

## Probă scrisă la INFORMATICĂ

**Filiera teoretică, profilul real, specializarile: matematică-informatică  
matematică-informatică intensiv informatică**

- Toate subiectele sunt obligatorii. Se acordă 10 puncte din oficiu.
- Timpul efectiv de lucru este de 3 ore.

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

1. Câte atribuiri se execută în secvența următoare, pentru  $n=245$ ? **(4p)**

```
s=0;
while (n!=0)
{
s=s+1;
n=n/100;
}
```

**a. 5****b. 7****c. 3****d. 1**

2. Fie secvența alăturată:

a) Ce va afișa algoritmul pentru  $a=832$  și  $b=97464$ ? **(6p.)**

b) Scrieți algoritmul pseudocod echivalent cu cel dat care folosește un alt tip de structură repetitivă . **(6p.)**

c) Scrieți un algoritm echivalent cu cel dat, în limbaj pseudocod sau limbaj de programare, care să nu utilizeze nici o structură repetitivă . **(4p.)**

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

```
citeste a,b {a,b ∈ N}
s←0
x←a%10*10+a%10
y←b%10*10+b%10
pentru i=x,y execută
┌dacă [i/10]=i%10
atunci
┌s←s+1
└─┘
scrie s
```

**SUBIECTUL al II-lea****(30 de puncte)**

1. Care este numărul maxim de componente conexe ale unui graf neorientat cu 40 de noduri și 25 de muchii? **(6p.)**  
 a. 25                      b. 31                      c. 33                      d. 35

2. Se consideră declarațiile alăturate, în care fiecare din variabilele s1 și s2 memorează coordonatele unui segment, în sistemul de coordonate **xOy**.

```
struct punct {
    float x,y;
};
struct segment {
    punct A,B;
}s1,s2;
```

Scrieți o expresie C/C++ care are valoarea 1 dacă segmentele s1 și s2 sunt ambele paralele cu axa **Ox** sau sunt ambele paralele cu axa **Oy** și care are valoarea 0, în caz contrar. **(6p.)**

3. În secvența următoare de program, variabilele **i** și **j** sunt de tip **intreg**, iar variabila **A** memorează o matrice cu **6** linii și **6** coloane (numerotate de la **1** la **6**) cu elemente de tip **int**. Care este suma elementelor de pe diagonala secundară a matricei **A** la finalul executării secvenței date? **(4p.)**

```
for (i=1; i<=6; i++)
for (j=1; j<=6; j++)
if ((i+j)%2==0)
    A[i][j]=(i+j)/2;
else
    A[i][j]=i+j;
```

4. Un graf hamiltonian are 6 noduri, toate de grad 3. Câte muchii trebuie eliminate din graf, astfel încât să devină arbore? **(4p.)**

a. 6

b. 4

c. 13

d. 7

5. Se consideră un text format din maximum 250 de caractere, litere mici ale alfabetului englez și spații. Scrieți un program C/C++ care să determine și să afișeze litera sau literele care apar o singură dată în textul dat. Dacă textul nu conține astfel de litere, se va afișa mesajul **nu există**. **(10p.)**

Exemple: Dacă textul este: **maine va fi soare** se vor afișa literele **v m f s n o r**, nu neaparat în această ordine. Dacă textul este: **carte cartea** se va afișa mesajul **nu există**.

**SUBIECTUL al III-lea****(30 de puncte)**

1. Utilizând metoda backtracking se generează lexicografic anagramele cuvântului **carte**. Ce anagrama trebuie eliminată din secvența de mai jos astfel încât cele rămase să reprezinte o succesiune corectă: **(6p)**

1. aectr      2. trcea      3. aeprt      4. aerct      5. aertc      6. aetcr

a. 1 și 2      b. 2 și 3      c. 5 și 6      d. 3 și 4

2. Subprogramul **f** este definit astfel:

```
void f(int n)
{
    int i;
    if(n>0)
    {
        for(i=1; i<=n; i++)
        {
            f(n-2);
            cout<<i; | printf("%d",i);
        }
    }
}
```

Scrieți ce se afișează în urma apelului **f(3)**. **(6p)**

3. Subprogramul **maxim** are doi parametri:

- **a**, prin care primește un număr natural ( $2 \leq a \leq 20000$ );
- **b**, prin care furnizează cel mai mare număr natural obținut prin permutarea circulară a cifrelor numărului **a** cu o poziție spre dreapta. **Exemplu:** dacă **a=3074** atunci **b** va avea valoarea **7430**.

Scrieți în limbajul **C/C++** definiția completă a subprogramului. **(10p)**

4. Numim **secvență fazan** a unui șir de numere naturale un subșir al acestuia, format din termeni aflați pe poziții consecutive în șirul dat cu proprietatea ca prima cifră a termenului curent este egală cu ultima cifră a termenului anterior. Lungimea secvenței este egală cu numărul de termeni ai acesteia.

Fișierul **bac.txt** conține un șir de cel puțin două și cel mult 1000000000 de numere naturale din intervalul  $[0, 10^9]$ . Numerele sunt separate prin câte un spațiu, iar în șir există cel puțin o secvență fazan. Se cere să se determine o **secvență fazan** de lungime maximă în șirul aflat în fișier și să se afișeze pe ecran lungimea acestei secvențe. Pentru determinarea secvenței cerute se utilizează un algoritm eficient din punctul de vedere al memoriei necesare și al timpului de executare. **Exemplu:** dacă fișierul **bac.txt** conține numerele **12 13 31 123 321 61 76 25 54 425 511 121 311 311** atunci pe ecran se afișează valoarea **5**.

a) Descrieți în limbaj natural algoritmul utilizat, justificând eficiența acestuia. **(2p.)**

b) Scrieți programul C/C++ corespunzător algoritmului descris. **(8p.)**