

La simularea judeteana din decembrie sunt valabile programele pentru sesiunea de Bacalaureat 2026

Programa Informatica Bac -specializarea mate-info

Specializarile matematica-informatica si matematica-informatica, intensiv informatica

II. COMPETENTE DE EVALUAT

- construirea algoritmilor corespunzatori unor prelucrari elementare si reprezentarea lor prin intermediul programelor pseudocod si programelor scrise in limbaj de programare (Pascal sau C/C++, la alegere);
- analiza rezolvarii unei probleme prin urmarirea evolutiei valorilor variabilelor prelucrate de algoritmul corespunzator;
- abstractizarea rezolvarii prin construirea unor algoritmi echivalenti;
- identificarea si utilizarea tipurilor de date predefinite specifice unui limbaj de programare;
- definirea si utilizarea unor tipuri de date proprii;
- identificarea si utilizarea operatorilor predefiniti elementari;
- identificarea si utilizarea subprogramelor predefinite elementare;
- identificarea si utilizarea regulilor sintactice specifice limbajului de programare studiat;
- definirea si apelul unor subprograme proprii cu intelegerea mecanismelor de transfer prin intermediul parametrilor;
- identificarea proprietatilor unor structuri de date necesare in rezolvarea problemelor cu ajutorul calculatorului si utilizarea unor modele de memorare a acestora;
- organizarea datelor ce intervin in rezolvarea unei probleme utilizand structuri de date adecvate;
- organizarea etapelor de prelucrare ce formeaza un algoritm utilizand structuri de control si module de program;
- folosirea unor metode sistematice de rezolvare pentru probleme de generare;
- analiza unor algoritmi echivalenti de rezolvare a unei probleme in vederea alegerii algoritmului optim.

III. CONTINUTURI

1. Algoritmi

- 1.1. Notiunea de algoritm, caracteristici
- 1.2. Date, variabile, expresii, operatii
- 1.3. Structuri de baza (liniara, alternativa si repetitiva)
- 1.4. Descrierea algoritmilor (programe pseudocod)

2. Elementele de baza ale unui limbaj de programare (Pascal sau C, la alegere)

- 2.1. Vocabularul limbajului
- 2.2. Constante. Identificatori
- 2.3. Notiunea de tip de data. Operatori aritmetici, logici, relationali
- 2.4. Definirea tipurilor de date
- 2.5. Variabile. Declararea variabilelor
- 2.6. Definirea constantelor
- 2.7. Structura programelor. Comentarii

- 2.8. Expresii. Instructiunea de atribuire
- 2.9. Citirea/scrierea datelor
- 2.10. Structuri de control (instructiunea compusa, structuri alternative si repetitive)

3. Subprograme predefinite

- 3.1. Subprograme. Mecanisme de transfer prin intermediul parametrilor
- 3.2. Proceduri si functii predefinite

4. Tipuri structurate de date

- 4.1. Tipul tablou
- 4.2. Tipul sir de caractere – operatori, proceduri si functii predefinite pentru: citire, afisare, concatenare, cautare, extragere, inserare, eliminare si conversii (sir $\leftrightarrow$ valoare numerica)
- 4.3. Tipul inregistrare

5. Fisiere text

- 5.1. Fisiere text. Tipuri de acces
- 5.2. Proceduri si functii predefinite pentru fisiere text

6. Algoritmi elementari

- 6.1. Probleme care opereaza asupra cifrelor unui numar
- 6.2. Divizibilitate. Numere prime. Algoritmul lui Euclid
- 6.3. Sirul lui Fibonacci. Calculul unor sume cu termenul general dat
- 6.4. Determinare minim/maxim
- 6.5. Metode de ordonare (metoda bulelor, insertiei, selectiei, numararii)
- 6.6. Interclasare
- 6.7. Metode de cautare (secventiala, binara)
- 6.8. Analiza complexitatii unui algoritm (considerand criteriile de eficienta *durata de executare si spatiu de memorie utilizat*)

7. Subprograme definite de utilizator

- 7.1. Proceduri si functii
 - declarare si apel
 - parametri formali si parametri efectivi
 - parametri transmisi prin valoare, parametri transmisi prin referinta
 - variabile globale si variabile locale, domeniu de vizibilitate
- 7.2. Proiectarea modulara a rezolvarii unei probleme

8. Recursivitate

- 8.1. Prezentare generala
- 8.2. Proceduri si functii recursive

9. Metoda backtracking (iterativa sau recursiva)

- 9.1. Prezentare generala
- 9.2. Probleme de generare. Oportunitatea utilizarii metodei backtracking

10. Generarea elementelor combinatoriale

- 10.1. Permutari, aranjamente, combinari
- 10.2. Produs cartezian, submultimi

11. Grafuri

11.1. Grafuri neorientate

- terminologie (nod/varf, muchie, adiacenta, incidenta, grad, lant, lant elementar, ciclu, ciclu elementar, lungime, subgraf, graf partial)
- proprietati (conex, componenta conexa, graf complet, hamiltonian, eulerian)
- metode de reprezentare (matrice de adiacenta, liste de adiacenta)

11.2. Grafuri orientate

- terminologie (nod/varf, arc, adiacenta, incidenta, grad intern si extern, drum, drum elementar, circuit, circuit elementar, lungime, subgraf, graf partial)
- proprietati (tare conexitate, componenta tare conexa)
- metode de reprezentare (matrice de adiacenta, liste de adiacenta)

11.3. Arbori

- terminologie (nod, muchie, radacina, descendent, descendent direct/fiu, ascendent, ascendent direct/parinte, frati, nod terminal, frunza)
- metode de reprezentare in memorie (matrice de adiacenta, liste "de descendenti", vector "de tati")

Programa Informatica Bac – specializarea stiinte ale naturii

II. COMPETENTE DE EVALUAT

- construirea algoritmilor corespunzatori unor prelucrari elementare si reprezentarea lor prin intermediul programelor pseudocod si programelor scrise in limbaj de programare (Pascal sau C/C++, la alegere);
- analiza rezolvării unei probleme prin urmarirea evolutiei valorilor variabilelor prelucrate de algoritmul corespunzator;
- abstractizarea rezolvării prin construirea unor algoritmi echivalenti;
- identificarea si utilizarea tipurilor de date predefinite specifice unui limbaj de programare;
- definirea si utilizarea unor tipuri de date proprii;
- identificarea si utilizarea operatorilor predefiniti elementari;
- identificarea si utilizarea subprogramelor predefinite elementare;
- identificarea si utilizarea regulilor sintactice specifice limbajului de programare studiat;
- identificarea proprietatilor unor structuri de date necesare in rezolvarea problemelor cu ajutorul calculatorului si utilizarea unor modele de memorare a acestora;
- organizarea datelor ce intervin in rezolvarea unei probleme utilizand structuri de date adecvate;
- organizarea etapelor de prelucrare ce formeaza un algoritm utilizand structuri de control;
- analiza unor algoritmi echivalenti de rezolvare a unei probleme in vederea alegerii algoritmului optim.

2. Elementele de baza ale unui limbaj de programare (Pascal sau C, la alegere)

2.1. Vocabularul limbajului

2.2. Constante. Identificatori

2.3. Notiunea de tip de data. Operatori aritmetici, logici, relationali

2.4. Definirea tipurilor de date

2.5. Variabile. Declararea variabilelor

2.6. Definirea constantelor

2.7. Structura programelor. Comentarii

2.8. Expresii. Instruirea 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 – tablouri unidimensionale

5. Fisier text

5.1. Fisier text. Tipuri de acces

5.2. Proceduri și funcții predefinite pentru fisier text

6. Algoritmi elementari

6.1. Probleme care operează asupra cifrelor unui număr

6.2. Divizibilitate. Numere prime. Algoritmul lui Euclid

6.3. Sirul 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 durată de execuție și spațiu de memorie utilizat)

NOTA: Programele de examen sunt realizate în conformitate cu prevederile programelor școlare în vigoare. Subiectele pentru examenul de bacalaureat 2026 se elaborează în baza prevederilor prezentelor programe și nu vizează conținutul unui manual anume.