

**CONTINUTURILE PROGRAMELOR ȘCOLARE  
PENTRU SUSȚINEREA EXAMENELOR DE DIFERENȚE  
LA TRANSFERUL ELEVILOR – disciplina INFORMATICĂ**

**1. Continuturile examenelor de diferențe pentru transferul elevilor care în clasa a IX-a au fost la profil uman iar în clasa a X-a se transfera la profil real, specializarea matematică-informatică neintensiv informatică :**

- Algoritmi
  - i. Etapele rezolvării problemelor. Exemple
  - ii. Noțiunea de algoritm. Caracteristici. Exemple.
  - iii. Date cu care lucrează algoritmi (constante, variabile, expresii).
  - iv. Operații asupra datelor (aritmetice, logice, relaționale).
- Limbajul pseudocod
  - i. Reprezentarea algoritmilor. Pseudocod.
  - ii. Principiile programării structurate. Structuri de bază:
    1. structura liniară
    2. structura alternativă
    3. structura repetitivă
- Algoritmi elementari
  - i. Prelucrarea numerelor :
    1. prelucrarea cifrelor unui număr (de exemplu, suma cifrelor, testarea proprietății de palindrom etc.)
    2. probleme de divizibilitate (de exemplu, determinarea divizorilor unui număr, determinarea c.m.m.d.c./c.m.m.m.c., testare primalitate etc.)
    3. calculul unor expresii simple (sume, produse etc.)
  - ii. Prelucrarea unor secvențe de valori
    1. determinare minim/maxim
    2. verificarea unei proprietăți (de exemplu, dacă toate elementele din secvență sunt numere perfecte etc.)
  - iii. calculul unor expresii în care intervin valori din secvență (de exemplu: numărarea elementelor pare/impare etc)
  - iv. generarea șirurilor recurente (de exemplu: șirul Fibonacci)

**2. Conținuturile examenelor de diferențe pentru transferul elevilor care în clasa a IX-a au fost la profil real (specializările matematică-informatică sau științele naturii) iar în clasa a X-a se transfera la profil real, specializarea matematică-informatică intensiv informatică :**

- Elemente de bază ale limbajului C++
  - Mediul limbajului de programare studiat
  - Prezentare generală
  - Editarea programelor sursă
  - Compilare, rulare, depanare
  - Elementele de bază ale limbajului de programare
  - Noțiuni introductive
  - Structura programelor
  - Vocabularul limbajului
  - Tipuri simple de date (standard)
  - Constante, variabile, expresii
  - Citirea/scrierea datelor
  - Structuri de control
  - Structura liniară
  - Structura alternativă
  - Structuri repetitive
- Fișiere text: Definiție, operații specifice
- Tipuri structurate de date. Tipul tablou
  - Tablouri unidimensionale
  - Tablouri bidimensionale
- Algoritmi fundamentali de prelucrare a datelor structurate în tablouri
  - căutare secvențială, căutare binară
  - sortare
  - interclasare
  - parcurgerea tablourilor bidimensionale pe linii/coloane etc.

3. **Continuturile examenelor de diferențe pentru transferul elevilor care în clasa a IX-a au fost la profil uman iar în clasa a X-a se transfera la profilul real, specializarea matematică-informatică intensiv informatică :**
- Algoritmi
    - Etapele rezolvării problemelor. Exemple
    - Noțiunea de algoritm. Caracteristici. Exemple.
    - Date cu care lucrează algoritmi (constante, variabile, expresii).
    - Operații asupra datelor (aritmetice, logice, relaționale).
  - Limbajul pseudocod
    - Reprezentarea algoritmilor. Pseudocod.
    - Principiile programării structurate. Structuri de bază:
      - structura liniară
      - structura alternativă
      - structura repetitivă
  - Algoritmi elementari
    - Prelucrarea numerelor :
      - prelucrarea cifrelor unui număr (de exemplu, suma cifrelor, testarea proprietății de palindrom etc.)
      - probleme de divizibilitate (de exemplu, determinarea divizorilor unui număr, determinarea c.m.m.d.c./c.m.m.m.c., testare primalitate etc.)
      - calculul unor expresii simple (sume, produse etc.)
    - Prelucrarea unor secvențe de valori
      - determinare minim/maxim
      - verificarea unei proprietăți (de exemplu, dacă toate elementele din secvență sunt numere perfecte etc.)
      - calculul unor expresii în care intervin valori din secvență (de exemplu: numărarea elementelor pare/impare etc)
      - generarea șirurilor recurente (de exemplu: șirul Fibonacci)
  - Elemente de bază ale limbajului C++
    - Mediul limbajului de programare studiat
    - Prezentare generală
    - Editarea programelor sursă
    - Compilare, rulare, depanare
    - Elementele de bază ale limbajului de programare
    - Noțiuni introductive
    - Structura programelor
    - Vocabularul limbajului
    - Tipuri simple de date (standard)
    - Constante, variabile, expresii
    - Citirea/scrierea datelor
    - Structuri de control
    - Structura liniară
    - Structura alternativă
    - Structuri repetitive
  - Fișiere text: Definiție, operații specifice
  - Tipuri structurate de date. Tipul tablou
    - Tablouri unidimensionale
    - Tablouri bidimensionale
  - Algoritmi fundamentali de prelucrare a datelor structurate în tablouri
    - căutare secvențială, căutare binară
    - sortare
    - interclasare
    - parcurgerea tablourilor bidimensionale pe linii/coloane etc.

4. **Continuturile examenelor de diferențe pentru transferul elevilor care în clasa a X-a au fost la profil uman iar în clasa a XI-a se transfera la profil real, specializarea matematică-informatică intensiv informatică :**

- **Algoritmi**
  - Etapele rezolvării problemelor. Exemple
  - Noțiunea de algoritm. Caracteristici. Exemple.
  - Date cu care lucrează algoritmi (constante, variabile, expresii).
  - Operații asupra datelor (aritmetice, logice, relaționale).
- **Limbaajul pseudocod**
  - Reprezentarea algoritmilor. Pseudocod.
  - Principiile programării structurate. Structuri de bază:
    - structura liniară
    - structura alternativă
    - structura repetitivă
- **Algoritmi elementari**
  - Prelucrarea numerelor :
    - prelucrarea cifrelor unui număr (de exemplu, suma cifrelor, testarea proprietății de palindrom etc.)
    - probleme de divizibilitate (de exemplu, determinarea divizorilor unui număr, determinarea c.m.m.d.c./c.m.m.m.c., testare primalitate etc.)
    - calculul unor expresii simple (sume, produse etc.)
  - Prelucrarea unor secvențe de valori
    - determinare minim/maxim
    - verificarea unei proprietăți (de exemplu, dacă toate elementele din secvență sunt numere perfecte etc.)
    - calculul unor expresii în care intervin valori din secvență (de exemplu: numărarea elementelor pare/impare etc)
    - generarea șirurilor recurente (de exemplu: șirul Fibonacci)
- **Elemente de bază ale limbajului C++**
  - Mediul limbajului de programare studiat
  - Prezentare generală
  - Editarea programelor sursă
  - Compilare, rulare, depanare
  - Elementele de bază ale limbajului de programare
  - Noțiuni introductive
  - Structura programelor
  - Vocabularul limbajului
  - Tipuri simple de date (standard)
  - Constante, variabile, expresii
  - Citirea/scrierea datelor
  - Structuri de control
  - Structura liniară
  - Structura alternativă
  - Structuri repetitive
- **Fișiere text: Definiție, operații specifice**
- **Tipuri structurate de date. Tipul tablou**
  - **Tablouri unidimensionale**
  - Tablouri bidimensionale
- **Algoritmi fundamentali de prelucrare a datelor structurate în tablouri**
  - căutare secvențială, căutare binară
  - sortare
  - interclasare
  - parcurgerea tablourilor bidimensionale pe linii/coloane etc.
- **Subprograme**
  - Declararea, definiția și apelul subprogramelor
  - Transferul parametrilor la apel
  - Returnarea valorilor de către subprograme
  - Variabile locale și globale
  - Modularizarea unui program prin intermediul subprogramelor
- **Tipuri structurate de date – șiruri de caractere**

- Funcții standard la nivel de caracter și la nivel de structură
- Algoritmi fundamentali: prelucrarea unui șir de caractere la nivel de caracter și la nivel de structură, utilizând funcții specifice
- **Tipuri structurate de date – tipul înregistrare**
  - Înregistrare (structură/record)
  - Algoritmi fundamentali: prelucrarea unei înregistrări/ structuri la nivel de câmp și la nivel de structură
- **Tipuri structurate de date – lista, stiva, coada - operații specifice (creare, adaugare, ștergere, inserare etc)**
- **Subprograme recursive**
  - Mecanismul de realizare a recursivității
  - Compararea implementării recursive, a unui algoritm, cu cea iterativă, avantaje și dezavantaje ale celor două tipuri de implementări.
  - Algoritmi elementari implementați recursiv
- **Metoda de programare Divide et Impera**
  - Sortarea eficientă a unei mulțimi de valori aplicând metoda Divide et Impera (sortarea rapidă, sortarea prin interclasare)
  - Căutarea eficientă a unui element într-o mulțime ordonată aplicând metoda Divide et Impera (căutarea binară)
- **Structuri de date alocate dinamic**
  - Alocarea dinamică a memoriei (operații și mecanisme specifice)
  - Structuri de date alocate dinamic (definiții, utilitate)
  - Liste simplu înlănțuite
  - Liste particulare (stiva, coada)
  - Liste dublu înlănțuite
  - Liste circulare
  - Operații elementare pe liste înlănțuite (inserare element, ștergere element, parcurgere, căutarea unui element într-o listă)

**5. Continuturile examenelor de diferențe pentru transferul elevilor care în clasa a X-a au fost la profil real, specializarea matematică-informatică iar în clasa a XI-a se transfera la profil real, specializarea matematică-informatică intensiv informatică :**

- **Subprograme**
  - Declararea, definirea și apelul subprogramelor
  - Transferul parametrilor la apel
  - Returnarea valorilor de către subprograme
  - Variabile locale și globale
  - Modularizarea unui program prin intermediul subprogramelor
- **Tipuri structurate de date – șiruri de caractere**
  - Funcții standard la nivel de caracter și la nivel de structură
  - Algoritmi fundamentali: prelucrarea unui șir de caractere la nivel de caracter și la nivel de structură, utilizând funcții specifice
- **Tipuri structurate de date – tipul înregistrare**
  - Înregistrare (structură/record)
  - Algoritmi fundamentali: prelucrarea unei înregistrări/ structuri la nivel de câmp și la nivel de structură
- **Tipuri structurate de date – lista, stiva, coada - operații specifice (creare, adaugare, ștergere, inserare etc)**
- **Subprograme recursive**
  - Mecanismul de realizare a recursivității
  - Compararea implementării recursive, a unui algoritm, cu cea iterativă, avantaje și dezavantaje ale celor două tipuri de implementări.
  - Algoritmi elementari implementați recursiv
- **Metoda de programare Divide et Impera**
  - Sortarea eficientă a unei mulțimi de valori aplicând metoda Divide et Impera (sortarea rapidă, sortarea prin interclasare)
  - Căutarea eficientă a unui element într-o mulțime ordonată aplicând metoda Divide et Impera (căutarea binară)
- **Structuri de date alocate dinamic**
  - Alocarea dinamică a memoriei (operații și mecanisme specifice)
  - Structuri de date alocate dinamic (definiții, utilitate)
  - Liste simplu înlănțuite
  - Liste particulare (stiva, coada)
  - Liste dublu înlănțuite
  - Liste circulare
  - Operații elementare pe liste înlănțuite (inserare element, ștergere element, parcurgere, căutarea unui element într-o listă)

**6. Continuturile examenelor de diferențe pentru transferul elevilor care în clasa a X-a au fost la profil uman iar în clasa a XI-a se transfera la profil real, specializarea matematică-informatică neintensiv informatică :**

- **Algoritmi**
  - Etapele rezolvării problemelor. Exemple
  - Noțiunea de algoritm. Caracteristici. Exemple.
  - Date cu care lucrează algoritmi (constante, variabile, expresii).
  - Operații asupra datelor (aritmetice, logice, relaționale).
- **Limbaajul pseudocod**
  - Reprezentarea algoritmilor. Pseudocod.
  - Principiile programării structurate. Structuri de bază:
    - structura liniară
    - structura alternativă
    - structura repetitivă
- **Algoritmi elementari**
  - Prelucrarea numerelor :
    - prelucrarea cifrelor unui număr (de exemplu, suma cifrelor, testarea proprietății de palindrom etc.)
    - probleme de divizibilitate (de exemplu, determinarea divizorilor unui număr, determinarea c.m.m.d.c./c.m.m.m.c., testare primalitate etc.)
    - calculul unor expresii simple (sume, produse etc.)
  - Prelucrarea unor secvențe de valori
    - determinare minim/maxim
    - verificarea unei proprietăți (de exemplu, dacă toate elementele din secvență sunt numere perfecte etc.)
    - calculul unor expresii în care intervin valori din secvență (de exemplu: numărarea elementelor pare/impare etc)
    - generarea șirurilor recurente (de exemplu: șirul Fibonacci)
- **Elemente de bază ale limbajului C++**
  - Mediul limbajului de programare studiat
  - Prezentare generală
  - Editarea programelor sursă
  - Compilare, rulare, depanare
  - Elementele de bază ale limbajului de programare
  - Noțiuni introductive
  - Structura programelor
  - Vocabularul limbajului
  - Tipuri simple de date (standard)
  - Constante, variabile, expresii
  - Citirea/scrierea datelor
  - Structuri de control
  - Structura liniară
  - Structura alternativă
  - Structuri repetitive
- **Fișiere text: Definiție, operații specifice**
- **Tipuri structurate de date. Tipul tablou**
  - **Tablouri unidimensionale**
- **Algoritmi fundamentali de prelucrare a datelor structurate în tablouri**
  - căutare secvențială, căutare binară
  - sortare
  - interclasare
- element minim/maxim