

Programa pentru Concursul Județean de Informatică „Future for Future” – Secțiunea Liceu

Observație. Cu excepția clasei a IX-a, programa fiecărei clase include și programele pentru toate clasele de liceu precedente.

Clasa a IX-a

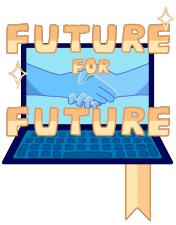
- Tipuri simple de date. Tipuri de întregi (`int`, `long long`), tipul real (`double`), tipul caracter (`char`), tipul logic (`bool`)
- Structurile liniară, alternativă, repetitivă
- Prelucrarea cifrelor numerelor naturale scrise în baza 10
- Calculul unor expresii folosind operatorii aritmetici de bază și alte operații (de exemplu: factorial, ridicarea la putere cu exponent natural)
- Algoritmul de ridicare la putere în timp logaritmic (exponențiere rapidă)
- Divizibilitate (divizorii numerelor naturale, numere prime, descompunerea numerelor naturale în factori primi, determinarea c.m.m.d.c. și c.m.m.m.c. prin algoritmul lui Euclid, numere prime între ele, simplificarea fracțiilor)
- Generarea șirurilor pe baza unor reguli
- Șirul lui Fibonacci și alte șiruri recurente
- Sisteme de numerație și reguli de conversie
- Citirea/scrierea datelor dintr-un/într-un fișier
- Tablouri unidimensionale (vectori):
 - Prelucrări elementare (de exemplu: parcurgere, inversare, verificarea unor proprietăți)
 - Căutarea secvențială și căutarea binară
 - Vectori caracteristici/de frecvență
 - Algoritmi de sortare în complexitate pătratică (sortarea prin selecție, sortarea prin inserție, metoda bulelor)
 - Sortarea prin numărare (folosind vectori de frecvență)
 - Operații cu multimi
 - Precalculatează unor informații pentru prefixe/sufixe în tablouri unidimensionale (de exemplu: sume partiale, maxime partiale)
 - Probleme cu secvențe de valori (de exemplu: determinarea unei secvențe de lungime maximă cu o anumită proprietate, numărarea secvențelor, prelucrarea secvențelor de lungime fixată ce nu implică stive, cozi sau alte structuri de date avansate, determinarea secvenței de sumă maximă, tablouri unidimensionale de diferențe – *Difference Arrays*, tehnica *Two Pointers*)



- Căutarea aparițiilor unei subsecvențe într-o secvență de valori în timp pătratic
- Interclasarea tablourilor unidimensionale
- Determinarea elementului majoritar
- Ciurul lui Eratostene
- Metoda *Greedy*
- Tablouri bidimensionale (matrice):
 - Prelucrări elementare ale tablourilor bidimensionale (de exemplu: parcurgeri pe linii/coloane/diagonale/în spirală, generări, simulări, 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
 - Precalcularea unor informații în tablouri bidimensionale (de exemplu: sume parțiale pe prefixe/sufixe de linii/coloane, suma elementelor dintr-o submatrix)

Clasa a X-a

- Tablouri bidimensionale de diferențe
- Siruri de caractere. Funcții specifice
- Funcții:
 - Declarare, definire, apel
 - Variabile locale și globale
 - Transmiterea parametrilor prin valoare și prin referință
- Structuri de date:
 - Stiva
 - Coada. Algoritmul lui Lee
 - Coada cu două capete (*deque*)
 - Liste simplu și dublu înlántuite
- *Standard Template Library* (STL):
 - Utilizarea funcțiilor din biblioteca `algorithm` pentru sortare și căutare
 - Structuri de date (`pair`, `vector`, `list`, `stack`, `queue`, `deque`, `priority_queue`, `set`, `unordered_set`, `multiset`, `map`, `unordered_map`, `multimap`, `bitset`)
- Reprezentarea numerelor naturale în memoria calculatorului. Operații pe biți
- Operații cu numere mari:
 - Adunarea și scăderea numerelor mari
 - Înmulțirea unui număr mare cu un număr natural



- Aritmetică modulară (operațiile de adunare, scădere, înmulțire; inversul modular – pentru modulo număr prim)
- Elemente de combinatorică (generare și numărare de submulțimi, produs cartezian, permutări, aranjamente, combinări)
- Recursivitate
- Metoda *Divide et impera*

Clasele XI-XII

- Elemente de geometrie (sistemul de coordonate carteziene, 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, înfășurătoarea convexă)
- Elemente de teoria grafurilor:
 - 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/arcelor)
 - Grafuri ponderate. Reprezentarea grafurilor ponderate (matricea costurilor, liste de adiacență cu costuri, lista muchiilor/arcelor cu costuri)
 - Algoritmi de prelucrare a grafurilor:
 - * Parcurgea grafurilor în lățime (BFS), în adâncime (DFS)
 - * Determinarea componentelor conexe ale unui graf neorientat
 - * Determinarea componentelor tare conexe ale unui graf orientat. Algoritmul Kosaraju-Sharir. Graful componentelor tare conexe
 - * Determinarea matricei lanțurilor/drumurilor. Algoritmul Roy-Warshall
 - * Descompunerea unui graf orientat fără circuite pe niveluri. Sortarea topologică
 - * Determinarea drumurilor de cost minim într-un graf. Algoritmul lui Dijkstra, algoritm Bellman-Ford, algoritm Roy-Floyd
 - Arbori:
 - * Definiție, proprietăți
 - * Arbori parțiali
 - * Arbori parțiali de cost minim. Algoritmul lui Kruskal, algoritmul lui Prim
- Structuri de date arborescente:
 - Operații pe structuri de date (interrogări, actualizări)
 - Arbori cu rădăcină (definiție, proprietăți, reprezentarea arborilor cu rădăcină)
 - Arbori binari (definiție, proprietăți, reprezentarea arborilor binari)
 - Arbori binari compleți (definiție, proprietăți, reprezentare secvențială)



-
- *Heap-uri* (definiție, proprietăți, operații specifice: inserarea unui nod, extragerea nodului de cheie maximă/minimă)
 - Arbo里 binari de căutare (definiție, proprietăți, operații specifice: inserarea unui nod, ștergerea unui nod, căutarea unui element)
 - Păduri de mulțimi disjuncte (DSU). Algoritmii *Union-Find*
 - Metoda *Backtracking*
 - Metoda programării dinamice:
 - Probleme de numărare
 - Probleme de determinare a unei soluții optime
 - Memoizare
 - Programare dinamică pe arbori și grafuri