

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

Model

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 intervalul căruia îi aparține valoarea variabilei reale x , dacă și numai dacă expresia C/C++ de mai jos are valoarea 1.
`!(x<2004) && !(x<2005 || x>2024) && !(x>2025)`
a. [2004, 2005] b. [2004, 2024] c. [2005, 2024] d. [2005, 2025]
- Pentru a verifica dacă în tabloul unidimensional (2, 5, 10, 15, 16, 25, 50) există elementul cu valoarea x , se aplică metoda căutării binare. Știind că valoarea x a fost comparată cu două elemente ale tabloului pe parcursul aplicării metodei, indicați două valori posibile ale lui x .
a. 2, 5 b. 5, 25 c. 10, 16 d. 25, 50
- Variabilele x și y sunt de tip real și x are valoarea 2.5, iar y are valoarea 5.2. Indicați expresia C/C++ cu valoarea -3.
a. `floor(-x-y)` b. `floor(-x+y)` c. `floor(x-y)` d. `floor(x+y)`
- În secvența alăturată, toate variabilele sunt de tip întreg, iar de la tastatură se citesc numere naturale, ultimul fiind un număr nul.

```
s=0;  
do  
{ cin>>x; | scanf("%d",&x);  
  .....  
}while(x!=0);
```

Indicați o instrucțiune care poate înlocui punctele de suspensie astfel încât, în urma executării secvenței obținute, valoarea variabilei s să fie suma numerelor citite mai mari sau egale cu 2025.

a. `if(x>=2025) s=s+x;` b. `if(x<2025) s=s-x;`
c. `if(x>=2025) s=s+x; else s=0;` d. `if(x<2025) s=s-x; else s=s+x;`
- În secvența alăturată, toate variabilele sunt întregi. Indicați ce reprezintă numărul obținut ca valoare a variabilei x în urma executării secvenței.

```
x=2025; y=280;  
while (x!=y)  
{ if(x>y) x=x-y;  
  else y=y-x;
```

a. câtul împărțirii lui 2025 la 280 b. restul împărțirii lui 2025 la 280
c. cel mai mic multiplu comun lui 2025 și 280 d. cel mai mare divizor comun lui 2025 și 280

SUBIECTUL al II-lea (40 de puncte)

- Algoritmul alăturat este reprezentat în pseudocod.
S-a notat cu $a \% b$ restul împărțirii numărului natural a la numărul natural nenul b .
a. Scrieți valoarea afișată în urma executării algoritmului, dacă se citesc, în această ordine, numerele 15, 3, 4. (6p.)
b. Scrieți două seturi distincte de date de intrare, astfel încât, în urma executării algoritmului, pentru fiecare dintre acestea să se afișeze valoarea 0. (6p.)
c. Scrieți programul C/C++ corespunzător algoritmului dat. (10p.)
d. Scrieți în pseudocod un algoritm echivalent cu cel dat, înlocuind adecvat structura **pentru...execută** cu o structură repetitivă cu test inițial. (6p.)

```
citește n,x,y  
(numere naturale nenule,  $x \leq n, y \leq n$ )  
nr ← 0  
pentru i ← n, 1, -1 execută  
{ dacă i % x = 0 și i % y ≠ 0 atunci  
  nr ← nr + 1  
  altfel  
  dacă i % x ≠ 0 și i % y = 0 atunci  
    nr ← nr + 1  
  scrie nr
```

2. Tabloul unidimensional **A** are elementele **A**=(3,21,25,37,55), iar în urma interclasării lui crescătoare cu tabloul unidimensional **B**, se obține tabloul cu elementele (3,4,7,8,21,25,37,45,55,70). Indicați elementele tabloului **B**, în ordinea apariției lor în acesta. (6p.)
3. O mașină de epocă a fost prezentată la două expoziții auto, în primele două luni ale anului curent. Variabilele **ziua1** și **luna1** memorează ziua, respectiv luna în care a avut loc una dintre cele două expoziții, iar variabilele **ziua2** și **luna2** memorează ziua, respectiv luna în care a avut loc cealaltă expoziție. Zilele sunt memorate ca numere naturale din intervalul [1,31], iar pentru fiecare lună se memorează prima literă, mare, din denumirea acesteia. Declarați variabilele **luna1** și **luna2** și scrieți o secvență de instrucțiuni C/C++ în urma executării căreia să se afișeze pe ecran data la care a avut loc prima expoziție, cu ziua și numele complet al lunii, scris cu litere mari ale alfabetului englez.
Exemplu: dacă **ziua1** și **luna1** memorează numărul 15, respectiv litera **I**, iar **ziua2** și **luna2** memorează numărul 20, respectiv litera **I** sau numărul 10, respectiv litera **F**, se afișează pe ecran
- 15 IANUARIE (6p.)

SUBIECTUL al III-lea

(30 de puncte)

1. Două numere se numesc **oglundite** dacă fiecare se obține din celălalt, prin parcurgerea cifrelor acestuia de la dreapta la stânga. Două numere se numesc **par-oglundite** dacă numerele obținute din acestea, prin îndepărtarea tuturor cifrelor lor impare sau nule, sunt oglindite.
Se citește de la tastatură două numere naturale din intervalul $[0, 10^9]$, **x** și **y**, și se cere să se scrie valoarea 1, dacă **x** și **y** sunt par-oglundite, sau valoarea 0, în caz contrar.
Scrieți, în pseudocod, algoritmul de rezolvare a problemei enunțate.
Exemplu: dacă **x**=814 și **y**=7003485, sau **x**=14 și **y**=700345, se scrie 1,
iar dacă **x**=814 și **y**=7003465, sau **x**=814 și **y**=7003845, sau **x**=15 și **y**=510, se scrie 0. (10p.)
2. Scrieți un program C/C++ care citește de la tastatură două numere naturale din intervalul $[2, 10^2]$, **k** și **n**, și construiește în memorie un tablou unidimensional cu **n** elemente, numerotate de la 0 la **n**-1, astfel încât parcurgând fiecare poziție pară a sa, de la stânga la dreapta, să se obțină un șir strict crescător format din primii multipli naturali nenuli ai lui **k**, și parcurgând fiecare poziție impară a sa, de la stânga la dreapta, să se obțină șirul strict descrescător al primelor numere naturale, ca în exemplu. Elementele tabloului obținut sunt afișate pe ecran, separate prin câte un spațiu.
Exemplu: pentru **k**=5 și **n**=9 se obține tabloul (5, 3, 10, 2, 15, 1, 20, 0, 25). (10p.)
3. La o expoziție auto se află, în șir, mașini de epocă, fiecare având câte un cod, format prin alipirea, în această ordine, a două numere naturale nenule: identificatorul colecționarului care deține mașina, respectiv anul fabricației acesteia.
Fișierul **bac.txt** conține numere naturale: pe prima linie un număr **x** ($x \in [1880, 1950]$), reprezentând un an calendaristic, iar pe a doua linie cel mult 10^5 numere din intervalul $[10^4, 10^9]$, reprezentând codurile mașinilor, în ordinea din șirul în care sunt expuse. Numerele aflate pe aceeași linie în fișier sunt separate prin câte un spațiu.
Se cere să se afișeze pe ecran identificatorul colecționarului care deține ultima mașină, din șirul celor expuse, fabricată în anul **x**. Dacă nu există o astfel de mașină, se afișează pe ecran mesajul **nu exista**. Proiectați un algoritm eficient din punctul de vedere al memoriei utilizate și al timpului de executare.
Exemplu: dacă fișierul conține numerele
- 1925
31885 21925 8931925 121900 201925 31925 151925 61950 201925 121880
- atunci pe ecran se afișează 20
- a. Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. (2p.)
b. Scrieți programul C/C++ corespunzător algoritmului proiectat. (8p.)