

**EXAMENUL DE BACALAUREAT – 2007**  
**Proba scrisă la INFORMATICĂ**  
**PROBA E, limbajul C/C++**

Varianta 25

- ♦ Toate subiectele sunt obligatorii. Se acordă 10 puncte din oficiu.
- ♦ Timpul efectiv de lucru este de 3 ore.
- ♦ În programele cerute la subiectele II și III, 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).

**SUBIECTUL I (40 de puncte)**

Pentru fiecare din itemii de la 1 la 8, scrieți pe foaia de examen litera corespunzătoare răspunsului corect. Fiecare răspuns corect se notează cu 5 puncte.

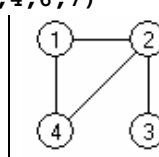
1. Fie subprogramul **P** definit mai jos. Ce se afișează în urma apelului **P(3)**?  

```
void P(int x){
    if(x)if(x%2==0){
        cout<<x; | printf("%d",x);
        P(x/2);}
    else{
        P(x-1);
        cout<<x; | printf("%d",x);}
}
```

a. 321                      b. 231                      c. 213                      d. 123
2. Se consideră mulțimea {1,7,5,16,12}; se generează prin metoda backtracking toate submulțimile sale formate din exact 3 elemente: primele patru soluții generate sunt, în ordine: {1,7,5}, {1,7,16}, {1,7,12}, {1,5,16}. Care dintre soluții trebuie eliminată din șirul următor astfel încât cele rămase să apară în șir în ordinea generării lor?  
{1,16,12}, {5,16,12}, {7,5,16}, {7,5,12}  

a. {1,16,12}                      b. {7,5,16}                      c. {7,5,12}                      d. {5,16,12}
3. Numărul de noduri ale unui arbore cu 100 de muchii este:  

a. 101                      b. 99                      c. 100                      d. 50
4. Se consideră două tablouri unidimensionale **A** și **B**: **A**=(1,3,5,9,10) respectiv **B**=(2,4,6,7). În urma interclasării lor în ordine crescătoare se obține tabloul cu elementele:  

a. (1,2,3,4,5,6,9,7,10)                      b. (1,2,3,4,5,6,7,9,10)  
c. nu se poate realiza interclasarea                      d. (1,3,5,9,10,2,4,6,7)
5. Se consideră graful neorientat din figura alăturată. Câte grafuri parțiale distincte, diferite de el însuși, fără vârfuri izolate, se pot obține? Două grafuri sunt distincte dacă matricele lor de adiacență sunt diferite.  


a. 3                      b. 13                      c. 5                      d. 4
6. Ce valoare are variabila întreagă **n** în urma executării secvenței alăturate?  

```
n=0 ;
while(n<=11) n=n+2;
```

a. 11                      b. 9                      c. 10                      d. 12
7. Știind că variabila reală **x** are o valoare ce aparține intervalului închis [3,7], care dintre expresiile de mai jos, scrise în limbajul C/C++, **NU** are valoarea 1?  

a. !(x<3 || x>7)                      b. !(x<3) && !(x>7)  
c. x>=3 && x<=7                      d. x<3 && x<=7
8. Variabilele **a** și **b** sunt declarate astfel: **char a[20],b[20]**; Ce se afișează în urma executării secvenței de mai jos?  

```
strcpy(a,"bac20"); strcpy(b,"07");
cout<<strcat(a,"")<<b; | printf("%s%s",strcat(a,""),b);
```

a. bac20\*07                      b. bac2007                      c. bac20bac2007                      d. bac140

### SUBIECTUL II (20 de puncte)

Se consideră programul pseudocod alăturat:

S-a notat cu  $a_k$  elementul aflat pe poziția  $k$  în tabloul unidimensional cu numele  $a$ , cu  $|x|$  valoarea absolută a numărului natural  $x$  și cu  $[x]$  partea întreagă a numărului  $x$ .

1. Ce se afișează dacă sunt citite, în ordine, numerele 3, 5, 8, 1, 7? (5 p.)
2. Scrieți o succesiune de valori citite pentru elementele  $a_1, a_2, a_3, a_4, a_5$  astfel încât să se afișeze prima valoare introdusă. (3 p.)
3. Explicați în limbaj natural **efectul** atribuirilor  
 $m \leftarrow a_i + a_{i+1} + |a_i - a_{i+1}|$   
 $a_{i+1} \leftarrow a_i + a_{i+1} - [m/2]$   
 $a_i \leftarrow [m/2]$   
 din algoritmul dat. (2 p.)
4. Scrieți programul C/C++ corespunzător algoritmului dat alăturat. (10 p.)

```

pentru i ← 1, 5 execută
| citește  $a_i$  (număr întreg)
■
pentru i ← 1, 4 execută
|  $m \leftarrow a_i + a_{i+1} + |a_i - a_{i+1}|$ 
|  $a_{i+1} \leftarrow a_i + a_{i+1} - [m/2]$ 
|  $a_i \leftarrow [m/2]$ 
■
scrie  $a_5$ 

```

### SUBIECTUL III (30 de puncte)

1. Scrieți un program C/C++ care citește de pe prima linie a fișierului **BAC.TXT** un număr natural  $n$  ( $1 \leq n \leq 100$ ) și de pe următoarele  $n$  linii, câte două numere reale reprezentând coordonatele carteziene (abscisă, ordonată) a  $n$  puncte din planul  $xy$ . Programul afișează pe ecran coordonatele punctelor care se află la cea mai mică distanță de originea  $O$  a planului  $xy$ ; coordonatele fiecărui astfel de punct vor fi separate prin virgulă și grupate între paranteze rotunde. (10 p.)

**Exemplu:** dacă în fișier se află numerele

```

3
2.0 1.0
2.0 2.0
1.0 2.0

```

Se afișează: (2.0,1.0) (1.0,2.0)

2. Se consideră șirul  $f: 1, 5, 2, 13, 10, \dots$  generat după regula de mai jos, în care s-a notat cu  $[x]$  partea întreagă a numărului  $x$ :

$$f_n = \begin{cases} 1 & \text{pentru } n = 1 \\ 2 * f_{[n/2]} + 3 & \text{pentru } n > 1, n \text{ par} \\ 2 * f_{[n/2]} & \text{pentru } n > 1, n \text{ impar} \end{cases}$$

a) Scrieți un program C/C++ care citește un număr natural  $x$  (cu cel mult 9 cifre) apoi afișează pe ecran mesajul **DA** dacă  $x$  este printre primii un miliard de termenii ai șirului dat și mesajul **NU** în caz contrar. Se va utiliza un algoritm eficient din punct de vedere al memoriei utilizate și al timpului de executare.

**Exemplu:** pentru  $x=10$  se va afișa **DA** (8 p.)

b) Descrieți metoda utilizată și explicați în ce constă eficiența ei. (2 p.)

3. Se consideră subprogramul **cifra**, cu 3 parametri, care
  - primește prin intermediul parametrului **nr** un număr natural cu cel mult 9 cifre
  - returnează prin intermediul parametrilor **max** și **ap** cifra cea mai mare care apare în numărul **nr**, respectiv numărul de poziții pe care se află aceasta în numărul **nr**.

**Exemplu:** dacă **nr=5255** funcția returnează **max=5** și **ap=3**

a) Scrieți definiția completă a subprogramului **cifra**. (6 p.)

b) Scrieți un program C/C++ care citește de la tastatură un număr natural  $a$  cu cel mult 9 cifre și afișează pe ecran mesajul **DA** dacă numărul dat are cel puțin două cifre 9 în scrierea sa și afișează mesajul **NU** în caz contrar. În rezolvare se va utiliza cel puțin un apel al subprogramului **cifra**.

**Exemplu:** dacă  $a=31390$ , se afișează **NU**, iar dacă  $a=79099$  se afișează **DA**. (4 p.)