

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

Varianta 7

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.  $x\% (x\%2021) == 0$
- Utilizând metoda backtracking se generează toate valorile naturale formate cu cifre ale numărului 2021 și care sunt strict mai mici decât acesta. Primele 12 soluții generate sunt, în această ordine, 0, 1, 10, 100, 1000, 1001, 1002, 101, 1010, 1011, 1012, 102. Indicați două soluții care se generează pe poziții consecutive, în această ordine.  
a. 2002, 2020      b. 2020, 21      c. 122, 1221      d. 21, 22
- Variabilele  $x$  și  $y$  sunt declarate alăturat. Indicați o instrucțiune de atribuire corectă din punctul de vedere sintactic.  

```
struct masina  
{ char model[51];  
  int an;  
}x[10],y;
```

  
a.  $x[1]=y;$       b.  $x.an[1]=y.an;$   
c.  $x[1].an.masina=y.an.masina;$       d.  $model.x[1]=model.y;$
- Un arbore cu 11 noduri, numerotate de la 1 la 11, este reprezentat prin vectorul de „tați” (9, 3, 4, 6, 4, 8, 6, 0, 4, 4, 4). Indicați numărul de descendenți direcți („fii”) ai nodului 4.  
a. 2      b. 3      c. 5      d. 6
- Indicați numărul de noduri ale unui graf neorientat, complet, cu 210 muchii.  
a. 14      b. 15      c. 20      d. 21

**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. Subprogramul  $f$  este definit alăturat. Scrieți ce valori au  $f(2)$ , respectiv  $f(21)$ . (6p.)
- ```
int f(int x)
{ if (x<=4) return x;
  else return x-f(x-2);
}
```
3. Variabila  $i$  este de tip întreg, iar variabila  $x$  permite memorarea unui șir cu cel mult 100 de caractere. Scrieți ce se afișează în urma executării secvenței alăturate. (6p.)
- ```
strcpy(x, "bac2021");
cout<<x+3<<endl; | printf("%s\n", x+3);
for (i=0; i<strlen(x); i++)
  if (strchr("0123456789", x[i])==0)
    cout<<x[i]<<'!'; | printf("%c! ", x[i]);
```

**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. Subprogramul `imog` are trei parametri:
- $x$  și  $y$ , prin care primește câte un număr natural din intervalul  $[0, 10^9]$ ;
  - $rez$ , prin care furnizează valoarea 1 dacă  $x$  și  $y$  sunt impar-oglundite sau valoarea 0 în caz contrar.
- Scrieți definiția completă a subprogramului.  
**Exemplu:** dacă  $x=523$  și  $y=84356$ , după apel  $rez=1$ , iar dacă  $x=523$  și  $y=84536$  sau  $x=523$  și  $y=84576$  sau  $x=40$  și  $y=86$ , după apel  $rez=0$ . (10p.)
2. Scrieți un program C/C++ care citește de la tastatură numere naturale:  $n$  ( $n \in [2, 20]$ ),  $k$  ( $k \in [2, n]$ ) și  $n \cdot n$  numere din intervalul  $[0, 10^9]$ , elemente ale unui tablou bidimensional cu  $n$  linii și  $n$  coloane. Programul transformă tabloul în memorie, interschimbând șirul elementelor situate pe linia a  $k$ -a, în stânga diagonalei principale, parcurse de la stânga la dreapta, cu șirul elementelor situate pe coloana a  $k$ -a, deasupra diagonalei principale, parcurse de sus în jos, ca în exemplu. Elementele tabloului obținut sunt afișate pe ecran, fiecare linie pe câte o linie a ecranului, cu elementele fiecărei linii separate prin câte un spațiu.
- Exemplu:** pentru  $n=5$ ,  $k=4$  și tabloul
- |   |   |   |   |   |
|---|---|---|---|---|
| 2 | 4 | 3 | 5 | 6 |
| 8 | 0 | 9 | 8 | 7 |
| 2 | 6 | 9 | 0 | 5 |
| 6 | 1 | 3 | 6 | 9 |
| 7 | 3 | 9 | 4 | 2 |
- se obține tabloul
- |   |   |   |   |   |
|---|---|---|---|---|
| 2 | 4 | 3 | 6 | 6 |
| 8 | 0 | 9 | 1 | 7 |
| 2 | 6 | 9 | 3 | 5 |
| 5 | 8 | 0 | 6 | 9 |
| 7 | 3 | 9 | 4 | 2 |
- (10p.)
3. Se consideră șirul 1, 3, 7, 13, 21, 31, 43 ... definit astfel:  $f_0=1$ , iar  $f_n=f_{n-1}+2 \cdot n$ , dacă  $n \geq 1$  (unde  $n$  este un număr natural). Se citesc de la tastatură două numere naturale din intervalul  $[1, 10^9]$ ,  $x$  și  $y$  ( $x < y$ ), reprezentând doi termeni aflați pe poziții consecutive 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  $y$ , 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  $x=21$  și  $y=31$ , fișierul conține valorile  
31 21 13 7 3 1
- a. Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. (2p.)
- b. Scrieți programul C/C++ corespunzător algoritmului proiectat. (8p.)