

Examenul național de bacalaureat 2025

**Proba E. d)
Informatică
Limbaajul C/C++**

Simulare

*Filieră teoretică, profil real, specializare matematică-informatică / matematică-informatică intensiv informatică
Filieră vocațională, profil militar, specializare matematică-informatică*

- Toate subiectele sunt obligatorii. Se acordă zece puncte din oficiu.
- Timpul de lucru efectiv este de trei ore.
- Identificatorii utilizați în rezolvări trebuie să respecte precizările din enunț (**bold**), iar în lipsa unor precizări explicite, notațiile trebuie să corespundă cu semnificațiile asociate acestora (eventual în formă prescurtată). Datele de intrare se consideră corecte, validarea lor nefiind necesară.
- În grafurile din cerințe oricare muchie are extremități distincte și oricare două muchii diferă prin cel puțin una dintre extremități.

SUBIECTUL I

(20 de puncte)

Pentru fiecare dintre itemii de la 1 la 5, scrieți pe foaia de examen litera corespunzătoare răspunsului corect. Fiecare răspuns corect se notează cu 4 puncte.

- Indicați valoarea expresiei C/C++ alăturate.
 $5\%(-3)+7*(-5)\%3$
a. 2 b. -4 c. 0 d. 4
- Subprogramul **f** este definit alăturat.
Indicați ce se afișează pentru **f(121452, 2)**.

```
int f(int n, int c)
{ if(n==0) return 0;
  if(n%10==c) return 1+f(n/10,c+1);
  return 1-f(n/10,c-1);
}
```


a. 4 b. 3 c. 2 d. 1
- Utilizând metoda backtracking se generează toate paletete de culori posibile, formate din patru culori distincte din mulțimea {roșu,galben,albastru,verde,portocaliu,mov,roz}. În aceeași paletă nu poate exista mai mult decât o culoare primară. Mulțimea de culori primare este formată din {roșu,galben,albastru}. O paletă diferă prin cel puțin o culoare. Primele cinci soluții sunt, în această ordine: {roșu,verde,portocaliu,mov}, {roșu,verde,portocaliu,roz}, {roșu,portocaliu,mov,roz}, {galben,verde,portocaliu,mov} și {galben,verde,portocaliu,roz}. Indicați soluția generată înainte de {albastru,verde,portocaliu,mov}.
a. {albastru,verde,portocaliu,mov} b. {galben,portocaliu,portocaliu,mov}
c. {galben,albastru,mov,roz} d. {galben,portocaliu,mov,roz}
- Variabila **b** este declarată alăturat.
Indicați o expresie corectă din punct de vedere sintactic.

```
struct biblioteca
{ char nume[25],id;
  struct angajat
  { char nume[50]; int id;
  }a[10];
}b;
```


a. **b.nume[0]=id;** b. **b.a[0].nume[b.a[1].id];**
c. **b[0].nume[0]=b.a.nume[0];** d. **b.a[0].nume[0][b.a[1].id];**
- Într-un arbore cu rădăcină un nod se află pe nivelul **x** dacă lanțul elementar care are o extremitate în nodul respectiv și cealaltă extremitate în rădăcina arborelui are lungimea **x**. Pe nivelul 0 se află un singur nod (rădăcina).
Un arbore cu 12 noduri, numerotate de la 1 la 12, este reprezentat prin vectorul de "tați" (**x, 5, 7, 2, 0, 5, 5, 3, 2, y, 2, z**). Indicați un set de date corespunzător valorilor lui **x, y** și **z** astfel încât înălțimea maximă a arborelui să fie 4.
a. **x=3, y=3, z=1** b. **x=8, y=3, z=9** c. **x=3, y=4, z=9** d. **x=3, y=4, z=1**

SUBIECTUL al II-lea

(40 de puncte)

1. Algoritmul alăturat este reprezentat în pseudocod.

S-a notat cu $a \% b$ restul împărțirii numărului natural a la numărul natural nenul b .

- Scrieți valoarea afișată în urma executării algoritmului dacă se citesc pe rând valorile 167 5 37 24 2167 167 0. **(6p.)**
- Scrieți două seturi de date distincte astfel încât, în urma executării algoritmului valoarea afișată să fie egală cu valoarea inițială oferită lui x . **(6p.)**
- Scrieți programul C/C++ corespunzător algoritmului dat. **(10p.)**

```
p ← 1; k ← 0
citește x, n (numere naturale nenule)
pentru i ← 1, n execută
    citește nr (număr natural)
    dacă nr % p = x % p atunci
        k ← k + 1
    p ← p * 10
scrie k
```

- Scrieți în pseudocod un algoritm echivalent cu cel dat, înlocuind adecvat structura **pentru...execută** cu o structură repetitivă cu test final. **(6p.)**

2. Un oraș este format din 6 instituții principale, notate de la A la F. Între aceste instituții există un drum cu un singur sens de mers: de la instituția A la B, de la instituția A la E, de la instituția B la C și de la instituția C la E. Indicați numărul minim de arce ce trebuie adăugate, dar și mulțimea arcelor respective, astfel încât în graful orientat să existe cel puțin un drum de la o instituție către oricare alta. **(6p.)**

3. Variabila s poate memora un șir de maxim 30 de caractere, iar variabila k este de tip întreg. Utilizând eventuale alte variabile, scrieți o secvență de instrucțiuni, în urma căreia să se afișeze pe ecran variabila k ce reprezintă numărul de apariții ale caracterelor speciale din șir care NU se află pe prima sau ultima poziție. Se consideră caractere speciale cele din mulțimea {!, @, #, \$}. **(6p.)**

Exemplu: dacă în s se memorează șirul !nf#rm@t!c@ valoarea lui k afișată pe ecran va fi 3. **(6p.)**

SUBIECTUL al III-lea

(30 de puncte)

1. Subprogramul prime are doi parametri, x și y prin care primește două numere naturale din intervalul [5, 10⁶] cu $x < y$. Subprogramul afișează pe ecran toate perechile formate din câte 3 numere, între paranteze, separate prin câte un spațiu, unde numărul din mijloc este un număr prim din intervalul [x, y], iar vecinii lui reprezintă tot numere prime, mai mici, respectiv mai mari decât el, dar doar cea mai apropiată valoare posibilă. Scrieți definiția completă a subprogramului. **(10p.)**

Exemplu: dacă $x=12$ și $y=23$ se va afișa (11 13 17) (13 17 19) (17 19 23) (19 23 29). **(10p.)**

2. Scrieți un program C/C++ care citește de la tastatură un număr natural n din intervalul [2, 20], elementele unui tablou bidimensional cu n linii și n coloane, numere naturale din intervalul [0, 10⁶] și o cifră k. Programul permută circular la stânga toate elementele pare de pe diagonala principală cu k poziții și afișează tabloul obținut, fiecare linie a tabloului pe câte o linie a ecranului, elementele de pe aceeași linie fiind separate prin câte un spațiu. **(10p.)**

Exemplu: dacă $n=4$, matricea

2	1	9	7
0	3	3	5
2	8	4	7
6	5	4	0

și $k=2$, atunci se va afișa

0	1	9	7
0	3	3	5
2	8	2	7
6	5	4	4

(10p.)

3. Fișierul bac.txt conține pe prima linie două numere naturale n și m, unde n este un număr din intervalul [3, 10], iar m este un număr din intervalul [1, 10⁶], iar pe următoarele două linii numere naturale din intervalul [0, 10⁹]: pe prima linie un șir de n numere, iar pe a 2-a linie un șir de m numere. Se cere să se afișeze pe ecran de câte ori apare primul șir în cel de-al 2-lea sau mesajul nu exista în caz contrar. Proiectați un algoritm eficient din punct de vedere al timpului de executare. **(3 10**

Exemplu: dacă fișierul conține numere alăturate se afișează pe ecran 2.

3	10
2	6 0
2	6 2 6 0 3 2 6 0 9

- Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. **(2p.)**
- Scrieți programul C/C++ corespunzător algoritmului proiectat. **(8p.)**