorzystując przy tym narzędzia informatyczne do zdalnego kontaktu z nauczycielem.	I1_K03
EK1	Umie pracować indywidualnie oraz przekazywać uzyskane rezultaty pracy w zrozumiały sposób, wykorzystując przy tym narzędzia informatyczne do zdalnego kontaktu z nauczycielem.	I1_K04

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
7	4	Z	18	0	0	0	18	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	W	Organizacja zajęć, omówienie planu wykładu i warunków zaliczenia przedmiotu.	1
2	W	Wprowadzenie do zaawansowanych technologii internetowych.	2
3	W	Wprowadzenie do tworzenia aplikacji przy użyciu Symfony oraz Springa.	2
4	W	Systemy zarządzania pakietami, MVC, routing oraz tworzenie kontrolerów i widoków w Symfony.	2
5	W	Odwzorowanie obiektowej architektury systemu informatycznego na bazę danych wykorzystaniem ORM lub ODM.	2
6	W	Domain-driven Design, REST i CQRS.	2
7	W	Mikroserwis a monolit, praca z API.	2
8	W	Zdarzenia, kontener DI, formularze i kontrola dostępu.	1
9	W	Podział aplikacji na backend i frontend, popularne metody uwierzytelniania dla API.	1
10	W	Asynchroniczność - systemy kolejkowania wiadomości Apache Kafka i RabbitMQ.	1
11	W	Tworzenie testów, przedstawienie możliwych podejść testowania aplikacji.	1
12	W	Skalowanie aplikacji, sposoby efektywnego zarządzania danymi w aplikacjach webowych.	1
13	P	Organizacja zajęć, omówienie zakresu projektu, konfiguracja IDE pod wybrany framework.	1
14	P	Zgłoszenie projektu i stworzenie prototypu aplikacji.	1

15	P	Zarządzanie cyklem życia i budowaniem projektu (Maven, Gradle, Composer).	2
16	P	Routing i kontrolery, adnotacje. Uruchomienie aplikacji, stworzenie pierwszego powitalnego kontrolera.	1
17	P	Połączenie systemu zarządzania bazą danych z aplikacją, stworzenie odpowiedniej konfiguracji.	3
18	P	Wdrożenie usług do komunikacji z urządzeniami klienckimi w oparciu o HTTP REST API.	3
19	P	Bezpieczeństwo: Uwierzytelnienie, Autoryzacja. Zbudowanie tokenu do autoryzacji i transmisji danych w oparciu o JSON Web Token (JWT).	2
20	P	Implementacja zdarzeń, nasłuchiwanie i publikowanie zdarzeń w aplikacji.	2
21	P	Narzędzia do kolejkowania wiadomości.	1
22	P	Projektowanie i implementacja testów.	1
23	P	Oddawanie indywidualnych projektów i ocena pracy studentów.	1

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Konsultacje przedmiotowe (w bezpośrednim kontakcie z prowadzącym)	10
2	Przygotowanie się do zajęć, w tym studiowanie zalecanej literatury	15
3	Opracowanie wyników	15
4	Przygotowanie raportu, projektu, prezentacji, dyskusji	40

Metody dydaktyczne:

dyskusja, konsultacje, projekt, wykład, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, dyskusja, egzamin pisemny, kolokwium, projekt, sprawozdanie z pracy indywidualnej, sprawdzian wiadomości z wykładów, praca własna-zadanie, quiz, test, rysunek

Kryteria oceny:

1. Na ocenę 5.0 student zna i rozumie podstawowe metody, techniki i narzędzia pozwalające budować złożone aplikacje internetowe. Posiada zaawansowaną wiedzę na temat wyboru optymalnego frameworka do budowy aplikacji webowej o konkretnej charakterystyce.
2. Na ocenę 5.0 student potrafi przeprowadzić szczegółową analizę wdrożeniową w celu wyboru odpowiedniej technologii dla podanego problemu. Potrafi utworzyć w pełni działającą aplikację webową w oparciu o wybrany framework.
3. Na ocenę 5.0 student umie pracować indywidualnie oraz przekazywać uzyskane rezultaty pracy w zrozumiały sposób, wykorzystując przy tym narzędzia informatyczne do zdalnego kontaktu z nauczycielem.

Literatura:

obowiązkowa:

1. Robert C. Martin, Czysta architektura. Struktura i design oprogramowania. Przewodnik dla profesjonalistów, 2018, Helion
2. F. Potencier, Symfony 5: Szybki start, 2020, Symfony SAS
3. Craig Walls, Spring w akcji. Wydanie V, 2019, Helion

zalecana/fakultatywna:

1. Vaughn Vernon, DDD dla architektów oprogramowania, 2016, Helion
2. Mariusz Walczak, Tworzenie nowoczesnych systemów webowych, 2016, Helion

Zachowania organizacyjne nazwa przedmiotu
Organizational behavior nazwa przedmiotu w języku angielskim
polski język wykładowy
przedmioty ogólne kategoria przedmiotu/grupa zajęć

Cykl kształcenia rozpoczynający się od: rok akademicki 2025/26, semestr zimowy

Jednostka organizacyjna: Wydział Informatyki i Telekomunikacji

Kierunek studiów: Informatyka Niestacjonarne

Specjalność: -

Profil studiów: ogólnoakademicki

Poziom studiów: I stopnia

Forma studiów: niestacjonarne

Wymagania wstępne:

1. brak wymagań

Cele przedmiotu:

1. Wykształcenie umiejętności budowania efektywnych systemów motywacyjnych dla zwinnych zespołów
2. Wykształcenie postawy przeciwdziałania nieetycznym zachowaniom w kontekście presji na zwiększanie efektywności
3. Ukształtowanie postawy otwartości na zmianę, gotowości do racjonalnego rozwiązywania problemów, podejmowania decyzji również w ramach zespołów międzykulturowych czy

wirtualnych

Efekty uczenia się:

Kod efektu uczenia się	Opis efektu uczenia się	Kod kierunkowego efektu uczenia się
Wiedza Absolwent zna i rozumie:		
BRAK DANYCH	BRAK DANYCH	BRAK DANYCH
Umiejętności Absolwent potrafi:		
EU1	Student potrafi wykorzystać adekwatne koncepcje motywacyjne do projektowania efektywnych zespołów	I1_U05
Kompetencje społeczne Absolwent jest gotów do:		
EK1	Student potrafi przeciwdziałać zachowaniom nieetycznym w organizacji	I1_K02
EK2	Student jest gotów do podejmowania decyzji, rozwiązywania problemów samodzielnie lub w grupie funkcjonując w ramach różnorodności kulturowej /zespołów wirtualnych	I1_K03

Forma zajęć, semestralna liczba godzin:

Semestr	Punkty ECTS	Forma zaliczenia (E/Z)	Wykłady (W)	Ćwiczenia (C)	Laboratoria (L)	Laboratoria komputerowe (LK)	Projekty (P)	Seminaria (S)	Praktyki zawodowe (PZ)
6	1	Z	0	9	0	0	0	0	BRAK DANYCH

E - egzamin; Z - zaliczenie

Treści programowe:

Lp.	Forma zajęć	tematyka zajęć	Liczba godzin
1	C	Uwarunkowania zachowań indywidualnych i grupowych w organizacji	1
2	C	Style kierowania a zachowania organizacyjne	2
3	C	Wybrane metody i techniki motywacji w kształtowaniu zachowań organizacyjnych (jednostki i zespołu)	3
4	C	Konflikty i ich wpływ na zachowania ludzi w organizacjach	2

5	C	Zachowania organizacyjne w sytuacji zmiany	1
---	---	--	---

Praca własna studenta:

Lp.	Opis pracy własnej	Liczba godzin
1	Przygotowanie raportu, projektu, prezentacji, dyskusji	5
2	Konsultacje przedmiotowe i zaliczenie w bezpośrednim kontakcie z prowadzącym	11
3	Przygotowanie do zajęć w tym studiowanie literatury	5

Metody dydaktyczne:

analiza przypadku, ćwiczenia, dyskusja, praca w grupach, projekt, prezentacja

Sposoby weryfikacji i oceny efektów uczenia się:

uczestnictwo w zajęciach, dyskusja, odpowiedź ustna, projekt, rozwiązanie zadania problemowego

Kryteria oceny:

1. Student na ocenę 5.0 projektuje motywacyjne podstawy tworzenia zespołu/usprawnienia pracy zespołowej w organizacji
2. Student na ocenę 5.0 dobiera stosowne metody i techniki zarządzania konfliktem
3. Student na ocenę 5.0 wskazuje zalety i wady proponowanych metod i technik zarządzania zespołem wirtualnym, wielokulturowym

Literatura:

obowiązkowa:

1. K. Kmiotek, T. Piecuch, Zachowania organizacyjne. Teoria i przykłady, Warszawa, 2012, Difin
2. S. P. Robins, T. A. Judge, Zachowania w organizacji, Warszawa, 2011, PWE

zalecana/fakultatywna:

1. A. Osterwalder, S. Mastrogiacomo, Skuteczne zarządzanie zespołem. Jak uzyskać harmonię, zaufanie i widoczne efekty pracy w zespole, Gliwice, 2022, Helion