

# TEMATICA

## A. MATEMATICĂ:

### *Algebră:*

Ecuatii și sisteme de ecuații de gradul I și II; funcția afină; funcția de gradul II;

a) Funcția exponențială și logaritmică; ecuații și inecuații exponențiale și logaritmice; progresii aritmetice și geometrice; elemente de combinatorică; binomul lui Newton; mulțimea numerelor complexe: operații cu numere complexe sub formă algebrică, modulul unui număr complex, ecuații bipătrate;

b) Matrice, determinanți, sisteme de ecuații liniare.

### *Elemente de analiză matematică:*

a) Șiruri de numere reale, convergență; limite pentru funcții reale, asimptotele la graficul funcțiilor reale; continuitatea funcțiilor reale; derivabilitate;

b) Primitive; integrala definită: metode de calcul a integralelor definite.

### *Elemente de geometrie și trigonometrie:*

a) Elemente de trigonometrie: cercul trigonometric, funcțiile trigonometrice directe, formule trigonometrice; vectori în plan: segment orientat, vectorul de poziție al unui punct, operații cu vectori, condiția de coliniaritate, produsul scalar a doi vectori, condiția de perpendicularitate, vectorul de poziție al unui punct care împarte un segment orientat într-un raport dat;

b) Formula distanței, mijlocul unui segment, coordonatele centrului de greutate; ecuația dreptei, condiția de paralelism și perpendicularitate, drepte concurente; distanța de la un punct la o dreaptă, aria triunghiului.

## B. INFORMATICĂ

### COMPETENȚE DE EVALUAT

– construirea algoritmilor corespunzători unor prelucrări elementare și reprezentarea lor în limbajul de programare C++;

– analiza rezolvării unei probleme prin urmărirea evoluției valorilor variabilelor prelucrate de algoritmul corespunzător;

– abstractizarea rezolvării prin construirea unor algoritmi echivalenți;

– identificarea și utilizarea tipurilor de date predefinite specifice unui limbaj de programare;

– definirea și utilizarea unor tipuri de date proprii;

– identificarea și utilizarea operatorilor predefiniți elementari;

– identificarea și utilizarea subprogramelor predefinite elementare;

– identificarea și utilizarea regulilor sintactice specifice limbajului de programare studiat;

– definirea și apelul unor subprograme proprii cu înțelegerea mecanismelor de transfer prin intermediul parametrilor;

– identificarea proprietăților unor structuri de date necesare în rezolvarea problemelor cu ajutorul calculatorului și utilizarea unor modele de memorare a acestora;

– organizarea datelor ce intervin în rezolvarea unei probleme utilizând structuri de date adecvate;

– organizarea etapelor de prelucrare ce formează un algoritm utilizând structuri de control și module de program;

– folosirea unor metode sistematice de rezolvare pentru probleme de generare;

– analiza unor algoritmi echivalenți de rezolvare a unei probleme în vederea alegerii algoritmului optim.

## CONȚINUTURI

1. Algoritmi
  - 1.1. Noțiunea de algoritm, caracteristici
  - 1.2. Date, variabile, expresii, operații
  - 1.3. Structuri de bază (liniară, alternativă și repetitivă)
  - 1.4. Descrierea algoritmilor
2. **Elementele de bază ale limbajului de programare C++**
  - 2.1. Vocabularul limbajului
  - 2.2. Constante. Identificatori
  - 2.3. Noțiunea de tip de dată. Operatori aritmetici, logici, relaționali
  - 2.4. Definiția tipurilor de date
  - 2.5. Variabile. Declararea variabilelor
  - 2.6. Definiția constantelor
  - 2.7. Structura programelor. Comentarii
  - 2.8. Expresii. Instrucțiunea de atribuire
  - 2.9. Citirea /scrierea datelor
  - 2.10. Structuri de control (instrucțiunea compusă, structuri alternative și repetitive)
3. **Subprograme predefinite**
  - 3.1. Subprograme. Mecanisme de transfer prin intermediul parametrilor
  - 3.2. Proceduri și funcții predefinite
4. **Tipuri structurate de date**
  - 4.1. Tipul tablou
  - 4.2. Tipul șir de caractere –operatori, proceduri și funcții predefinite pentru: citire, afișare, concatenare, căutare, extragere, inserare, eliminare și conversii (șir ↔ valoare numerică)
  - 4.3. Tipul înregistrare
5. **Fișiere text**
  - 5.1. Fișiere text. Tipuri de acces
  - 5.2. Proceduri și funcții predefinite pentru fișiere text
6. **Algoritmi elementari**
  - 6.1. Probleme care operează asupra cifrelor unui număr
  - 6.2. Divizibilitate. Numere prime. Algoritmul lui Euclid
  - 6.3. Șirul lui Fibonacci. Calculul unor sume cu termenul general dat
  - 6.4. Determinare minim/maxim
  - 6.5. Metode de ordonare ( metoda bulelor, inserției, selecției, numărării)
  - 6.6. Interclasare
  - 6.7. Metode de căutare (secvențială, binară)
  - 6.8. Analiza complexității unui algoritm (considerând criteriile de eficiență *durata de executare* și *spațiu de memorie utilizat*)
7. **Subprograme definite de utilizator**
  - 7.1. Proceduri și funcții
    - declarare și apel
    - parametri formali și parametri efectivi
    - parametri transmiși prin valoare, parametri transmiși prin referință
    - variabile globale și variabile locale, domeniu de vizibilitate
  - 7.2. Proiectarea modulară a rezolvării unei probleme
8. **Recursivitate**
  - 8.1. Prezentare generală
  - 8.2. Proceduri și funcții recursive
9. **Metoda backtracking (iterativă sau recursivă)**
  - 9.1. Prezentare generală
  - 9.2. Probleme de generare. Oportunitatea utilizării metodei backtracking
10. **Generarea elementelor combinatoriale**
  - 10.1. Permutări, aranjamente, combinații

10.2. Prodos cartezian, submulțimi

## 11. Grafuri

### 0.1. Grafuri neorientate

- terminologie (nod/vârf, muchie, adiacență, incidență, grad, lanț, lanț elementar, ciclu, ciclu elementar, lungime, subgraf, graf parțial)
- proprietăți (conex, componentă conexă, graf complet, hamiltonian, eulerian)
- metode de reprezentare (matrice de adiacență, liste de adiacență)

### 0.2. Grafuri orientate

- terminologie (nod/vârf, arc, adiacență, incidență, grad intern și extern, drum, drum elementar, circuit, circuit elementar, lungime, subgraf, graf parțial)
- proprietăți (tare conexitate, componentă tare conexă)
- metode de reprezentare (matrice de adiacență, liste de adiacență)

### 0.3. Arbori

- terminologie (nod, muchie, rădăcină, descendent, descendent direct/fiu, ascendent, ascendent direct/părinte, frați, nod terminal, frunză)
- metode de reprezentare în memorie (matrice de adiacență, liste „de descendenți”, vector „de tați”).

## C. LIMBA ENGLEZĂ

*Domeniile tematice:*

- Domeniul privat (viața personală): identitate, idealuri și proiecte, sentimente, modă, stil de viață, activități în timpul liber, strategii de studiu, relații interumane, comportament social;
- Domeniul social (societate): tinerii, familia și comunitatea, tinerii și idoli lor, relații/conflicte între tineri/generații, toleranță, sănătate, comunicare pe diverse canale/în diverse situații (cumpărături, turism, servicii), mass-media, mediul înconjurător; democrație, civism, drepturile omului;
- Domeniul educației și profesie: școală, ocupații, profesii, piața muncii;
- Domeniul știință și cultură: patrimoniul național, britanic și american – personalități, opere, obiceiuri, tradiții, sărbători, evenimente.

*Elemente de gramatică:*

- Substantivul – pluralul substantivelor; substantive defective de număr;
- Adjectivul – comparația intensivă;
- Verbul – timpurile verbale; diateza pasivă; verbele modale/ modalitatea; infinitivul/ participiul/ gerunziul; construcții cu infinitivul și cu participiul;
- Adverbul – gradele de comparație;
- Cuvinte de legătură – prepoziții, conjuncții, locuțiuni;
- Articolul – hotărât, nehotărât, zero, omisiunea articolului; cazuri speciale de utilizare a articolului;
- Sintaxa – ordinea cuvintelor în propoziție; tipuri de propoziții (enuțiative, interogative etc.); propozițiile condiționale; condiționalele I, II, III; corespondența timpurilor; vorbirea directă/ indirectă.