

Clasa a IX-a:

1. Algoritmi elementari:

- Tipuri simple de date (întreg, real, caracter, logic)
- Structura liniară, alternativă și repetitivă
- Prelucrarea cifrelor numerelor naturale scrise în baza 10
- Divizibilitate (divizorii numerelor naturale, numere prime, determinarea cmmdc prin algoritmul lui Euclid, cmmmc, numere prime între ele, simplificarea fracțiilor, descompunerea numerelor în factori primi)
- Calculul unor expresii (de exemplu, factorial, ridicare la putere)
- Algoritmul de exponențiere rapidă

2. Generări de șiruri:

- Generarea șirurilor pe baza unor reguli
- Șirul lui Fibonacci și alte șiruri recurente

3. Sisteme de numerație și reguli de conversie

4. Tablouri unidimensionale (vectori):

- Prelucrări elementare (de exemplu, parcurgere, inversare, verificarea unor proprietăți)
- Algoritmi de sortare în complexitate pătratică (sortarea prin selecție, sortarea prin inserție, metoda bulelor)
- Sortare prin numărare (folosind vectori de frecvență)
- Interclasarea tablourilor unidimensionale
- Vectori caracteristici/ de frecvență
- Operații cu mulțimi
- Ciurul lui Eratostene - fără precalculări avansate/aplicarea directă
- Căutare binară
- Determinarea elementului majoritar
- Sume parțiale în tablouri unidimensionale
- Probleme cu secvențe de valori (de exemplu, determinarea unei secvențe maxime cu o anumită proprietate, numărarea secvențelor, determinarea secvenței de sumă maximă,

prelucrarea secvențelor de lungime fixată, tehnica Two Pointers, tablouri de diferențe - Difference Arrays)

## 5. Metoda Greedy

Clasa a X-a:

### 1. Tabele de diferențe (Difference Arrays) 2D

### 2. Șiruri de caractere. Funcții specifice

### 3. Funcții:

- Declarare, definire, apel
- Variabile locale, variabile globale
- Transmiterea parametrilor prin valoare și prin referință

### 4. Utilizarea funcțiilor din biblioteca STL pentru sortare și căutare

### 5. Structuri de date:

- Stiva (stack). Aplicații specifice
- Coada (queue). Aplicații specifice. Algoritmul lui Lee
- Deque. Aplicații specifice
- Lista (liste simplu și dublu înlănțuite alocate dinamic)
- Structuri de date din biblioteca STL (Standard Template Library): pair, vector, list, deque, queue, priority\_queue, stack, set (inclusiv variantele unordered\_set și multiset), map (inclusiv variantele unordered\_map și multimap), bitset.

### 6. Operații cu numere mari:

- Adunarea numerelor mari
- Scăderea numerelor mari
- Înmulțirea unui număr mare cu un număr natural

### 7. Generări de elemente combinatoriale:

- Submulțimi
- Produs cartezian
- Permutări

- Combinări
- Aranjamente

#### 8. Elemente de combinatorică:

- Numărarea elementelor combinatoriale (submulțimi, produs cartezian, permutări, aranjamente, combinări)
- Aritmetică modulară (adunare, scădere, înmulțire, invers modular - pentru modulo număr prim)

#### 9. Recursivitate

#### 10. Metoda Divide et Impera

#### Clasele XI-XII:

##### 1. Elemente de geometrie:

- sistemul de coordonate cartezian
- distanța dintre două puncte
- ecuația dreptei
- distanța dintre un punct și o dreaptă
- panta unei drepte
- intersecții de drepte și segmente
- arii
- algoritmi de baleiere
- înfășurătoare convexă

##### 2. Metoda Backtracking

##### 3. Metoda programării dinamice:

- Probleme de numărare
- Probleme de determinare a unei soluții optime
- Memoizarea relațiilor de recurență
- Programare dinamică pe arbori și grafuri

##### 4. Grafuri orientate și neorientate:

- 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 bipartit, graf turneu)
- Reprezentarea grafurilor (matrice de adiacență, liste de adiacență, lista muchiilor/arcilor)
- Grafuri ponderate. Reprezentarea grafurilor ponderate (matricea costurilor, liste de adiacență cu costuri, lista muchiilor/arcilor cu costuri)
- Algoritmi de prelucrare a grafurilor:
  - o Parcurgerea grafurilor în lățime (BFS), în adâncime (DFS)
  - o Determinarea componentelor conexe ale unui graf neorientat
  - o Determinarea componentelor tare conexe ale unui graf orientat. Algoritmul Kosaraju-Sharir. Graful componentelor tare-conexe:
  - o Determinarea matricei lanțurilor/drumurilor (algoritmul Roy-Warshall)
  - o Descompunerea unui graf orientat fără circuite pe niveluri. Sortare topologică
  - o Determinarea drumurilor de cost minim într-un graf. Algoritmul lui Dijkstra, algoritmul Bellman-Ford, algoritmul Roy-Floyd
- Arbori:
  - o Definiție, proprietăți
  - o Arbori parțiali
  - o Arbori parțiali de cost minim (algoritmul lui Kruskal și algoritmul lui Prim)

#### 5. Structuri de date arborescente:

- Arbori cu rădăcină (definiție, proprietăți, reprezentarea arborilor cu rădăcină)
- Arbori binari (definiție, proprietăți specifice; reprezentarea arborilor binari)
- Operații pe structuri de date (interogări, actualizări)
- Arbore binar complet – definiție, proprietăți, reprezentare secvențială
- Heap-uri – definiție, proprietăți, operații specifice (inserare nod, extragerea nodului cu cheie maximă/minimă)
- Arbore binar de căutare – definiție, proprietăți, operații specifice (inserare nod, ștergere nod, căutare element)

- Reprezentarea mulțimilor disjuncte. Algoritmii Union-Find

Notă:

- Exceptând clasa a IX-a, programa fiecărei clase include și programele pentru toate clasele precedente.