

Examenul Național de Bacalaureat 2026

Proba E. d)
INFORMATICĂ
Limbajul C/C++

Simulare județeană – Varianta 1

Filieră teoretică, profil real, specializare științe ale naturii

- 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.

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. Variabila **x** este de tip întreg. Indicați o expresie C/C++ care are valoarea **1** dacă și numai dacă numărul memorat în variabila **x** aparține intervalului **(-15, 20]**.

- a. **$x > -15 \parallel x \leq 20$** b. **$x \geq -15 \&\& x < 20$** c. **$x > -15 \&\& x \leq 20$** d. **$x < -15 \parallel x > 20$**

2. În secvența de mai jos, toate variabilele sunt de tip întreg. Indicați ce se va afișa în urma executării secvenței.

```
for (i = 0; i < 4; i++) {  
    for (j = 0; j < 4; j++)  
        if (i + j == 3) cout << 1;  
        else cout << 2;  
    cout << endl;  
}
```

a.	b.	c.	d.
2221	1222	2221	1111
2212	2122	2221	2222
2122	2212	2221	2222
1222	2221	2221	2222

3. Pentru a verifica dacă în tabloul unidimensional (2, 5, 9, 14, 20, 28, 35) există elementul cu valoarea **x=10**, se aplică metoda căutării binare. Indicați succesiunea de elemente ale tabloului a căror valoare se compară cu valoarea lui **x** pe parcursul aplicării metodei.

- a. 14, 5, 9 b. 14, 20, 9 c. 14, 9, 5 d. 2, 5, 9

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

 $\text{floor}(10.9) + \text{ceil}(10.1) - \text{sqrt}(16)$

- a. 16 b. 17 c. 18 d. 20

5. În secvența C/C++ alăturată, variabilele **n** și **p** sunt de tip întreg, iar **n** memorează un număr natural nenul.

Valoarea variabilei **p** în urma executării secvenței este egală cu:

- a. Produsul cifrelor numărului **n**.
b. Produsul cifrelor pare ale numărului **n**.
c. Produsul cifrelor impare ale numărului **n** sau **1** dacă nu există cifre impare.
d. Ultima cifră impară a numărului **n**.

```
p = 1;  
while (n > 0) {  
    if (n % 2 == 1)  
        p = p * (n % 10);  
    n = n / 10;  
}
```

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 și cu $[c]$ partea întreagă a numărului real c .

- a. Scrieți ce se afișează dacă se citește numărul 38275. **(6p.)**
b. Scrieți două numere distincte din intervalul $[100, 999]$ care pot fi citite astfel încât, în urma executării algoritmului, valoarea afișată pentru k să fie 0. **(6p.)**
c. Scrieți programul C/C++ corespunzător algoritmului dat. **(10p.)**
d. Scrieți în pseudocod un algoritm echivalent cu cel dat, înlocuind structura cât timp...execută cu o structură repetitivă cu test final. **(6p.)**

citește n (număr natural) $k \leftarrow 0$ $m \leftarrow 0$ cât timp $n > 0$ execută $c \leftarrow n \% 10$ dacă $c > 4$ atunci $k \leftarrow k * 10 + c$

altfel

 $m \leftarrow m + c$ $n \leftarrow [n / 10]$ scrie k, m

2. Tablourile unidimensionale A și B au valorile $A=(15, 12, 8, 4, 3)$ și $B=(20, 10, 5, 2)$ și sunt interclasate în ordine descrescătoare. Scrieți elementele noului tablou obținut, în ordinea din acesta. **(6p.)**

3. Un magazin online gestionează date despre laptopuri. Pentru fiecare laptop se memorează: un cod de identificare (număr natural) și prețul (număr real). Variabilele $cod1$, $pret1$ memorează datele primului laptop, iar $cod2$, $pret2$ datele celui de-al doilea.

Declarați variabilele necesare și scrieți o secvență de instrucțiuni C/C++ care afișează pe ecran codul de identificare al laptopului cu prețul mai mic. Dacă prețurile sunt egale, se va afișa mesajul **Preturi egale**. **(6p.)**

SUBIECTUL al III-lea**(30 de puncte)**

1. Se citește un număr natural n ($n \in [1, 10^9]$) și se cere să se scrie cel mai mare divizor propriu al său. Un divizor propriu este un divizor diferit de 1 și de numărul însuși. Dacă numărul nu are divizori proprii (este prim), se va afișa mesajul **PRIM**.

Scrieți, în pseudocod, algoritmul de rezolvare pentru problema enunțată.

Exemplu: Dacă $n = 20$, se scrie 10. Dacă $n = 13$, se scrie **PRIM**.

(10p.)

2. Scrieți un program C/C++ care citește de la tastatură un număr natural $n \in [2, 50]$ și apoi n numere naturale, elemente ale unui tablou unidimensional. Programul construiește în memorie și afișează pe ecran un nou tablou care conține doar elementele pătrate perfecte din șirul inițial, în ordinea apariției lor. Dacă nu există pătrate perfecte, se va afișa mesajul **NU EXISTA**.

Exemplu: Pentru $n=6$ și tabloul (10, 4, 35, 9, 81, 12), se afișează 4 9 81.

(10p.)

3. Fișierul **bac.txt** conține un șir de cel puțin două și cel mult 10^6 numere naturale din intervalul $[0, 10^9]$, separate prin câte un spațiu.

Se cere să se determine și să se afișeze pe ecran cele mai mari două valori pare distincte din șir. Dacă în șir nu există două numere pare distincte, se va afișa mesajul **Nu exista**.

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

Exemplu: Dacă fișierul conține 11 20 5 8 20 4 32 9, se afișează 32 20.

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

(2p.)

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

(8p.)