

Zagadnienia na egzamin dyplomowy

INFORMATYKA I stopień

Przedmioty wspólne:

Wprowadzenie do informatyki

- Informatyka, informacja, zadanie algorytmiczne, algorytm i sposoby jego zapisu.
- Podstawy algorytmizacji, schematy zwarte NS (Nassi-Schneiderman).
- Język programowania, składnia, semantyka. Język proceduralny. Konstrukcje strukturalne i ich realizacja.
- Proste typy danych w językach programowania. Sposoby kodowania znaków. Liczby stałopozycyjne i zmiennopozycyjne.
- Strukturalne typy danych w językach programowania.
- Podejście zstępujące i wstępujące w programowaniu. Procedury i funkcje. Rekurencja w programowaniu. Problemy rozwiązywane z użyciem rekurencji. Modele danych.
- Podstawowe pojęcia ze złożoności obliczeniowej algorytmów. Złożoność obliczeniowa, funkcja złożoności obliczeniowej, rząd złożoności obliczeniowej.
- Podstawowe pojęcia lingwistyki matematycznej. Wzorce, automaty, wyrażenia regularne i gramatyki.
- Języki formalne, klasy P i NP, problemy NP-zupełne, teoria Turinga.

Programowanie strukturalne (zagadnienia dotyczą języka C)

- Struktura języka C. Zmienne, stałe, operatory, wyrażenia. Wypisywanie wyników, wprowadzenie danych do programowania. Tworzenie i uruchamianie programu.
- Wprowadzenie do algorytmiki. Schematy Nassi-Schneidermana. Programy imperatywne.
- Funkcje standardowe i funkcje własne. Prototypy funkcji.
- Programowanie z instrukcjami strukturyzującymi: IF, IF ... ELSE, SWITCH.
- Konwersja i rzutowanie typów.
- Programowanie iteracyjne z WHILE, DO ... WHILE, FOR.
- Stałe, zmienne lokalne i globalne. Zasięg i zakres zmiennych. Operatory arytmetyczne, relacyjne, logiczne i bitowe.
- Zmienne wskaźnikowe, operatory i arytmetyka wskaźnikowa. Tablice wskaźnikowe. Wskaźniki jako argumenty funkcji.
- Tablice statyczne jednowymiarowe i wielowymiarowe. Inicjalizacja tablic. Tablice o zmiennych rozmiarach. Tablice wskaźników.
- Dynamiczna alokacja pamięci. Dostęp do danych. Klasy pamięci. Operatory bitowe.
- Łańcuchy znakowe, tablice, działania na łańcuchach, wczytywanie i wypisywanie łańcuchów.

- Złożone typy danych: struktury, unie, pola bitowe, wyliczenie enum, definicje nazwy typu typedef.
- Podstawowe operacje plikowe, znaczniki trybu dostępu do plików. Binarne operacje plikowe. Dostęp swobodny do plików.
- Standardowe funkcje wejścia-wyjścia nieformatowane i formatowane. Specyfikacje kodów dla grupy printf(), scanf(), modyfikatory, flagi.
- Nowe operatory arytmetyczne. Priorytety operatorów. Zarządzanie pamięcią, data i czas.
- Preprocesor. Standard C99.

Programowanie obiektowe w C++

- Język C++ – paradygmaty programowania obiektowego.
- Przeciążanie operatorów, wskaźnik this, referencje.
- Funkcje i klasy wirtualne.
- System wejścia-wyjścia, mechanizmy formatowania.
- Dynamiczna alokacja pamięci, statyczne elementy klas.
- Szablony funkcji i klas, przestrzenie nazw.
- Wyjątki i ich obsługa.
- Klasa string – łańcuchy znakowe w stylu C++, standardowa biblioteka szablonów
- Wizualne biblioteki obiektowe, VCL, .NET, Qt.
- Metody tworzenia aplikacji wizualnych z wykorzystaniem biblioteki Qt.

Podstawy algorytmiki

- Definicja algorytmu, cechy poprawnego algorytmu, złożoności algorytmów i sposoby jego zapisu.
- Podstawowe techniki algorytmiczne: dziel i zwyciężaj, metoda zachłanna, programowanie dynamiczne, rekurencja.
- Algorytmy sortowania elementów w tablicy: BubbleSort, InsertSort, SelectSort, QuickSort, MergeSort.
- Algorytm flagi polskiej i flagi francuskiej. Selekcja - algorytm Hoare'a, „magicznych piątek”.
- Algorytmy wyszukiwania informacji w tablicach.
- Algorytmy wyszukiwania wzorca w tekście, zastosowanie funkcji haszujących.
- Dynamiczne struktury danych: stos, kolejka, lista.
- Definicja drzewa i podstawowe na nim operacje: BST, AVL, B-drzewa.
- Definicja kopca, sortowanie przez kopcowanie.
- Grafy, algorytmy grafowe i ich zastosowania.
- Otoczka wypukła – algorytm Grahama.
- Problemy klasy NP i NP-zupełne.

Wstęp do systemów operacyjnych

- Przegląd systemów operacyjnych. Zadania i zasady działania systemów operacyjnych. Struktura systemu.
- Uprawnienia, listy praw dostępu, rodzaje plików, struktura dysku i systemy plików. Strumienie, potoki i filtry, znaki specjalne i cytowanie, kontrola procesów, środowisko powłoki.
- Programowanie w powłoce. Sygnały i pułapki. Edytor strumieniowy sed i interpreter poleceń awk.
- Program make i pliki Makefile. Tworzenie bibliotek archiwalnych i współdzielonych w języku C.
- Podstawy systemu kontroli wersji Git.
- Komunikacja międzyprocesowa: potoki, pliki FIFO, kolejki komunikatów, pamięć współdzielona i semaforey. Implementacja w języku C.
- Procesy i wątki. Współbieżność. Planowanie przydziału procesora. Szeregowanie zadań. Przełączanie kontekstu, wyłączenie. Problemy zastoju i głodzenia.
- Wątki POSIX, muteksy, synchronizacja, semaforey i sekcje krytyczne. Implementacja w języku C.
- Zarządzanie pamięcią. Stronicowanie, pamięć wirtualna – pojęcia i algorytmy sterujące. Odwzorowywanie plików w pamięci.
- Podstawowe zadania systemu plików. Aliasy, systemy hierarchiczne. Alokacja indeksowa i listowa. Przykładowe systemy plików. Pliki i katalogi z poziomu aplikacji w języku C.
- System wejścia-wyjścia, obsługa programowa, przerwania, bezpośredni dostęp do pamięci.
- Zarządzanie pamięcią masową. Sterowniki urządzeń. Wejście-wyjście w bibliotece C.
- Ochrona i bezpieczeństwo. Problemy bezpieczeństwa w wybranych systemach operacyjnych.

Podstawy sieci komputerowych

- Wprowadzenie do sieci komputerowych: modele warstwowe, rodzaje sieci teleinformatycznych, urządzenia sieci teleinformatycznych.
- Adresacja urządzeń sieciowych warstwy 2 i 3.
- Rodzina standardów IEEE 802.1 i IEEE 802.3 (m. in.:STP, wybrane firmowe metody zwiększania niezawodności, agregacja łączy, metody zabezpieczania portów oraz urządzeń sieciowych, jedno i wielowarstwowe sieci wirtualne).
- Charakterystyka warstwy sieciowej. Podstawy metod doboru trasy w sieciach IP (urządzenia pracujące w warstwie III, rodzaje metod doboru trasy, trasowanie statyczne, trasowanie dynamiczne, RIP, OSPF).
- Listy sterowania dostępem, translacja adresów (listy sterowania dostępem: standardowe i rozszerzone, statyczna i dynamiczna translacja adresów).

- Dynamiczne przydzielanie adresów.
- Omówienie wybranych protokołów warstwy transportowej.
- Charakterystyka wybranych protokołów usług sieciowych (DNS, protokołów pocztowych, SSH, RDP, VNC, FTP).
- Podstawy bezpieczeństwa w sieciach komputerowych.
- Kierunki rozwoju lokalnych sieci komputerowych.

Algorytmy analizy numerycznej

- Arytmetyka zmiennopozycyjna, rodzaje błędów i ich oszacowania. Reguła Kahana. Uwarunkowanie zadania. Algorytmy numerycznie poprawne, algorytmy stabilne.
- Schemat Hornera. Interpolacja wielomianowa. Wzór iteracyjny Neville'a.
- Funkcje sklejjane. Wielomiany ortogonalne.
- Aproksymacja średniokwadratowa, aproksymacja jednostajna, metoda najmniejszych kwadratów. Układy Czebyszewa.
- Całkowanie numeryczne. Kwadratury interpolacyjne. Kwadratury Newtona-Cotesa. Kwadratury Gaussa.
- Metody rozwiązywania układów równań liniowych. Metoda ortogonalizacji Householdera.
- Metody rozwiązywania równań nieliniowych i ich układów. Wielowymiarowa metoda Newtona. Zbieżność metod iteracyjnych.
- Numeryczne rozwiązywanie równań różniczkowych zwyczajnych. Metoda Eulera. Metoda Runge'go-Kutty.
- Wartości własne i wektory własne macierze.

Programowanie obiektowe w Java

- Wprowadzenie do języka Java: cechy języka, środowiska programistyczne, rodzaje programów i struktura programu, porównanie z innymi językami, modularyzacja, zarządzanie zależnościami, kontrola wersji.
- Podstawy programowania: składnia języka, instrukcje, typy danych, operatory, tablice, funkcje matematyczne, praca z łańcuchami znaków.
- Programowanie obiektowe: klasy i tworzenie obiektów, konstruktory, przeciążanie i przesłanianie metod, specyfikatory dostępu, kompozycja i dziedziczenie, składowe statyczne, klasy abstrakcyjne i wewnętrzne, interfejsy, finalizacja.
- Podstawy programowania funkcyjnego, parametryzacja typów, kolekcje obiektów, przetwarzanie strumieniowe i wzorce projektowe.
- Graficzny interfejs użytkownika: tworzenie kontenerów i komponentów, zarządzanie układem komponentów, obsługa zdarzeń.
- Obsługa wyjątków i logowanie pracy aplikacji.
- Obsługa systemu wejścia/wyjścia, praca ze strumieniami, zbiorami danych i plikami.
- Przetwarzanie rozproszone: programowanie sieciowe, usługi sieciowe.

- Wielowątkowość języka Java: tworzenie, zarządzanie i synchronizacja wątków.
- Trwałe przechowywanie danych, bazy danych i ORM.
- Podstawy bezpieczeństwa w aplikacjach Java.

Wprowadzenie do systemów baz danych

- Pojęcie bazy danych, jej charakterystyka i znaczenie we współczesnej organizacji. Klasyfikacja współczesnych baz danych, ich charakterystyka i obszary wykorzystania.
- System zarządzania bazą danych i jego funkcje. Pojęcie modelu danych. Rodzaje modeli danych i ich charakterystyka.
- Relacyjny model danych – struktury danych, integralność danych, klucze, operacje algebry relacji.
- Modelowanie danych – model związków encji, reguły modelowania encji, atrybuty encji, związki między encjami.
- Transformacja modelu związków encji do postaci modelu relacyjnego – reguły transformacji encji, rodzaje związków między tabelami, reguły transformacji związków.
- Współczesne silniki relacyjnych baz danych (RBD). Wybrane narzędzia dostępne do współczesnych silników RBD i ich funkcjonalności.
- Tworzenie RBD w języku SQL – tworzenie tabel i związków między nimi, więzów integralności oraz indeksów.
- Wstawianie, modyfikowanie i usuwanie danych w RBD przy użyciu języka SQL.
- Tworzenie zapytań do RBD w języku SQL.
- Optymalizacja zapytań w języku SQL.

Podstawy grafiki komputerowej

- Podstawowe pojęcia grafiki 2D.
- Przegląd narzędzi i środowisk ułatwiających przetwarzanie grafiki rastrowej.
- Podstawowe operacje rastrowe, przekształcenia bezkontekstowe i kontekstowe, interpolacja.
- Kompresja obrazu rastrowego.
- Przegląd narzędzi i środowisk ułatwiających przetwarzanie grafiki wektorowej 2D, przekształcenia geometryczne obiektów 2D, rysowanie krzywych.
- Podstawowe pojęcia grafiki 3D, rodzaje i budowa obiektów 3D, metody konstrukcji, hierarchizacja obiektów 3D.
- Przekształcenia geometryczne obiektów 3D.
- Przegląd narzędzi i środowisk ułatwiających przetwarzanie grafiki 3D.
- Operacje na siatkach 3D.
- Wizualizacja sceny 3D, rzutowanie obrazu 3D, Podstawy obliczania oświetlenia scen 3D, modele oświetlenia i cieniowania.
- Podstawy animacji rastrowej i wektorowej, rodzaje animacji.

Podstawy inżynierii oprogramowania

- Dziedzina Inżynierii Oprogramowania. Etyka IO. Typowe etapy wytwarzania oprogramowania i ich zawartość. Podstawowe modele wytwarzania oprogramowania.
- Analiza systemów informacyjno-decyzyjnych. Modelowanie procesów biznesowych i specyfikacja zawartości dokumentów. Notacje BPMN i XML.
- Inżynieria wymagań. Pozyskiwanie, określenie i dokumentowanie wymagań.
- Projektowanie: metodyki strukturalne i obiektowe. Notacje DFD, ERD, UML. Narzędzia CASE – typologia, konstrukcja, metody pracy.
- Szczegółowe techniki modelowania procesów i struktur danych oprogramowania. Modele konceptualne, logiczne i implementacyjne. Mapowanie modeli. Podejście zorientowane obiektowo i UX.
- Standardy i projekt interfejsu z użytkownikiem. Zasady konstrukcji poprawnego interfejsu. Narzędzia wspomagające prototypowanie interfejsu. Internacjonalizacja interfejsu - problemy i metody.
- Kodowanie danych. Typy i zasady. Konstrukcja kodów. Cyfry kontrolne. Projektowanie systemu kodów – zasady i przykłady.
- Testowanie i walidacja oprogramowania. Cele, zakres i typy metod testowania oprogramowania. Problem organizacji procesu testowania.
- Rozwój oprogramowania na etapie eksploatacji. Proces wnoszenia zmian do oprogramowania. Zarządzanie konfiguracjami. Wymiana oprogramowania.
- Modele i systemy zapewnienia jakości oprogramowania.

Bezpieczeństwo informacji

- Pojęcie bezpieczeństwa informacji. Typowe zagrożenia bezpieczeństwa. Podstawowe sposoby obrony przed atakami. Standardy zarządzania bezpieczeństwem informacji.
- Wirtualizacja – rodzaje i zastosowania wirtualizacji, wady i zalety.
- Mechanizmy bezpieczeństwa systemów operacyjnych. Utwardzanie ochrony systemów operacyjnych.
- Wprowadzenie do kryptografii. Podstawowe pojęcia oraz mechanizmy stosowane w kryptografii.
- Kryptografia symetryczna i asymetryczna.
- Skrót kryptograficzny. Przegląd funkcji skrótu.
- Infrastruktura klucza publicznego. Podpis cyfrowy, SSL/TLS.
- Bezpieczeństwo danych. Przegląd najczęstszych zagrożeń bezpieczeństwa danych oraz podstawowe metody ochrony przed zagrożeniami.
- Szkodliwe oprogramowanie i obrona przed nim.
- Zagrożenia w sieciach komputerowych, podstawowe metody obrony przed nimi.
- Zapory sieciowe, wirtualne sieci prywatne.
- Polityka bezpieczeństwa informacji.

Podstawy aplikacji internetowych

- Podstawowa architektura aplikacji internetowej. Serwer i klient HTTP. Podstawy działania protokołu HTTP. Interakcja w środowisku WWW. Ewolucja standardu języka HTML. Struktura dokumentu HTML5. Podstawowe znaczniki HTML5 i ich atrybuty. Znaczenie walidacji kodu HTML. Walidatory W3C.
- Wizualna prezentacja stron internetowych za pomocą kaskadowych arkuszy stylów (CSS). Definicja i rodzaje arkuszy stylów. Sposoby osadzania arkuszy w dokumencie HTML i pojęcie kaskadowości. Box Model. Selektory i deskryptory. Klasy i wyjątki. Walidacja arkuszy CSS.
- Formularze HTML jako podstawa interakcji z użytkownikiem aplikacji internetowej. Bezpieczeństwo aplikacji internetowej i znaczenie walidacji danych przesyłanych z formularza HTML. Metody walidacji danych w formularzu – typy pól i atrybuty HTML5. Zastosowanie wyrażeń regularnych w walidacji danych.
- Projektowanie interfejsu użytkownika aplikacji internetowej w oparciu o standardy W3C. Responsywność (RWD), elastyczność i dostępność stron. Progresywne strony internetowe. Optymalizacja serwisów pod kątem wyszukiwarek internetowych i standardy SEO. Narzędzia deweloperskie w przeglądarkach internetowych.
- Programowanie po stronie klienta. Wprowadzenie do JavaScript. Interakcja z użytkownikiem witryny internetowej za pomocą JavaScript. Umieszczanie skryptów w dokumencie HTML. Składnia i obiekty języka JavaScript.
- Obiektowy model dokumentu (DOM). Obsługa zdarzeń. Dostęp do pól formularzy po stronie klienta za pomocą JavaScript. Obsługa wyrażeń regularnych w JavaScript.
- Biblioteka jQuery i jej zastosowania.
- Wprowadzenie do szkieletu Bootstrap: struktura strony, klasy CSS i skrypty JS pomocne do tworzenia stron responsywnych.

Podstawy paradygmatów programowania

- Rodzaje paradygmatów, generacje języków programowania.
- Semantyka oraz syntaktyka języków programowania, gramatyka bezkontekstowa, typy danych, przekazywanie parametrów, przeciążenia operatorów i podprogramów.
- Programowanie imperatywne: zmienne, atrybuty, wiązania statyczne i dynamiczne, organizacja wywołań podprogramów, przydział pamięci. Przykłady z różnych języków programowania.
- Programowanie funkcyjne. Wprowadzenie, rachunek lambda, rekursja, funkcje wyższego rzędu, kolekcje i operacje, leniwa ewaluacja, wnioskowanie(inferencja) typów. Przykłady z języka Haskell, Python.
- Programowanie w logice. Rachunek predykatów, fakty, reguły, nawroty, rezolucja, unifikacja, listy, wnioskowanie, rekursja. Przykłady w języku Prolog.
- Przegląd dodatkowych paradygmatów: modularne, aspektowe, komponentowe, agentowe, strukturalne, zdarzeniowe, kontraktowe, generyczne, refleksyjne,

programowanie sterowane przepływem danych oraz współbieżne, równoległe i rozproszone.

Podstawy sztucznej inteligencji w języku Python

- Podstawy uczenia maszynowego.
- Przegląd środowisk i bibliotek Python. Podstawowe informacje nt. języka.
- Klasyfikacja najbliższego sąsiada.
- Regresja liniowa.
- Maszyny wektorów nośnych.
- Analiza głównych składowych i liniowa analiza dyskryminacyjna.
- Drzewa decyzyjne.
- Wprowadzenie do sieci neuronowych.
- Uczenie i działanie sieci neuronowej.
- Sieci jednowarstwowe, wielowarstwowe i propagacja wsteczna.
- Sieci samouczące, samoorganizujące się i rekurencyjne.
- Głębokie sieci neuronowe i uczenie głębokich sieci neuronowych.
- Głębokie splotowe sieci neuronowe.
- Głębokie rekurencyjne sieci neuronowe.
- Autoenkodery.