

Zagadnienia na egzamin dyplomowy

Kierunek Informatyka, studia licencjackie, stacjonarne

Niniejszy dokument zawiera wykaz zagadnień, w oparciu o które będą formułowane pytania na egzaminach dyplomowych studentów kierunku Informatyka (studia licencjackie, stacjonarne). Zaleca się, aby egzamin składał się z trzech pytań. Jedno z nich może dotyczyć tematyki pracy seminaryjnej i wykraczać poza poniższy zestaw zagadnień.

Matematyka i jej zastosowania w informatyce

1. Rachunek zdań. Tautologie. Drzewa formuł. Zastosowanie rachunku zdań.
2. Podstawowe działania na zbiorach i prawa rachunku zbiorów.
3. Podstawowe typy relacji i ich grafy, relacje częściowego porządku i równoważności.
4. Podstawowe definicje dotyczące funkcji, przykłady i własności funkcji elementarnych.
5. Zasada indukcji matematycznej i jej zastosowania, w szczególności w rekurencji.
6. Rozwiązywanie układów równań liniowych, macierze i wyznaczniki.
7. Liczby zespolone i ich zastosowania.
8. Algorytm Euklidesa, liczby pierwsze i względnie pierwsze, relacja kongruencji i rozwiązywanie układów kongruencji.
9. Kryptografia z kluczem publicznym – algorytmy i ich implementacje. Podpis cyfrowy, elektroniczny, szyfrowanie.
10. Pochodna funkcji jednej zmiennej. Interpretacja geometryczna pochodnej. Zastosowanie rachunku różniczkowego do badania przebiegu zmienności funkcji jednej zmiennej.
11. Pojęcie funkcji pierwotnej i podstawowe metody całkowania.

Statystyczna analiza danych

1. Rozkłady zmiennych losowych. Przykłady rozkładów dyskretnych i absolutnie ciągłych.
2. Charakterystyki liczbowe w statystyce opisowej (miary tendencji centralnej i rozproszenia).
3. Wnioskowanie statystyczne: estymacja wartości oczekiwanej i wariancji, testowanie prostych hipotez o wartości oczekiwanej.

Obliczalność i złożoność obliczeniowa

1. Języki regularne. Automaty skończone (deterministyczne i niedeterministyczne) i wyrażenia regularne oraz ich zastosowania.
2. Języki bezkontekstowe. Gramatyki bezkontekstowe i automaty ze stosem.
3. Pojęcie obliczalności. Teza Churcha. Podstawowe modele obliczeń.
4. Rozstrzygalność i częściowa rozstrzygalność problemów obliczeniowych.
5. Czasowa i pamięciowa złożoność obliczeniowa w modelach deterministycznych i niedeterministycznych (twierdzenie Savitcha, itp.).
6. Problemy trudne i zupełne, przykłady problemów o różnej złożoności. Problem „Czy $P=NP$?”. Problem „Czy $P=NP$?”.

Algorytmy i struktury danych

1. Złożoność obliczeniowa (czasowa i pamięciowa) oraz poprawność algorytmów. Notacja asymptotyczna.
2. Podstawowe metody projektowania algorytmów (metoda zachłanna, „dziel i zwyciężaj”, programowanie dynamiczne, itp.). Przykłady algorytmów ilustrujących poszczególne metody.
3. Przykłady algorytmów grafowych i ich zastosowań.
4. Podstawowe algorytmy geometryczne.
5. Podstawowe algorytmy tekstowe.
6. Algorytmy sortowania i ich złożoność.
7. Dynamiczne struktury danych: stos, kolejka, lista, drzewo BST, kolejka priorytetowa, itp.

Programowanie obiektowe

1. Zasady programowania zorientowanego obiektowo. Pojęcie klasy i obiektu, konstruktory, tryby dostępu do składowych klasy.
2. Dziedziczenie. Klasy abstrakcyjne, interfejsy, metody wirtualne.
3. Wątki, zasada działania, synchronizacja i komunikacja.
4. Mechanizmy obsługi błędów. Wyjątki.

Programowanie funkcyjne

1. Porównanie paradygmatów programowania imperatywnego i deklaratywnego.
2. Funkcje czyste: definicja i przykłady.
3. Znaczenie i zastosowania rekurencji ogonowej w językach funkcyjnych.

Bazy danych

1. System zarządzania bazami danych. Funkcje systemu. Przykłady SZDB.
2. Model relacyjnych bazy danych. Relacje, związki, klucze, wartość nieoznaczona, języki zapytań.
3. Język SQL – podstawowe konstrukcje. Selekcja, złączenia, agregacje.
4. Optymalizacja zapytań – indeksy, wyszukiwanie pełnotekstowe, przepisywanie zapytań, tuning bazy.
5. Perspektywy i wyrażenia tablicowe. Proceduralne rozszerzenia języka SQL.
6. Bezpieczeństwo danych.

Inżynieria oprogramowania

1. Systemy kontroli wersji oraz sposoby ich działania.
2. Metodyki tworzenia oprogramowania: klasyczne i zwinne.
3. Testowanie oprogramowania: rodzaje testów, testy automatyczne.
4. Wzorce projektowe: definicje wraz z omówieniem idei, uczestników i konsekwencji stosowania na wybranych przykładach. Wzorzec Model View Controller.

Systemy operacyjne

1. System operacyjny, jego składowe, funkcje, zasoby i programy systemowe.
2. Procesy, rodzaje planistów i algorytmy planowania przydziału procesora.
3. Pamięć wirtualna i algorytmy zastępowania stron.
4. Organizacja plików, standardowe i sieciowe systemy plików.

Sieci komputerowe, programowanie sieciowe

1. Programowanie sieciowe. Model klient-serwer, peer-to-peer. Komunikacja między procesami (na jednej oraz na różnych maszynach).
2. Modele TCP/IP i ISO/OSI. Stos protokołów i protokoły różnych warstw.
3. Routing statyczny i dynamiczny w sieciach komputerowych.
4. Wybrane usługi sieciowe np. DHCP, DNS, HTTP, SMTP, POP3, IMAP, NFS, SAMBA, SNTTP, TFTP, FTP, transmisja multimediów (RTP, RTSP, RTCP, SIP) – zasada działania, rodzaje serwerów i oprogramowanie serwerowe.
5. Metody i narzędzia ochrony sieci lokalnej przed atakami pochodzącymi z sieci zewnętrznej i wewnętrznej (tunelowanie, firewalle różnych warstw sieciowych, szyfrowanie w różnych warstwach sieciowych).