

**Examenul de bacalaureat național 2014**  
**Proba E. d)**  
**Informatică**  
**Limbajul C/C++**

Varianta 10

*Filiera teoretică, profilul real, specializările: matematică-informatică  
matematică-informatică intensiv informatică*  
*Filiera vocațională, profilul militar, specializarea matematică-informatică*

- Toate subiectele sunt obligatorii. Se acordă 10 puncte din oficiu.
- Timpul de lucru efectiv este de 3 ore.
- În rezolvările cerute, identificatorii utilizați 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).
- În programele cerute, datele de intrare se consideră corecte, validarea acestora nefiind necesară.

**SUBIECTUL I** **(30 de puncte)**

Pentru itemul 1, scrieți pe foaia de examen litera corespunzătoare răspunsului corect.

1. Variabila  $x$  este de tip întreg și poate memora un număr natural cu cel mult două cifre. Valoarea maximă pe care o poate avea expresia C/C++ alăturată este: **(4p.)**  $x \% 7$
- a. 6                      b. 14.14                      c. 93                      d. 693

2. Se consideră algoritmul alăturat, reprezentat în pseudocod.

- a) Scrieți valoarea afișată dacă se citesc, în această ordine, numerele 10, 8, 11, 1, 21, 0. **(6p.)**
- b) Scrieți un set de patru numere distincte din intervalul  $[0, 9]$  care pot fi citite astfel încât, în urma executării algoritmului, să se afișeze valoarea 0. **(4p.)**
- c) Scrieți în pseudocod un algoritm, echivalent cu cel dat, în care să se înlocuiască prima structură **repetă...până când** cu o structură repetitivă cu test inițial. **(6p.)**

```
n ← 0
repetă
    citește x
    (număr natural)
    a ← 0
    b ← 1
    repetă
        c ← a + b
        a ← b
        b ← c
    până când c ≥ x
    dacă x = c atunci
        n ← n + 1
    până când x = 0
scrie n
```

- d) Scrieți programul C/C++ corespunzător algoritmului dat. **(10p.)**

**SUBIECTUL al II-lea**

**(30 de puncte)**

**Pentru fiecare dintre itemii 1 și 2 scrieți pe foaia de examen litera corespunzătoare răspunsului corect.**

1. Un arbore cu 4 noduri, numerotate de la 1 la 4, **NU** poate fi reprezentat prin vectorul de „tați”:  
**(4p.)**  
a. (0,1,2,3)      b. (2,1,0,3)      c. (4,4,4,0)      d. (2,3,4,0)
2. Se consideră un graf neorientat complet, cu 9 noduri. Pentru a obține un graf parțial al său cu două componente conexe, fiecare dintre acestea fiind grafuri complete, numărul maxim de muchii care pot fi eliminate este:  
**(4p.)**  
a. 14      b. 18      c. 20      d. 24

**Scrieți pe foaia de examen răspunsul pentru fiecare dintre cerințele următoare.**

3. Se consideră graful orientat cu 6 vârfuri, numerotate de la 1 la 6, reprezentat prin matricea de adiacență alăturată. Scrieți un drum elementar din graf, cu extremitatea inițială în vârful 4 și extremitatea finală în vârful 6.  
**(6p.)**

0	0	0	0	1	0
1	0	0	1	0	1
0	0	0	0	1	0
1	0	0	0	1	0
0	1	0	0	0	0
0	0	0	0	1	0
4. Variabila **s** poate memora un șir cu maximum 20 de caractere, iar variabila **i** este de tip întreg. Scrieți ce se afișează în urma executării secvenței de instrucțiuni de mai jos.  

```
strcpy(s, "BACALAUREAT");  
cout<<strlen(s); | printf("%d",strlen(s));  
i=0;  
while (i<strlen(s)-1)  
{ if(strchr("EAIOU",s[i])!=NULL) strcpy(s+i+1,s+i+2);  
  i++;  
}  
cout<<s; | printf("%s",s);
```

**(6p.)**
5. Scrieți un program C/C++ care citește de la tastatură două numere naturale, **m** și **n** ( $2 \leq m \leq 20$ ,  $2 \leq n \leq 20$ ), și construiește în memorie un tablou bidimensional **A**, cu **m** linii și **n** coloane, astfel încât parcurgându-l linie cu linie, de sus în jos, și fiecare linie de la stânga la dreapta, să se obțină șirul primelor  $n \cdot m$  numere naturale, pare, care **NU** sunt divizibile cu 5, ordonat strict crescător. Programul afișează pe ecran tabloul obținut, fiecare linie a tabloului pe câte o linie a ecranului, elementele de pe aceeași linie fiind separate prin câte un spațiu.

**Exemplu:** pentru  $m=4$  și  $n=3$  se obține tabloul alăturat.

**(10p.)**

	2	4	6
	8	12	14
	16	18	22
	24	26	28

**SUBIECTUL al III-lea**

**(30 de puncte)**

**Pentru itemul 1, scrieți pe foaia de examen litera corespunzătoare răspunsului corect.**

1. Utilizând metoda backtracking, se generează toate posibilitățile de a forma succesiuni de câte 5 genuri muzicale distincte din mulțimea {jazz, rock, latino, house, pop}, astfel încât în fiecare succesiune genul latino precede genul house. Două succesiuni sunt distincte dacă genurile muzicale sunt în altă ordine.  
Primele cinci soluții generate sunt, în această ordine, (jazz, rock, latino, house, pop), (jazz, rock, latino, pop, house), (jazz, rock, pop, latino, house), (jazz, latino, rock, house, pop), (jazz, latino, rock, pop, house). Imediat înainte de (pop, latino, house, jazz, rock) este generată soluția: **(4p.)**
- a. (rock, jazz, house, latino, pop)      b. (rock, jazz, latino, house, pop)  
c. (pop, latino, rock, house, jazz)      d. (pop, rock, latino, house, jazz)

**Scrieți pe foaia de examen răspunsul pentru fiecare dintre cerințele următoare.**

2. Se consideră subprogramul `f`, definit alăturat. Scrieți două valori naturale distincte din intervalul  $[1, 50]$  pe care le poate avea variabila întreagă `x`, astfel încât `f(30, x)` să aibă valoarea 5. **(6p.)**
- ```
int f(int a, int b)
{ if (b==0) return a;
  else return f(b, a%b);
}
```
3. Se consideră subprogramul `triplete`, cu un singur parametru, `n`, prin care primește o valoare naturală din intervalul  $[2, 10^4]$ . Subprogramul afișează pe ecran toate tripletele de numere naturale  $(x, y, z)$  cu proprietatea că  $x < y < z$  și  $x \cdot y + y \cdot z = n$ . Fiecare triplet se afișează pe câte o linie a ecranului, iar numerele din fiecare triplet sunt separate prin câte o virgulă și încadrate între paranteze rotunde, ca în exemplu.  
Scrieți definiția completă a subprogramului.  
**Exemplu:** pentru  $n=8$  se afișează pe ecran, nu neapărat în această ordine, tripletele:  
(0, 1, 8)  
(0, 2, 4)  
(1, 2, 3) **(10p.)**
4. Fișierul `bac.txt` conține pe prima linie un număr natural  $n$  ( $1 \leq n \leq 10^6$ ), iar pe a doua linie cel mult 1000000 de numere naturale de forma  $10^p$  ( $0 \leq p \leq 9$ ), separate prin câte un spațiu. Se cere să se afișeze pe ecran numărul care ar apărea pe poziția  $n$  în șirul ordonat crescător obținut din toate numerele aflate pe a doua linie a fișierului. Dacă șirul are mai puțin de  $n$  termeni, se afișează pe ecran mesajul **Nu exista**.  
Pentru determinarea numărului cerut se utilizează un algoritm eficient din punctul de vedere al timpului de executare.  
**Exemplu:** dacă fișierul `bac.txt` conține numerele  
5  
100 100000 1 100000 1000 100 10  
atunci pe ecran se afișează valoarea  
1000  
a) Descrieți în limbaj natural algoritmul utilizat, justificând eficiența acestuia. **(4p.)**  
b) Scrieți programul C/C++ corespunzător algoritmului descris. **(6p.)**