

# Egzamin wstępny na studia drugiego stopnia

## **Informatyka stosowana**

6 czerwca 2021

### **1 Matematyka dyskretna**

- Zbiory, podzbiory i multizbiory.
- Funkcje i rozmieszczenia. Permutacje.
- Zależności rekurencyjne.
- Funkcje tworzące.
- Pojęcia wstępne teorii grafów: izomorfizm, spójność, drzewa i lasy.
- Grafy planarne.
- Droga Eulera. Cykle i ścieżki Hamiltona.
- Kolorowanie wierzchołków i krawędzi w grafach.

### **2 Podstawy systemów operacyjnych/UNIX**

- Podstawy systemu operacyjnego, jego cechy, dystrybucje i instalacja.
- Systemy plików, urządzenia, zasoby i ich podział.
- Jądro systemu, jego budowa, konfiguracja oraz kompilacja.
- Powłoka systemowa, logowanie lokalne i zdalne.
- Prawa dostępu, zmienne środowiskowe, procesy i zarządzanie nimi, potoki, filtry, standardowe we/wy oraz podstawowe polecenia powłoki.
- Zarządzanie użytkownikami, grupami, pakietami, usługami oraz zasobami.
- Wyrażenia regularne.

### 3 Algorytmy i struktury danych

- Złożoność pamięciowa algorytmów sortujących.
- Stabilność algorytmów sortujących.
- Złożoność czasowa algorytmów sortujących.
- Złożoność obliczeniową algorytmu Strassena.
- Wybór zachłanny w problemie wyboru zajęć
- Złożoność obliczeniowa algorytmów rekurencyjnych.
- Własność kopca typu max.
- Na czym polega rozwiązywanie kolizji za pomocą metody łańcuchowej w tablicach z haszowaniem.
- Algorytmy wypisujące węzły drzewa poszukiwań binarnych.
- Struktury danych konieczne do implementacji przeszukiwania grafu w głąb (DFS) i wszerek (BFS).
- Na czym polega metoda dziel i zwyciężaj?
- Zastosowanie strategii zachłannej do problemu wydawania reszty.
- Zastosowanie programowania dynamicznego do problemu mnożenia ciągu macierzy.
- Wykorzystanie twierdzenia o rekurencji uniwersalnej do rozwiązywania równań rekurencyjnych.
- Procedury dzielące algorytmu sortowania szybkiego.
- Własność drzewa wyszukiwań binarnych (BST).
- Niezmiennik pętli algorytmu, który buduje kopiec typu max.
- Pojęcie optymalnej podstruktury dla dyskretnego problemu plecakowego.
- Zastosowanie programowania dynamicznego do problemu wydawania reszty.
- Zastosowanie strategii zachłannej do konstrukcji optymalnego kodu Huffmana.

### 4 Programowanie proceduralne

- Jak obliczyć silnię przy użyciu pętli for?
- Jak używać inicjatorów desygnowanych?
- W jaki sposób zainicjować tablicę przy pomocy pętli for i funkcji scanf?
- Zasady arytmetyki wskaźników.
- Jak przy pomocy wskaźnika przeiterować po kolumnie tablicy dwuwymiarowej?
- Jakich poleceń należy użyć, aby napisać program, który wypisuje swoją nazwę?
- Jak deklarujemy modyfikowalne i niemodyfikowalne zmienne tekstowe?

- Kiedy mamy do czynienia z wyciekami pamięci?
- W jaki sposób można napisać komparator, który przekazany do funkcji `qsort` umożliwi posortowanie tablicy liczb całkowitych?
- W jaki sposób przy użyciu wskaźnika można wczytać zawartość pliku binarnego do tablicy?
- Porównaj działanie funkcji `strlen` i operatora `sizeof`?
- W jaki sposób funkcja `strcpy` kopiuje łańcuchy?
- Jak skopiować zawartość jednej zmiennej strukturalnej do drugiej?
- Jak zdefiniować funkcję o zmiennej liczbie argumentów?
- Jak zdefiniować funkcję, która doda element na koniec listy jednokierunkowej?
- Jak przy użyciu funkcji `fputc` oraz `fgetc` napisać program, który kopiuje pliki?
- Jak przy użyciu funkcji `bsearch` przeszukujemy tablice wskaźników do łańcuchów?
- Jak poprawnie napisać funkcję, która zamienia wartości przekazanych argumentów?
- Jak zadeklarować funkcję, której argumentem jest tablica dwuwymiarowa?
- Jak zdefiniować funkcję, która zwolni pamięć, którą zajmowała lista jednokierunkowej?

## 5 Podstawy programowania obiektowego

- Podstawy programowania obiektowego — wstęp teoretyczny. Obiekty i klasy. Wstępne omówienie abstrakcji, enkapsulacji, dziedziczenia, polimorfizmu. Języki programowania obiektowego.
- Język C++ : podstawowe cechy i zastosowania, kompilatory. Przykłady prostych programów. Składnia i elementy języka: typy danych, zmienne, instrukcje sterujące.
- Operatory. Referencje. Funkcje. Dostęp do biblioteki C.
- Przeładowanie nazw funkcji. Obiekty lokalne i globalne. Dynamiczna alokacja pamięci.
- Struktury. Klasy. Składniki publiczne, chronione i prywatne. Statyczne pola klas.
- Funkcje składowe (metody) klas. Wskaźnik „this”.
- Konstruktory klas. Destrukutory.
- Funkcje i klasy zaprzyjaźnione. Klasy zagnieżdżone. Konwersje. Tablice obiektów.
- Przestrzenie nazw. Dziedziczenie.
- Funkcje wirtualne. Polimorfizm. Klasy abstrakcyjne.
- Przeładowanie operatorów. Obsługa napisów w C++.

## 6 Programowanie obiektowe 1

- Pogłębienie zagadnień dotyczących dziedziczenia oraz funkcji wirtualnych.
- Szablony funkcji.
- Szablony klas.
- Biblioteka standardowa C++: kontenery, iteratory, algorytmy, alokatory.
- Rozszerzania języka C++ o: Prawe referencje, stałe wyrażenia, wyrażenia lambda, operatory przenoszenia.

## 7 Programowanie obiektowe 2

W kontekście programowania w języku Java (w aktualnej, najnowszej wersji), zagadnienia dotyczące:

- typów danych, rzutowania pomiędzy nimi oraz wyników zastosowania operatorów dostępnych dla poszczególnych typów
- definiowania własnych typów
- mechanizmów inicjalizacji zmiennych i instancji klas
- użycia modyfikatorów dostępu `public`, `protected`, `private` oraz dostępu *pakietowego*
- znaczenia słów kluczowych `abstract`, `static`, `final`, `var`, `this`, `super`
- zasad dziedziczenia pomiędzy klasami oraz interfejsami, oraz zachodzącego wtedy przesłaniania metod
- różnic między przesłanianiem i przeładowaniem (przeciążaniem) metod
- zasad definiowania i implementacji interfejsów, oraz dziedziczenia pomiędzy nimi, a także sposobu użycia w nich metod *domyślnych*
- wyrażień *lambda*
- sposobu obsługi wyjątków, typów reprezentujących wyjątki i relacji między nimi oraz różnic między wyjątkami *sprawdzanymi* i *niesprawdzanymi*
- zasad definiowania oraz użycia typów generycznych
- stosowania tablic, kontenerów (`ArrayList`, `LinkedList`, `TreeSet` itp.), oraz iteratorów i pętli *for-each*
- definiowania oraz stosowania typów wyczerpujących (`enum`)
- własności klas do obsługi napisów (`String`, `StringBuffer`, `StringBuilder`) oraz różnic między nimi
- podstawowych własności klasy `Class` i mechanizmu *refleksji*
- zastosowania podstawowych klas realizujących operacje wejścia/wyjścia, a także znaczenia interfejsów `Closeable` i `AutoCloseable`

## 8 Metody numeryczne

- Metody rozwiązywania układów równań liniowych: rozkład LU metodą Gaussa, wykorzystanie rozkładów LU, QR i SVD do rozwiązania układów równań (zwykłych i nadokreślonych), ortogonalizacja Grama–Schmidta.
- Metody iteracyjne rozwiązywania układów równań liniowych: metoda iteracji do ustalonego punktu (warunki zbieżności metody), metoda Jacobiego i Gaussa–Seidla, postać macierzy iterującej dla obu metod.
- Diagonalizacja macierzy: metoda potęgowa, redukcja Hotellinga, metoda iteracji odwrotnej, przesuwanie widma macierzy.
- Interpolacja wielomianowa Lagrange’a: założenia, wzór interpolacyjny, stopień wielomianu.
- Aproksymacja średniokwadratowa: założenia, norma, sposób wyznaczania współczynników.
- Rozwiązywanie równań nieliniowych: metody bisekcji, Regula Falsi, siecznych oraz stycznych (Newtona).
- Całkowanie numeryczne kwadraturami Newtona–Cotesa: wyprowadzenie wzorów zwykłych i złożonych dla metody trapezów i parabol.
- Całkowanie Monte Carlo: metoda podstawowa, błąd oszacowania całki, metoda orzeł-reszka.

## 9 Podstawy grafiki komputerowej

- Urządzenia wyjściowe a grafika wektorowa.
- Grafika rastrowa a wektorowa.
- Podstawowe elementy obrazu w grafice komputerowej.
- Powiększenie liniowe obrazu.
- Powiększenie optyczne obrazu.
- Odległość dobrego widzenia, rozdzielczość.
- Modele barw CMY, CMYK oraz RGB.
- Rozdzielczość koloru.
- Redukcja rozdzielczości kolorów.
- Odcień, nasycenie, jasność i jaskrawość.
- Prawa Grassmana.
- Własności wykresu chromatyczności  $xyY$ , przestrzenie barw.
- Biblioteka SFML, transformacje obiektów.
- Zdarzenia w bibliotece wxWidgets.
- Biblioteka wxWidgets, właściwości okien dialogowych.
- Biblioteka wxWidgets, kontrola procesu rysowania.
- Krzywe Beziera.
- Transformacje obiektów 2D.

## 10 Sieci komputerowe

- Model referencyjny budowy sieci komputerowych ISO OSI (Open System Interconnection).
- Mechanizmy rozstrzygania rywalizacji węzłów sieciowych o możliwość dostępu do medium transmisyjnego, charakterystyczne dla sieci LAN IEEE 802.3 (Ethernet) i bezprzewodowych Wireless LAN standardu IEEE 802.11 (WiFi).
- Budowa ramek warstwy łącza danych standardu IEEE 802.3 (Ethernet).
- Topologie fizyczne i logiczne sieci lokalnych LAN IEEE 802.
- Działanie przełącznika w sieci LAN typu IEEE 802.1D.
- Zastosowanie i budowa nagłówka sieci wirtualnych VLAN IEEE 802.1Q.
- Zasady adresacji w sieciach publicznych i prywatnych protokołu IPv4/CIDR.
- Zasady przesyłania pakietów przez routery IPv4 a przeznaczenie pól nagłówka datagramów IPv4.
- Mechanizmy multipleksowania transmisji na potrzeby usług transportu świadczone przez protokoły UDP i TCP.
- Zasada efektywnego i niezawodnego przesyłania segmentów przez protokół TCP.
- Mechanizm nawiązywania i rozwiązywania połączenia klient — serwer w protokole TCP.
- Przeznaczenie i zasady działania protokołów dynamicznego trasowania wewnątrz-domenowego (IGP) i między-domenowego (EGP) IPv4.
- Zasady działania systemu DNS, budowa i przeznaczenie podstawowych rekordów zasobów systemu (DNS resource records).
- Budowa i przeznaczenie etykiet MPLS.
- Rozwiązania służące do budowania wirtualnych sieci prywatnych VPN w środowisku IPv4.

## 11 Bazy danych 1

- Relacyjny model danych — model ER.
- Relacyjny model danych — definicja modelu.
- Relacyjny model danych — normalizacja.
- Język SQL — tworzenie struktur danych.
- Język SQL — przetwarzanie danych — polecenia DML.
- Procedury składowane i wyzwalacze w relacyjnych bazach danych.
- Dostęp do relacyjnych baz danych z języków programistycznych.

## 12 Grafy i ich zastosowania

- Grafy proste, multigrafy, digrafy, podgrafy, izomorfizm.
- Spójność,  $k$ -spójność, silna spójność.
- Stopnie wierzchołków i sekwencje graficzne.
- Ścieżki, cykle, grafy eulerowskie, grafy hamiltonowskie.
- Grafy dwudzielne ( $k$ -dzielne), grafy regularne.
- Reprezentacja grafów za pomocą macierzy sąsiedztwa, macierzy incydencji i list sąsiedztwa.
- Grafy drzewiaste, minimalne drzewa rozpinające, zliczanie drzew, kod Prüfera.
- Wyszukiwanie najkrótszych ścieżek.
- Silnie spójne składowe i sortowanie topologiczne.
- Skojarzenia, problem przydziału zadań — metoda węgierska.
- Problem chińskiego listonosza i problem komiwojażera.
- Grafy planarne, wzór Eulera (genus grafu), grafy dualne .
- Kolorowanie grafów i map, wielomiany chromatyczne.
- Przepływy na sieciach.
- Łańcuchy Markowa (algorytm *page rank*).

## 13 Metody inteligencji obliczeniowej

- Rodzaje uczenia: nadzorowane, nienadzorowane, ze wzmocnieniem.
- Model neuronu dyskretnego.
- Przebieg metody perceptronowej uczenia sieci neuronowej.
- Architektury wielowarstwowych sieci neuronowych.
- Sieć Hopfielda — budowa, cechy, własności.
- Sieć RBF — budowa, cechy, własności.
- Uczenie sieci neuronowych metodami WTA, delty i propagacji wstecznej.
- Funkcja przynależności zbioru rozmytego — definicja i własności.
- Operacje na zbiorach rozmytych — dopełnienie, przecięcie i suma.
- Bloki układu rozmytego i ich funkcje.
- Wnioskowanie Mamdaniego i Takagi–Sugeno.
- Algorytm genetyczny — jego schemat działania i podstawowe pojęcia: chromosom, gen, genotyp, fenotyp, funkcja przystosowania.
- Operatory selekcji i krzyżowania w algorytmach genetycznych.
- Schematy działania algorytmów symulowanego wyżarzania i PSO.

## 14 Inżynieria oprogramowania

- Podstawowe pojęcia inżynierii oprogramowania (cykl życia oprogramowania, proces wytwórczy, modele i metodyki).
- Rodzaje modeli liniowych i iteracyjnych, ich zalety i wady (kaskadowy, prototypowy, spiralny, przyrostowy, modele ewolucyjne itp).
- Rodzaje metodyk i ogólna ich charakterystyka (metodyki zwinne XP, Scrum, TDD, Kanban).
- Planowanie projektu: kosztów, harmonogramu, ryzyka (rodzaje ryzyka), zespołu, kontroli jakości.
- Inżynieria wymagań, wymagania funkcjonalne i нефункционалне (pozafunctionalne), podstawowe pojęcia, fazy.
- Modelowanie i projektowanie systemu, podstawowe diagramy UML.
- Testowanie systemu, rodzaje testów: modułowe, funkcjonalne, pozafunctionalne, akceptacyjne, testy regresji, testy białej i czarnej skrzynki, testy statyczne i dynamiczne, itp.
- Ewolucja oprogramowania, zarządzanie jakością, zmianami, testy związane ze zmianami.

## 15 Inżynierskie metody numeryczne

- Własności schematów różnicowych dla równań różniczkowych: A-stabilność, L-stabilność, zbieżność, spójność.
- schematy różnicowe dla równań różniczkowych zwyczajnych: jawna i niejawna metoda Eulera, metoda trapezów, współczynnik wzmocnienia dla problemu autonomicznego, stabilność schematów różnicowych.
- Metody Rungego–Kutty: tablica Butchera (relacje między współczynnikami), A-stabilność, L-stabilność, ogólna postać współczynnika wzmocnienia jawnych i niejawnych metod RK, określanie rzędu metody na podstawie tablicy Butchera.
- Ilorazy różnicowe: konstrukcja, błąd dyskretyzacji pochodnej.
- Metody rozwiązywania równania Poissona: metoda różnic skończonych, relaksacja lokalna i globalna.
- Metody rozwiązywania równania adwekcji: twierdzenie Couranta–Friedrichsa–Leviego dotyczące zbieżności, kryterium CFL, *zasada maksimum* dotycząca stabilności bezwzględnej schematu różnicowego, schematy upwind/downwind/z centralną pochodną przestrzenną/Crank–Nicolson dla równania adwekcji (wyprowadzenie + błąd dyskretyzacji schematu różnicowego + stabilność/zbieżność z *zasady maksimum*/kryterium CFL).