

Examenul național de bacalaureat 2025  
Proba E. d)  
INFORMATICĂ  
Limbajul C/C++

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.

- Indicați expresia C/C++ cu aceeași valoare ca a expresiei alăturate. 2025%2019+6  
a. 2025/2020+5      b. 2025/2021+8      c. 2025%2020+5      d. 2025%2021+8
- Pentru a verifica dacă în tabloul unidimensional (1, 4, 6, 8, 9, 11, 15, 18, 21, 25, 30) există elementul cu valoarea **x**, se aplică metoda căutării binare. Indicați trei dintre valorile pe care le poate avea **x**, astfel încât căutarea să se încheie după ce **x** a fost comparat cu cel mult două elemente ale tabloului.  
a. 1, 4, 30      b. 1, 25, 30      c. 4, 11, 25      d. 6, 11, 21
- Indicați valoarea expresiei C/C++ alăturate. floor(sqrt(25))-sqrt(20)  
a. -1      b. 0      c. 1      d. 5
- Variabilele **x** și **y** sunt de tip întreg și memorează numere naturale nenule, reprezentând prețurile a două calculatoare. Indicați o instrucțiune care poate înlocui punctele de suspensie astfel încât, în urma executării secvenței C/C++ obținute, să se interschimbe valorile inițiale ale variabilelor **x** și **y**.  
a. **x=y-x;**      b. **y=x+y;**      c. **y=x-y;**      d. **y=y-x;**  
x=x+y;  
.....  
x=x-y;
- În secvența alăturată, toate variabilele sunt întregi. Indicați expresia care poate înlocui punctele de suspensie astfel încât, în urma executării secvenței obținute, variabila **p** să memoreze valoarea produsului 3!·5! (unde s-a notat cu **x!**, pentru **x** ∈ {3, 5}, factorialul lui **x**, calculat ca produs al tuturor numerelor naturale nenule, distincte, mai mici sau egale cu **x**).  
p=1; f=1;  
for(i=3; i<=6; i++)  
if(i%2==1)  
{ f=.....;  
p=p\*f;  
}  
}  
a. **f\*(i-1)\*i**      b. **f\*i\*(i+1)**      c. **f\*i**      d. **(i-2)\*(i-1)\*i**

**SUBIECTUL al II-lea** (40 de puncte)

- Algoritmul alăturat este reprezentat în pseudocod.  
a. Scrieți ce se afișează în urma executării algoritmului dacă se citesc, în această ordine, numerele 7 și 17. (6p.)  
b. Dacă pentru variabila **n** se citește valoarea 25, scrieți două numere distincte care pot fi citite pentru variabila **m**, astfel încât, în urma executării algoritmului, pentru fiecare dintre acestea, să se afișeze valoarea 25. (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 m, n
(numerale naturale nenule, m ≤ n)
nr ← 0; i ← m
repetă
    x ← 1
    cât timp x*x < i execută
        x ← x+1
    dacă x*x = i atunci nr ← i
    altfel i ← i+1
până când i > n sau nr ≠ 0
scrie nr
```

2. Tablourile unidimensionale  $A = (20, 15, 10, 9, 7, 4)$  și  $B = (21, 12, 8, 6, 5)$  sunt interclasate în ordine crescătoare, fiind parcurse de la dreapta la stânga. Scrieți mulțimea tuturor valorilor din  $B$  care, pe parcursul aplicării metodei de interclasare, se compară cu valoarea 7 din  $A$ . (6p.)
3. Variabila  $p$  memorează prețul unei prăjituri (număr natural din intervalul  $[5, 50]$ ), iar variabila  $c$  memorează litera  $L$ , dacă prăjitura are cremă de ciocolată cu lapte, litera  $V$  dacă are cremă de vanilie și litera  $N$  dacă nu are cremă. Declarați variabila  $c$  și scrieți o secvență de instrucțiuni C/C++ în urma executării căreia să se afișeze, pe prima linie a ecranului, cuvântul **crema**, urmat de cuvântul **DA**, dacă prăjitura are cremă, sau de cuvântul **NU**, în caz contrar, iar pe a doua linie a ecranului cuvântul **pret**, urmat de prețul prăjiturii. Primul cuvânt de pe fiecare linie este urmat de un spațiu, ca în exemplu.  
**Exemplu:** dacă variabila  $c$  memorează litera  $V$ , iar variabila  $p$  memorează 10, se afișează pe ecran:  
**crema DA**  
**pret 10** (6p.)

**SUBIECTUL al III-lea**

**(30 de puncte)**

1. Numărul natural  $an$  este **ascendent** al numărului natural  $n$  dacă oricare dintre cifrele lui  $an$  este mai mare sau egală cu cifra unităților lui  $n$ .  
**Exemplu:** oricare dintre numerele 7, 9, 98 sau 7998 este ascendent al lui 827, dar numărul 857 nu este ascendent al lui 827.  
Se citesc trei numere naturale din intervalul  $[0, 10^3]$ ,  $n$ ,  $x$  și  $y$  ( $x < y$ ), și se cere să se scrie suma ascendenților lui  $n$  din intervalul  $[x, y]$ . Scrieți, în pseudocod, algoritmul de rezolvare a problemei enunțate.  
**Exemplu:** dacă  $n=827$ ,  $x=9$ ,  $y=800$ , se scrie 7893 ( $9+77+78+79+87+88+89+97+98+99+777+778+779+787+788+789+797+798+799=7893$ ). (10p.)
2. Un **șir semioglindit** se obține dintr-un șir de  $2 \cdot k$  ( $k \in [1, 10^2]$ ) termeni, prin interschimbarea în acesta a secvenței formate din primii  $k$  termeni cu secvența formată din ultimii  $k$  termeni.  
Scrieți un program C/C++ care citește de la tastatură un număr natural  $n$  ( $n \in [2, 10^2]$ ) și un șir de  $n$  numere naturale din intervalul  $[1, 10^6]$ , elemente ale unui tablou unidimensional, apoi modifică tabloul în memorie, dacă are un număr par de elemente, prin înlocuirea șirului memorat cu șirul semioglindit obținut din el. Programul afișează pe ecran elementele tabloului obținut, separate prin câte un spațiu, sau mesajul **nu exista**, dacă șirul are număr impar de termeni.  
**Exemplu:** pentru  $n=6$  și tabloul (2, 3, 4, 4, 5, 1) se obține tabloul (4, 5, 1, 2, 3, 4). (10p.)
3. Un tânăr pasionat de călătorii are o listă cu muzee virtuale și, pentru fiecare, câte un singur interval orar, în care acesta poate fi vizitat online, gratuit. Tânărul dispune zilnic de același interval orar pentru vizite; un muzeu este **convenabil** dacă poate fi vizitat online gratuit în timpul disponibil și dacă pentru vizită îi poate aloca cel puțin o oră.  
Fișierul text **bac.in** conține cel mult  $10^5$  linii, iar pe fiecare linie câte o pereche de numere, reprezentând limitele câte unui interval orar: pe prima linie intervalul orar de care tânărul dispune zilnic, iar pe fiecare dintre următoarele linii, intervalul orar de vizitare gratuită pentru câte un muzeu, în ordinea din listă. Limitele intervalelor sunt ore fixe, numere naturale din intervalul  $[8, 22]$ , iar cele aflate pe aceeași linie a fișierului sunt în ordine strict crescătoare și sunt separate printr-un spațiu.  
Se cere să se afișeze pe ecran numărul de muzee convenabile. Utilizați un algoritm eficient din punctul de vedere al timpului de executare și al memoriei utilizate.  
**Exemplu:** dacă fișierul conține valorile alăturate, atunci pe ecran se afișează numărul 3.  
(pot fi vizitate trei muzee, respectiv primul, al doilea și al patrulea, în intervalele 16–18, 17–19, respectiv 18–19).
- |    |    |
|----|----|
| 16 | 19 |
| 15 | 18 |
| 17 | 21 |
| 19 | 21 |
| 18 | 20 |
| 12 | 13 |
- a. Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. (2p.)  
b. Scrieți programul C/C++ corespunzător algoritmului proiectat. (8p.)