

Lista zagadnień na test kompetencji wymaganych od kandydatów na kierunek Informatyka studia II stopnia na Wydziale Informatyki i Matematyki w procesie rekrutacji na studia rozpoczynające się w semestrze letnim 2025/26, którzy otrzymali negatywną ocenę zgodności efektów uczenia się

Algorytmy

1. **Algorytmy sortowania i wyszukiwania** – klasyczne metody porządkowania danych i znajdowania elementów
2. **Algorytmy grafowe** (przeszukiwanie grafu i wyznaczanie najkrótszych ścieżek)
3. **Paradygmaty projektowania algorytmów** (dziel i zwyciężaj, programowanie dynamiczne, algorytmy zachłanne)
4. **Złożoność obliczeniowa algorytmów oraz problem NP-zupełności** – analiza wydajności i teoria problemów trudnych obliczeniowo
5. **Algorytmy równoległe i rozproszone** – przetwarzanie danych przy użyciu wielu wątków, procesorów lub komputerów jednocześnie

Struktury danych

1. **Struktury danych liniowe** (listy, stosy, kolejki)
2. **Drzewa i hierarchiczne struktury danych** (drzewa BST, kopce)
3. **Struktury haszujące** (tablice mieszające)
4. **Struktury danych dla dużych zbiorów danych** (indeksy i struktury wykorzystujące pamięć zewnętrzną)

Systemy operacyjne

1. **Zarządzanie procesami i szeregowanie zadań** – mechanizmy tworzenia procesów i algorytmy planowania procesora
2. **Wątki i mechanizmy synchronizacji** (muteksy, semaforey, monitory) – obsługa współbieżności w systemie
3. **Zarządzanie pamięcią i pamięć wirtualna** (alokacja, stronicowanie) – przydział pamięci dla procesów i techniki zwiększania przestrzeni adresowej
4. **Systemy plików i zarządzanie urządzeniami wejścia-wyjścia** – organizacja danych na dyskach oraz komunikacja z urządzeniami peryferyjnymi

Bazy danych

1. **Projektowanie relacyjnych baz danych i normalizacja** (postacie normalne) – tworzenie struktury bazy zapewniającej spójność i minimalizację redundancji
2. **Przetwarzanie i optymalizacja zapytań w SQL** – sposoby realizacji zapytań i techniki przyspieszania dostępu do danych
3. **Transakcje i mechanizmy współbieżności w bazach danych** (ACID, blokady) – zapewnienie integralności danych przy równoczesnym dostępie wielu użytkowników
4. **Systemy baz danych NoSQL i alternatywne modele danych** (dokumentowe, grafowe, klucz-wartość) – nierelacyjne podejścia do przechowywania i wyszukiwania informacji

Sieci komputerowe

1. **Modele sieciowe i protokoły warstwowe** (OSI, TCP/IP) – organizacja komunikacji w warstwach i role poszczególnych protokołów

2. **Routing i algorytmy wyznaczania tras w sieciach** – metody przesyłania pakietów i protokoły decydujące o ścieżkach w sieci
3. **Protokoły transportowe i kontrola przepływu danych (TCP/UDP)** – zapewnienie niezawodności transmisji oraz zarządzanie przepustowością i przeciążeniami
4. **Bezpieczeństwo sieciowe i kryptografia w transmisji danych** – szyfrowanie, uwierzytelnianie i zabezpieczanie komunikacji sieciowej
5. **Sieci bezprzewodowe i mobilne** – specyfika transmisji bezprzewodowej i protokoły dla sieci Wi-Fi oraz komórkowych

Grafika komputerowa

1. **Grafika komputerowa i techniki renderowania** (rasteryzacja vs ray tracing) – generowanie obrazów 2D/3D na podstawie sceny cyfrowej
2. **Modelowanie obiektów 3D i transformacje geometryczne** – reprezentacja i manipulacja obiektami trójwymiarowymi w przestrzeni
3. **Modele oświetlenia i cieniowania w grafice** – symulacja zachowania światła i nadawanie realizmu scenom 3D poprzez cieniowanie
4. **Przetwarzanie i analiza obrazów cyfrowych** (np. filtracja, segmentacja) – techniki poprawy jakości, wykrywania cech oraz interpretacji zawartości obrazów

Inżynieria oprogramowania

1. **Metodyki tworzenia oprogramowania** (kaskadowa, iteracyjna, zwinne) – sposoby organizacji procesu deweloperskiego od analizy po wdrożenie
2. **Analiza wymagań i projektowanie systemów** – zbieranie potrzeb użytkowników i tworzenie modeli systemu przed implementacją
3. **Architektura oprogramowania i wzorce projektowe** – wysokopoziomowa struktura systemu oraz sprawdzone schematy rozwiązywania typowych problemów projektowych
4. **Testowanie oprogramowania i zapewnianie jakości** – metody wykrywania błędów, automatyzacja testów oraz procesy zapewniające niezawodność produktu
5. **Procesy DevOps** (ciągła integracja i dostarczanie oprogramowania) – praktyki łączące rozwój i operacje w celu automatyzacji i usprawnienia cyklu wydawniczego

Sztuczna inteligencja

1. **Algorytmy uczenia maszynowego** (nadzorowane i nienadzorowane) – techniki pozwalające komputerom uczyć się na podstawie danych
2. **Sieci neuronowe i głębokie uczenie (Deep Learning)** – modele inspirowane biologicznymi neuronami oraz wielowarstwowe sieci uczące się z dużych zbiorów danych
3. **Reprezentacja wiedzy i systemy ekspertowe** – sposoby formalnego zapisu informacji i reguł oraz systemy wnioskujące na ich podstawie
4. **Algorytmy ewolucyjne i genetyczne** – metody inspirowane procesem ewolucji (np. algorytmy genetyczne) do rozwiązywania złożonych problemów
5. **Przetwarzanie języka naturalnego (NLP)** – techniki umożliwiające komputerom rozumienie i generowanie ludzkiego języka

Metody numeryczne

1. **Rozwiązywanie układów równań liniowych** (np. eliminacja Gaussa, rozkład LU) – metody obliczania rozwiązań dużych systemów równań

2. **Interpolacja i aproksymacja funkcji** – wyznaczanie funkcji przybliżających na podstawie dyskretnych punktów danych
3. **Metody numeryczne całkowania i różniczkowania** – przybliżone obliczanie całek oznaczonych oraz pochodnych funkcji
4. **Numeryczne rozwiązywanie równań różniczkowych** (np. metoda Eulera) – techniki wyznaczania przybliżonych rozwiązań równań opisujących zjawiska dynamiczne