

Examenul național de bacalaureat 2021
Proba E. d)
INFORMATICĂ
Limbajul C/C++

Testul 11

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 muchie are extremități distincte și oricare două 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.

- Variabila x este de tip întreg și memorează un număr nenul. Indicați expresia C/C++ cu valoarea 1 dacă 2021 este divizor al lui x .
a. $x / (x / 2021) == 0$ b. $x / (x \% 2021) == 0$ c. $x \% (x / 2021) == 0$ d. $x \% (x \% 2021) == 0$
- Indicați o expresie C/C++ care are valoarea 1.
a. $\text{ceil}(5.19) == \text{floor}(5.91)$ b. $\text{ceil}(5.19) == \text{ceil}(5.91)$
c. $\text{ceil}(5.19) - 1 == \text{floor}(5.19)$ d. $\text{ceil}(5) + 1 == \text{floor}(5)$
- Pentru a verifica dacă în tabloul unidimensional (91,83,22,19,12,8,3) există elementul cu valoarea x se aplică metoda căutării binare. Știind că valoarea x a fost comparată cu un singur element al tabloului pe parcursul aplicării metodei, x ar putea fi:
a. 19 b. 91 c. 8 sau 3 d. 83 sau 8
- Tablourile unidimensionale X și Y au elementele: $X = (4, 6, 9, 20, 21)$, iar $Y = (5, 12, 13, 24, 52)$. În urma interclasării lor în ordine crescătoare se obține tabloul cu elementele:
a. (6, 15, 19, 41, 70) b. (4, 5, 6, 12, 9, 13, 20, 24, 21, 52)
c. (4, 6, 9, 20, 21, 5, 12, 13, 24, 52) d. (4, 5, 6, 9, 12, 13, 20, 21, 24, 52)
- Variabila e memorează un număr natural din intervalul $[10^4, 10^5)$. Indicați o instrucțiune în urma executării căreia se elimină cifra din mijloc a acestui număr.
a. $e = e / 100 * 100 + e \% 100$; b. $e = e / 1000 + e \% 100$;
c. $e = e / 1000 * 100 + e \% 100$; d. $e = e \% 1000 * 100 + e / 100$;

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 și cu $a \leftrightarrow b$ operația de interschimbare a valorilor variabilelor a și b .
a. Scrieți succesiunea de caractere afișate în urma executării algoritmului dacă se citește, în această ordine, numerele 2 și 9. (6p.)
b. Dacă pentru n se citește numărul 79, scrieți două numere nenule care pot fi citite pentru m astfel încât, în urma executării algoritmului, pentru fiecare dintre acestea, să se afișeze de 40 de ori caracterul *.(6p.)
c. Scrieți programul C/C++ corespunzător algoritmului dat, fără a utiliza eventuale funcții predefinite pentru operația de interschimbare. (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  
(numere naturale)  
dacă m>n atunci  
  n←m  
dacă m%2=0 atunci  
  m←m+1  
cât timp m≤n execută  
  m←m+2  
  scrie '*'
```

2. Variabilele **serie** și **an** memorează date pentru o mașină: seria (o literă mare a alfabetului englez) și anul fabricației acesteia (un număr natural din intervalul $[2010, 2021]$). Declarați variabila **serie** și scrieți o secvență de instrucțiuni C/C++ în urma executării căreia să se afișeze pe ecran mesajul **Masina noua** dacă anul fabricației acesteia este peste 2015, mesajul **Mediu**, dacă anul fabricației acesteia este 2015, sau mesajul **Masina veche** în caz contrar. (6p.)

3. Variabila **c** este de tip **char**, iar variabilele **i** și **k** sunt de tip întreg; de la tastatură se citesc 7 cifre sau litere ale alfabetului englez.

Scrieți secvența alăturată înlocuind punctele de suspensie astfel încât, în urma executării secvenței obținute, valoarea variabilei **k** să fie egală cu numărul de cifre citite.

Exemplu: dacă se citesc literele și cifrele

b, a, c, 2, 0, 2, 1

variabila **k** are valoarea 4.

```
k=.....;
for (i=1;i<=7;i++)
{ cin>>c;
  .....
}
```

(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 **impar-oglundite** dacă numerele obținute din acestea, prin îndepărtarea tuturor cifrelor lor pare, sunt oglindite.

Se citesc două numere naturale, **x** și **y**, și se cere să se scrie valoarea 1 dacă **x** și **y** sunt impar-oglundite sau valoarea 0 în caz contrar.

Scrieți, în pseudocod, algoritmul de rezolvare a problemei enunțate.

Exemplu: dacă $x=523$ și $y=84356$, se scrie 1,

iar dacă $x=523$ și $y=84536$ sau $x=523$ și $y=84576$ sau $x=40$ și $y=86$, se scrie 0.

(10p.)

2. Scrieți un program C/C++ care citește de la tastatură numere naturale: **n** ($n \in [2, 20]$), **k** ($k \in [1, n]$) și $2 \cdot n$ numere din intervalul $[0, 10^9]$, elemente ale unui tablou unidimensional. Programul transformă tabloul în memorie, interschimbând șirul primelor **k** elemente cu șirul ultimelor **k** elemente, ambele șiruri fiind parcurse de la stânga la dreapta, ca în exemplu. Elementele tabloului obținut sunt afișate pe ecran, separate prin câte un spațiu.

Exemplu: pentru $n=5$, $k=3$ și tabloul $(6, 1, 3, 6, 9, 4, 8, 5, 8, 0)$

se obține tabloul $(5, 8, 0, 6, 9, 4, 8, 6, 1, 3)$.

(10p.)

3. Se consideră șirul 1, 3, 5, 7 . . . definit astfel: $f_0=1$, iar $f_n=f_{n-1}+2$ dacă $n \geq 1$ (unde **n** este un număr natural).

Se citește de la tastatură un număr natural, **x** ($x \in [1, 10^9]$), reprezentând un termen în șirul dat, și se cere să se scrie în fișierul text **bac.out**, separați prin câte un spațiu, toți termenii șirului mai mici sau egali cu **x**, în ordine inversă a apariției lor în șir. Proiectați un algoritm eficient din punctul de vedere al spațiului de memorie și al timpului de executare.

Exemplu: dacă $x=7$, fișierul conține valorile

7 5 3 1

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

(2p.)

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

(8p.)