

# Programa pentru examenul de diferență clasa a XI-a, informatică

Conform programei școlare aprobată prin ordin al ministrului nr. 5099/09.09.2009

## CONȚINUTURI

### **1. Identificarea datelor care intervin într-o problemă și aplicarea algoritmilor fundamentali de prelucrare a acestora**

Structuri de date alocate dinamic\*

Grafuri neorientate și grafuri orientate

- Terminologie (graf neorientat, graf orientat, lanț, lanț elementar, drum, drum elementar, ciclu, ciclu elementar, circuit, circuit elementar, grad, graf parțial, subgraf, conexitate, tare conexitate, arbore, graf ponderat, arbore parțial, arbore parțial de cost minim)
- Tipuri speciale de grafuri (graf complet, graf hamiltonian, graf eulerian, graf bipartit, graf turneu)
- Reprezentarea grafurilor (matrice de adiacență, liste de adiacență, lista muchiilor, matricea costurilor)
- Algoritmi de prelucrare a grafurilor
  - Parcurgerea grafurilor în lățime și în adâncime
  - Determinarea componentelor conexe ale unui graf neorientat
  - Determinarea componentelor tare conexe ale unui graf orientat
  - Determinarea matricei lanțurilor/drumurilor
  - Determinarea drumurilor de cost minim într-un graf (algoritmul lui Dijkstra, algoritmul Roy-Floyd)
  - Arbori parțiali de cost minim (algoritmul lui Kruskal sau algoritmul lui Prim)

Structuri de date arborescente

- Arbori cu rădăcină (definiție, proprietăți, reprezentare cu referințe ascendente, reprezentare cu referințe descendente)
- Arbori binari (definiție, proprietăți specifice; reprezentarea arborilor binari cu referințe descendente; operații specifice)
- Tipuri speciale de arbori binari
  - Arbore binar complet – definiție, proprietăți, reprezentare secvențială
  - Arbore binar de căutare – definiție, proprietăți, operații specifice (inserare nod, ștergere nod, căutare element)
  - Heap-uri – definiție, proprietăți, operații specifice (inserare nod, extragerea nodului cu cheie maximă/minimă)

### **2. Elaborarea algoritmilor de rezolvare a problemelor**

Metode de programare

- Metoda de programare *Greedy* (descrierea generală a metodei, utilitate, aplicații)
- Metoda de programare *Backtracking* (descrierea generală a metodei, utilitate, aplicații)
- Metoda de programare *Divide et Impera*\*
- Metoda programării dinamice (descrierea generală a metodei, utilitate, aplicații)

Analiza eficienței unui algoritm

### **3. Implementarea algoritmilor într-un limbaj de programare**

Rezolvarea unor probleme cu caracter practic pentru grafuri, arbori, metode de programare

Elemente de programare orientată pe obiecte

- Principiile programării orientate pe obiecte
- Clase și obiecte (definire, utilizare, operații specifice)

\* Se studiază în vederea corelării programei curente cu programa anterioară