

## **Programa pentru olimpiada națională de informatică (etapa județeană, etapa națională, baraje de selecție a lotului național lărgit)**

### **Gimnaziu**

#### **Clasa a V-a**

##### **1. Algoritmi elementari**

- Tipuri simple de date. Tipul întreg (pe 4 octeți), tipul 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, cmmm, numere prime între ele, simplificarea fracțiilor)
- Calculul unor expresii (de exemplu, factorial, ridicare la putere)

##### **2. Generări de șiruri**

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

##### **3. Fișiere text**

##### **4. Prelucrări de șiruri de numere citite succesiv, fără memorarea lor**

- Prelucrări ce necesită stocarea ultimului element (de exemplu, determinare maxim/minim, primele două maxime sau minime)
- Prelucrări ce necesită stocarea ultimelor  $p$  ( $p=2, 3, ..$ ) elemente (de exemplu, cea mai lungă subsecvență cu anumite proprietăți, numărarea secvențelor cu anumite proprietăți)

#### **Doar pentru etapa națională**

##### **5. Tablouri unidimensionale**

- Prelucrări elementare (de exemplu, parcurgere, inversare, verificare proprietăți)
- Căutarea liniară a unor valori
- Vectori caracteristici/ de frecvență
- Algoritmi de sortare în complexitate pătratică (sortarea prin selecție, sortarea prin inserție, metoda bulelor)
- Sortare prin numărare (folosind vectori de frecvență)

#### **Clasa a VI-a**

##### **1. Tipuri simple de date (întregi, reale, char)**

##### **2. Sisteme de numerație și reguli de conversie**

##### **3. Aritmetica modulară (adunări, scăderi, înmulțiri)**

##### **4. Divizibilitate**

- Ciurul lui Eratostene
- Descompunerea numerelor naturale în factori primi

## 5. Tablouri unidimensionale (vectori)

- Operații cu mulțimi
- Căutare binară
- Precalcularea unor informații pentru prefixe/sufixe în tablouri unidimensionale (de exemplu sume parțiale, maxime parțiale)
- Probleme cu secvențe de valori (de exemplu, determinarea unei secvențe maxime cu o anumită proprietate, numărarea secvențelor, prelucrarea secvențelor de lungime fixată ce nu implică stive, cozi sau alte structuri de date avansate)
- Căutarea aparițiilor unei subsecvențe într-o secvență de valori în timp pătratic
- Interclasarea tablourilor unidimensionale

### Doar pentru etapa națională

## 6. Tablouri bidimensionale

- Prelucrări elementare ale tablourilor bidimensionale (de exemplu, parcurgeri pe linii/coloane/diagonale/în spirală, generări, transpunere, bordare)
- Prelucrări specifice tablourilor bidimensionale pătratice (de exemplu, diagonale și zone determinate de diagonale)
- Căutări secvențiale în tablouri bidimensionale (de exemplu, a unui element, a unei secvențe de valori, a unei submatrice)
- Utilizarea vectorilor de direcție

## 7. Simulări

- reprezentarea sistemului de simulat, starea sistemului
- bucla de evenimente ce modifică starea sistemului

## Clasa a VII-a

### 1. Funcții

- Declaraire, definiție, apel
- Variabile locale, variabile globale
- Transmiterea parametrilor prin valoare și prin referință

### 2. Tablouri

- Tehnica *Two Pointers*
- Tablouri de diferențe - *Difference Arrays*
- Determinarea secvenței de sumă maximă
- Determinarea elementului majoritar
- Precalcularea unor informații în tablouri bidimensionale (de exemplu sume parțiale pe prefixe/sufixe de linii/coloane, suma elementelor dintr-o submatrice cu unul dintre colțuri fixat în unul dintre colțurile matricei)
- Tablouri multidimensionale

### 3. Tipuri de date neomogene (struct)

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

### 5. Metoda *Greedy*

### Doar pentru etapa națională

### 6. Operații cu numere mari

- Adunarea numerelor mari
- Scăderea numerelor mari
- Înmulțirea unui număr mare cu un număr natural
- Împărțirea cu rest a unui număr mare la un număr natural nenul

### 7. Algoritm de exponențiere rapidă

### 8. Stiva. Aplicații specifice

## Clasa a VIII-a

### 1. Șiruri de caractere. Funcții specifice

### 2. Generarea elementelor combinatoriale prin algoritmi de tip succesori

- Submulțimi
- Produs cartezian
- Permutări
- Combinări
- Aranjamente
- Utilizarea funcțiilor din biblioteca STL pentru permutări

### Doar pentru etapa națională

### 3. Coadă. Aplicații specifice

### 4. Coadă cu dublă prioritate (deque). Aplicații specifice

### 5. Elemente de geometrie

- sistemul de coordonate cartezian în plan
- puncte în planul cartezian
- distanța dintre două puncte
- arii

## Barajul de selecție a lotului național lărgit

1. Operații pe biți
2. Indicatorul lui Euler
3. Tablouri de diferențe (*Difference Arrays*) 2D
4. Recursivitate
5. Algoritm de fill
6. Algoritm lui *Lee*
7. Tehnica *Square root decomposition*
8. Metoda programării dinamice

## Note

- Exceptând clasa a V-a, programa fiecărei clase include și programele pentru toate clasele precedente.
- Barajul de selecție a lotului național lărgit include programele pentru clasele V-VIII, precum și temele suplimentare specificate.
- Pentru barajele de selecție a echipelor reprezentative ale României vor fi abordate teme suplimentare.

## Liceu

### 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, cmmm, 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. Reprezentarea numerelor naturale și întregi în memoria calculatorului. Operații pe biți

#### 5. Fișiere text

#### 6. 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*)

#### 7. Tablouri bidimensionale (matrice)

- Prelucrări elementare ale tablourilor bidimensionale (de exemplu, parcurgeri, generări, simulări)
- Prelucrări specifice tablourilor bidimensionale pătratice (de exemplu, diagonale și zone determinate de diagonale)
- Sume parțiale pe matrice

#### 8. Structuri de date neomogene (tipul *struct*)

#### 9. Metoda *Greedy*

## Doar pentru etapa națională

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

### 2. Indicatorul lui Euler

### 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. Generări de elemente combinatoriale

- Submulțimi
- Produs cartezian
- Permutări
- Combinări
- Aranjamente

## Clasa a X-a

### 1. Șiruri de caractere. Funcții specifice

### 2. Structuri de date

- Stiva (*stack*). Aplicații specifice
- Coada (*queue*). Aplicații specifice. Algoritmul lui *Lee*
- *Deque*. Aplicații specifice
- Lista (liste simplu și dublu înlanțuite 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*.

### 3. Operații cu numere mari

- Adunarea numerelor mari
- Scăderea numerelor mari
- Înmulțirea unui număr mare cu un număr natural
- Împărțirea cu rest a unui număr mare la un număr natural nenul

### 4. Elemente de combinatorică

- Numărarea elementelor combinatoriale (submulțimi, produs cartezian, permutări, aranjamente, combinări, parantezări, partiții)
- Determinarea numărului de ordine pentru elementele combinatoriale
- Aritmetică modulară (adunare, scădere, înmulțire, invers modular - pentru modulo număr prim)

## 5. Recursivitate

## 6. Metoda Divide et Impera

### Doar pentru etapa națională

#### 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 (varianta elementară și varianta în plan)

#### 3. Metoda programării dinamice

- Probleme de numărare
- Probleme de determinare a unei soluții optime
- Memoizarea relațiilor de recurență

## Clasele XI-XII

#### 1. Metoda programării dinamice

- Programare dinamică pe arbori și grafuri
- Programare dinamică pe stări exponențiale

#### 2. 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 hamiltonian, graf eulerian, 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
  - Parcurgerea grafurilor în lățime (BFS), în adâncime (DFS), parcurgerea euleriană
  - Determinarea componentelor conexe ale unui graf neorientat
  - Determinarea componentelor tare conexe ale unui graf orientat. Algoritmul *Kosaraju-Sharir*. Grafurile componentelor tare-conexe
  - Determinarea matricei lanțurilor/drumurilor (algoritmul *Roy-Warshall*)
  - Descompunerea unui graf orientat fără circuite pe niveluri. Sortare topologică
  - Determinarea drumurilor de cost minim într-un graf. Algoritmul lui *Dijkstra*, algoritmul *Bellman-Ford*, algoritmul *Roy-Floyd*
  - Determinarea unui lanț/ciclu hamiltonian
  - Determinarea unui lanț/ciclu eulerian
- **Arbori**
  - Definiție, proprietăți

- Arbori parțiali
- Arbori parțiali de cost minim (algoritmul lui *Kruskal* și algoritmul lui *Prim*)

### 3. 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*

### Doar pentru etapa națională

#### 1. Algoritmi pe grafuri

- Determinarea punctelor de articulație, a punților și descompunerea grafurilor în componente biconexe.
- Algoritmul lui Dial (optimizarea algoritmului lui Dijkstra pentru grafuri cu ponderi dintr-un interval mic de valori)

#### 2. Structuri de date arborescente

- Determinarea celui mai apropiat strămoș comun a două noduri dintr-un arbore (*lowest common ancestor* - LCA)
- Determinarea diametrului unui arbore
- Arbori indexați binar
- Arbori de intervale

#### 3. Square Root Decomposition. Algoritmul lui Mo

#### 4. Range Minimum Query (RMQ)

#### 5. Tehnica Meet in the Middle

#### 6. Ridicarea la putere a matricilor în timp logaritmic. Rezolvarea recurențelor liniare

#### 7. Principiul includerii și excluderii. Funcția Mobius

### Note

- Exceptând clasa a IX-a, programa fiecărei clase include și programele pentru toate clasele precedente.
- Barajele de selecție a lotului național lărgit vor include programele pentru clasele IX-XII
- Pentru barajele de selecție a echipelor reprezentative ale României vor fi abordate teme suplimentare.