

Examenul național de bacalaureat 2026
Proba E. d)
INFORMATICĂ
Limbajul C/C++

Model

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 arc/muchie are extremități distincte și oricare două arce/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.

1. Indicați valoarea expresiei C/C++ alăturate.

20/25*20/2

- a. 0 b. 0.02 c. 0.08 d. 8

2. Subprogramul **f** este definit alăturat.

```
int f(int s,int d,int v[])  
{ if(s==d) if(v[d]==2*d) return 1;  
  else return 0;  
  else return f(s,(s+d)/2,v) + f(1+(s+d)/2,d,v);  
}
```

Știind că variabila **x** memorează un tablou unidimensional cu elementele (2,0,2,6,8), în această ordine, numerotate de la 0 la 4, indicați valoarea **f(0,4,x)**.

- a. 1 b. 2 c. 4 d. 5

3. La un târg pentru copii, fiecare joc este asociat cu câte o singură abilitate – cheie, dintre cele pe care le poate dezvolta. Utilizând metoda backtracking, se generează toate posibilitățile de a expune seturi de câte trei jocuri din mulțimea {**jenga** (motricitate), **kendama** (motricitate), **lego** (creativitate), **șah** (strategie), **scrabble** (vocabulary)}, astfel încât să nu fie alese simultan două jocuri care dezvoltă aceeași abilitate - cheie, **scrabble** să NU apară pe prima poziție, iar **șahu1** să NU fie înainte de **jenga** sau **kendama**. Două seturi sunt distincte dacă diferă prin cel puțin un joc sau dacă ordinea jocurilor este diferită. Primele cinci seturi generate sunt, în această ordine (**jenga**, **lego**, **șah**), (**jenga**, **lego**, **scrabble**), (**jenga**, **șah**, **lego**), (**jenga**, **șah**, **scrabble**), (**jenga**, **scrabble**, **lego**). Indicați penultimul set generat.

- a. **șah**, **lego**, **scrabble** b. **șah**, **scrabble**, **lego**
c. **lego**, **jenga**, **șah** d. **lego**, **șah**, **jenga**

4. Variabila **z**, declarată alăturat, memorează unele date despre 100 dintre zonele continentului. Știind că densitatea unei zone este egală cu raportul dintre numărul de locuitori și suprafața acesteia (exprimată în km²), indicați o expresie a cărei valoare este egală cu numărul de locuitori ai primei zone.

```
struct zona  
{ char nume[21];  
  int densitate;  
  int suprafata;  
}z[100];
```

- a. **z[0].densitate*z[0].suprafata** b. **z.densitate[0]*z.suprafata[0]**
c. **densitate[0].z*suprafata[0].z** d. **densitate.z[0]*suprafata.z[0]**

5. Un graf neorientat are 7 noduri, numerotate de la 1 la 7, și 8 muchii, dintre care șase sunt: [1,2], [2,5], [2,6], [2,7], [3,7], [4,7]. Știind că unul dintre lanțurile elementare care au lungimea maximă este 3, 7, 4, 5, 2, 1, indicați care ar putea fi celelalte două muchii ale grafului.

- a. [1,6] și [4,5] b. [2,3] și [2,4] c. [2,3] și [4,5] d. [2,4] și [5,7]

SUBIECTUL al II-lea

(40 de puncte)

1. Algoritmul alăturat este reprezentat în pseudocod.

- a. Scrieți ce se afișează în urma executării algoritmului dacă se citește, în această ordine, valorile 27 și 38. (6p.)
- b. Dacă pentru variabila **m** se citește numărul 5, scrieți două valori care pot fi citite pentru variabila **n** astfel încât, în urma executării algoritmului, pentru fiecare dintre acestea, valoarea afișată să fie 10. (6p.)
- c. Scrieți programul C/C++ corespunzător algoritmului dat. (10p.)
- d. Scrieți în pseudocod un algoritm echivalent cu cel dat, înlocuind adecvat a doua structură **repetă...până când** cu o structură repetitivă cu test inițial. (6p.)

```

citește m,n
(numere naturale, 1<m<n)
repetă
    x←m; y←n; n←n-1
    repetă
        dacă x>y atunci x←x-y
        altfel y←y-x
    până când y=0
până când x≠1
scrie n+1
    
```

2. Un arbore cu 9 noduri, numerotate de la 1 la 9, are muchiile [1,2], [2,8], [2,9], [3,6], [4,6], [5,8], [6,8], [7,8]. Scrieți două noduri care pot fi alese drept rădăcină, astfel încât nodul 8 să aibă doi "frați". (6p.)

3. Variabilele **tI**, **pN** și **tL** permit accesul la câte un șir de maximum 20 de caractere. Inițial, șirul accesat prin **tL** este vid, șirul accesat prin **tI** reprezintă un număr de telefon în format internațional, iar șirul accesat prin **pN** reprezintă un prefix național sau este un șir vid.

Numărul de telefon în format internațional conține codul de țară, scris între paranteze rotunde, urmat de cifrele numărului propriu-zis. Codul de țară este format din cifre sau din simbolul + (plus), urmat de cifre. În multe țări, pentru a forma local un număr de telefon, se înlocuiește secvența formată din paranteze și codul de țară cu un prefix național format din cifre (de exemplu 0, în România) sau cu șirul vid (de exemplu în SUA).

Scrieți o secvență de instrucțiuni C/C++ astfel încât, în urma executării acesteia, șirul accesat prin **tL** să reprezinte numărul de telefon format local. Declarați corespunzător eventualele alte variabile utilizate.

Exemplu: dacă prin variabila **tI** se accesează șirul (+254) 722123456, iar prin variabila **pN** se accesează șirul 0, atunci numărul format local este 0722123456, iar dacă prin variabila **tI** se accesează șirul (+1)2121234567, iar prin variabila **pN** se accesează șirul vid, atunci numărul format local este 2121234567. (6p.)

SUBIECTUL al III-lea

(30 de puncte)

1. Subprogramul **Plus** are un singur parametru, **n**, prin care primește un număr natural ($n \in [10, 10^9]$). Subprogramul înlocuiește în **n** fiecare secvență 25 cu câte o secvență 26 și furnizează, prin același parametru, numărul obținut. Dacă nu se înlocuiește nicio secvență, subprogramul furnizează numărul nemodificat. Scrieți definiția completă a subprogramului C/C++.

Exemplu: dacă $n=202535250$, după apel $n=202635260$. (10p.)

2. Terenul alocat unui târg de jucării artisanale a fost împărțit în zone dispuse ca elementele unui tablou bidimensional cu număr egal de linii și coloane. Pentru a putea fi testate de copii, fiecare jucărie are alocată o zonă și este identificată printr-un număr natural. Organizatorii au hotărât eliminarea tuturor jucăriilor plasate pe diagonala secundară a tabloului și rearanjarea celor păstrate, prin deplasarea unora dintre acestea cu câte o poziție spre stânga, astfel încât să nu existe zone intermediare libere, ca în exemplu.

Scrieți un program C/C++ care citește de la tastatură numere naturale din intervalul $[2, 10^2]$, și anume **n** și elementele unui tablou bidimensional cu **n** linii și **n** coloane, reprezentând numerele de identificare ale jucăriilor, în ordinea plasării lor inițiale pe teren. Programul modifică apoi tabloul în memorie corespunzător hotărârii organizatorilor și afișează pe ecran tabloul obținut, fiecare linie a tabloului pe câte o linie a ecranului, elementele de pe aceeași linie fiind separate prin câte un spațiu.

Exemplu: pentru $n=4$ și tabloul

20	25	20	26
36	16	48	52
64	72	35	90
85	36	12	25

se obține tabloul

20	25	20
36	16	52
64	35	90
36	12	25

(10p.)

3. Se citește de la tastatură un număr natural, **n** ($n \in [1, 10^9]$), și se cere să se scrie în fișierul text **bac.txt** cel mai mare număr natural **p** cu proprietatea că numărul 26^p este divizor al numărului obținut prin calcularea produsului $1 \cdot 2 \cdot 3 \cdot \dots \cdot n$.

Proiectați un algoritm eficient din punctul de vedere al timpului de executare și al memoriei utilizate.

Exemplu: dacă $n=28$, atunci fișierul conține numărul 2 ($26^2=676$ este divizor al lui $1 \cdot 2 \cdot 3 \cdot \dots \cdot 28 = 304888344611713860501504000000$)

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