

**EXAMENUL DE BACALAUREAT – 2009**  
**Proba scrisă la INFORMATICĂ**  
**PROBA E, limbajul C/C++**  
**Specializarea Matematică-informatică**

- ♦ Toate subiectele sunt obligatorii. Se acordă 10 puncte din oficiu.
- ♦ Timpul efectiv de lucru 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).

**Subiectul I (30 de puncte)**

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

1. Câte atribuiri se execută, în total, în  
secvența alăturată, dacă **n** și **p** sunt  
variabile de tip întreg? **(4p.)**
- ```
p=1; n=279;  
while (n>=100)  
{ p=p*10;  
  n=n-100;  
}
```

a. 4                                      b. 6                                      c. 2                                      d. 8

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

- 2. Se consideră algoritmul alăturat, descris în pseudocod.**

S-a notat cu  **$x \% y$**  restul împărțirii numărului natural **x** la numărul natural nenul **y** și cu **[z]** partea întreagă a numărului real **z**.

- a) Scrieți numerele care se vor afișa în urma executării algoritmului dacă se citesc valorile **a=312** și **b=1354**. **(6p.)**
- b) Scrieți câte o valoare care poate fi citită pentru variabila **a**, respectiv **b**, astfel încât algoritmul să afișeze exact 2 valori. **(4p.)**
- c) Scrieți programul C/C++ corespunzător algoritmului dat. **(10p.)**
- d) Scrieți în pseudocod un algoritm echivalent cu cel dat, în care să se înlocuiască structura **pentru...execută** cu o structură repetitivă de alt tip. **(6p.)**

```
citește a,b  
  (numere naturale)  
a ← [a/10]*10+a%10  
b ← [b/10]*10+b%10  
pentru i ← a,b execută  
  dacă [i/10]=i%10 atunci  
    scrie i%10  
  ■  
  ■
```

**Subiectul II (30 de puncte)**

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

1. Care este numărul de muchii care trebuie eliminate dintr-un graf neorientat, complet, cu 7 noduri, astfel încât graful parțial obținut să fie arbore? **(4p.)**  
a. 15                      b. 1                      c. 6                      d. 21
2. Care dintre următoarele variante reprezintă o declarație corectă pentru o variabilă **x** care memorează simultan partea reală și partea imaginară a unui număr complex? **(4p.)**  
a. `struct x {float im,re;};`                      b. `char x[2];`  
c. `struct complex{ float im, re;} x;`                      d. `float x;`

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

3. Ce va afișa secvența alăturată de program, știind că variabila **x** memorează un șir cu cel mult 100 de caractere, iar variabila **i** este de tip întreg? **(6p.)**  

```
strcpy(x,"bac2008");  
for(i=3;i<strlen(x);i++)  
    cout<<x[i]; | printf("%c",x[i]);  
cout<<x<<endl; | printf("%s\n",x);
```
4. Care vor fi valorile primului și ultimului element extras dintr-o **coadă** inițial vidă, dacă se efectuează următoarele operații, în această ordine: se introduce valoarea 5; se introduce valoarea 4; se extrage un element; se introduce valoarea 2; se introduce valoarea 7; se extrage un element. **(6p.)**
5. Scrieți un program C/C++ care citește de la tastatură un număr natural **n** ( $2 < n < 20$ ), construiește în memorie și afișează pe ecran o matrice cu **n** linii și **n** coloane, numerotate de la 1 la **n**, în care fiecare element aflat pe o coloană impară este egal cu suma dintre numărul liniei și numărul coloanei pe care se află și fiecare element aflat pe o coloană pară este egal cu numărul liniei pe care se află.  
Elementele matricei vor fi afișate pe ecran, câte o linie a matricei pe câte o linie a ecranului cu câte un spațiu între elementele fiecărei linii.  
**Exemplu:** pentru **n=5** se va afișa matricea alăturată. **(10p.)**  

|   |   |   |   |    |
|---|---|---|---|----|
| 2 | 1 | 4 | 1 | 6  |
| 3 | 2 | 5 | 2 | 7  |
| 4 | 3 | 6 | 3 | 8  |
| 5 | 4 | 7 | 4 | 9  |
| 6 | 5 | 8 | 5 | 10 |

**Subiectul III (30 de puncte)**

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

1. Funcția **F** are definiția alăturată. Ce valoare are **F(18)**? (4p.)

```
int F(int x){  
    if (x<=1) return x;  
    else return x+F(x-2);  
}
```

a. 90

b. 171

c. 91

d. 18

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

2. Un algoritm generează în ordine crescătoare toate numerele de **n** cifre ( $n < 9$ ), cu cifre distincte, care nu au două cifre pare alăturate. Dacă pentru **n=5**, primele cinci soluții generate sunt 10325, 10327, 10329, 10345, 10347, precizați care sunt următoarele **trei** soluții generate, în ordinea obținerii lor. (6p.)

3. Subprogramul **aranjare** are doi parametri, **a** și **n**, prin care primește un tablou unidimensional cu maximum 100 de numere reale nenule și, respectiv, numărul de elemente din tablou. Subprogramul rearanjează elementele tabloului astfel încât toate valorile negative să se afle pe primele poziții, iar valorile pozitive în continuarea celor negative. Ordinea în cadrul secvenței de elemente pozitive, respectiv în cadrul secvenței de elemente negative, poate fi oricare.

**Exemplu:** dacă este transmis ca parametru un tablou unidimensional cu 6 elemente de forma (12, -7.5, 6.5, -3, -8, 7.5), după apel, acesta ar putea fi: (-7.5, -3, -8, 12, 6.5, 7.5).

**a)** Scrieți definiția completă a subprogramului **aranjare**. (10p.)

**b)** Scrieți un program C/C++ care citește de la tastatură un număr natural **n** ( $1 \leq n \leq 100$ ) și apoi un șir de **n** numere reale nenule și care, folosind apeluri utile ale subprogramului **aranjare**, afișează pe ecran, separate prin spațiu, mai întâi elementele pozitive din șir și apoi cele negative. Ordinea în cadrul secvenței de elemente pozitive, respectiv în cadrul secvenței de elemente negative, poate fi oricare.

**Exemplu:** pentru **n=5** și pentru șirul 6, -16.3, 8, -18, 20.7 se poate afișa pe ecran soluția  
6 8 20.7 -18 -16.3 (4p.)

4. În fișierul **numere.txt** sunt memorate mai multe numere reale separate prin câte un spațiu. Scrieți un program C/C++ care verifică dacă printre numerele din fișier există cel puțin 10 numere naturale. Programul afișează pe ecran mesajul **DA** în caz afirmativ și **NU** în caz contrar.

**Exemplu:** dacă fișierul **numere.txt** conține numerele 60 -12.67 15 -1 -22.3 4 se afișează mesajul **NU**. (6p.)