

Examenul național de bacalaureat 2026

Proba E.d)
INFORMATICĂ

Varianta 1

Filieră teoretică, profil real, specializare matematică-informatică / matematică-informatică intensiv informatică

Filieră vocațională, profil militar, specializare matematică-informatică

- Toate subiectele sunt obligatorii. Se acordă 10 puncte din oficiu.
- Timpul de lucru efectiv este de 3 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.

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. Ce va afișa următoarea secvență?

```
int a=5, b=2;  
int c=a/b; float d=(float)a/b;  
cout << c << " " << d;
```

a. 2 2

b. 2 2.5

c. 2.5 2.5

d. 5 2

2. Subprogramul `numar` este definit alăturat. Indicați valoarea `numar(3,0)`.

```
int numar(int n, int a)  
{ if(n==0) return a;  
  else  
  { a = numar(n-1,a+1);  
    return numar(n-1,a+2); } }
```

a. 13

b. 16

c. 18

d. 21

3. Considerăm 6 obiecte având greutatea 1, 3, 5, 7, 9, 15. Un robot are două brațe și fiecare obiect trebuie alocat unui braț. Robotul poate funcționa doar dacă diferența dintre totalurile celor două brațe nu depășește valoarea 4. Folosind un algoritm backtracking se generează primele 3 soluții:

1. Stânga: 1, 3, 5, 9 Dreapta: 7, 15

2. Stânga: 1, 3, 7, 9 Dreapta: 5, 15

3. Stânga: 1, 3, 15 Dreapta: 5, 7, 9

Care este a patra soluție generată?

a. Stânga: 1, 5, 15 Dreapta: 3, 7, 9

b. Stânga: 3, 7, 9 Dreapta: 1, 5, 15

c. Stânga: 1, 5, 7, 9 Dreapta: 3, 15

d. Stânga: 7, 15 Dreapta: 1, 3, 5, 9

4. Se definește: `struct Punct { int x, y; } p1, p2;`

Cum se calculează distanța euclidiană dintre punctele `p1` și `p2`?

a. $\sqrt{(p1.x-p2.x)^2 + (p1.y-p2.y)^2}$

b. $\sqrt{(p1.x-p2.x)*(p1.x-p2.x) + (p1.y-p2.y)*(p1.y-p2.y)}$

c. $\text{abs}(p1.x-p2.x) + \text{abs}(p1.y-p2.y)$

d. $p1.x-p2.x + p1.y-p2.y$

5. Fiind dat un arbore cu 8 noduri, dat prin vectorul de tați (0, 1, 1, 3, 3, 5, 5, 6) să se precizeze care este lungimea maximă a unui lanț elementar al acestuia.

a. 4

b. 1

c. 5

d. 2

SUBIECTUL al II-lea

(40 de puncte)

1. Algoritmul alăturat este reprezentat în pseudocod.

a. Scrieți valoarea care se va afișa dacă pentru `n` se citește valoarea **2356**? (6p.)

b. Care este cea mai mică valoare pară, de patru cifre, pentru variabila `n` astfel încât, în urma executării algoritmului, să se afișeze exact două valori. (6p.)

c. Scrieți programul C/C++ corespunzător algoritmului dat. (10p.)

d. Scrieți în pseudocod un algoritm echivalent cu cel dat, în care să înlocuiți structura repetitivă **pentru** **execută** cu o structură repetitivă de alt tip. (6p.)

```
citește n (număr natural nenul)  
p ← 1; nrc ← 0; cn ← n  
cât timp (cn ≠ 0) execută  
|   nrc ← nrc + 1  
|   p ← p * 10  
|   cn ← [cn/10]  
└─  
p ← [p/10]  
pentru i ← nrc, 1, -1 execută  
|   u ← n % 10  
|   n ← u * p + [n/10]  
|   dacă u ≠ 0 atunci  
|   |   scrie n, ' '  
|   └─  
└─
```

2. Fie un graf **G** cu **V** noduri. Graful complementar **C** al grafului **G** este un graf format din aceleași noduri **V**, însă cu muchii astfel încât doar nodurile neadiacente din **G** sunt adiacente în **C**. Considerăm că **G** este un graf neorientat cu 6 noduri, în care nodurile 2 și 4, 1 și 5 respectiv 3 și 6, sunt adiacente. Care este numărul de muchii al grafului complementar?
3. Scrieți o secvență de instrucțiuni care modifică elementele unui tablou bidimensional cu **m** linii și **n** coloane, numere întregi distincte de cel mult 4 cifre fiecare, eliminând din tablou, la nivelul memoriei, linia corespunzătoare elementului de valoare minimă.

Ex. Matricea inițială

2	7	1	4
14	6	12	3
9	22	8	5

Matricea modificată

14	6	12	3
9	22	8	5

SUBIECTUL al III-lea

(30 de puncte)

1. Subprogramul **simulare** are un singur parametru **n**, prin care primește un număr natural ($n \in [10, 10^9]$). Subprogramul înlocuiește cifrele prime ale numărului **n** cu cifra **1** și furnizează, prin același parametru, numărul obținut. Dacă nu se înlocuiește nicio cifră, subprogramul furnizează numărul nemodificat. Scrieți definiția completă a subprogramului.

Exemplu: dacă **n=124390**, după apel **n=114190**.

(10p.)

2. Scrieți un program C/C++ care citește de la tastatură, în această ordine, un număr natural **n**, apoi **n** cuvinte, separate prin Enter, urmate de un cuvânt **cuv**. Numărul **n** aparține intervalului $[1, 20]$, iar fiecare cuvânt este format din cel mult 20 de caractere, numai litere mici ale alfabetului englez. Programul construiește în memorie, apoi afișează pe ecran, un șir care conține toate cuvintele citite care **NU** conțin secvența formată din caracterele cuvântului **cuv**, separate prin câte un spațiu, sau mesajul **nu există** dacă toate cuvintele conțin cuvântul **cuv**. **Exemplu:** dacă se citesc, în această ordine, datele:

5

elev

bacalaureat

laureat

tabacar

bacterie

bac atunci se va afișa **elev laureat**

(10p.)

3. În inima Arhivei Cronos, timpul nu curgea pur și simplu; el era înregistrat, catalogat și păstrat în două mari Cărți Eterne: **Volumul A** și **Volumul B**. Volumul A conținea toate evenimentele majore din istorie, ordonate cu mîgală de la cel mai vechi la cel mai recent. Volumul B conținea toate evenimentele minore, dar esențiale (descoperiri uitate, întâlniri tănuite), sortate și ele perfect, de la zorii lumii până la clipa prezentă. Veghetorii Timpului, Maestrul K și ucenicul său, Elara, primeau o misiune critică: trebuiau să creeze Cartea Finală a Destinului, un singur **Volum C**, care să cuprindă toată istoria, mare și mică, dar care să mențină o ordine cronologică absolută. Ajută-l pe Elara să realizeze Cartea Finală a Destinului perfect ordonată și fără a risipi o singură secundă de energie. Fișierul text **bac1.txt** conține cel mult 10^6 numere naturale de maxim nouă cifre reprezentând evenimentele majore din Volumul A, iar fișierul text **bac2.txt** conține cel mult 10^6 numere naturale de maxim nouă cifre reprezentând evenimentele minore din Volumul B. Se cere să se afișeze pe ecran, separate printr-un spațiu, conținutul Volumului C. Proiectați un algoritm eficient din punctul de vedere al memoriei utilizate și al timpului de executare.

Exemplu: Dacă volumul **A (Majore)** conține evenimentele **(1935, 1945, 1980, 1996, 2008)**, iar volumul **B (Minore)** conține evenimentele **(1920, 1935, 1955, 1977)** atunci Marea Carte finală a destinului va conține evenimentele **(1920, 1935, 1935, 1945, 1955, 1977, 1980, 1996, 2008)**

a. Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia.

(2p.)

b. Scrieți programul C/C++ corespunzător algoritmului proiectat.

(8p.)