

Probă scrisă la INFORMATICĂ

Filiera teoretică, profilul real, specializarea științe ale naturii

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

SUBIECTUL I**(30 de puncte)**

- a) 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)**

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

1. Se consideră șirul lui Fibonacci, în care primii 2 termeni sunt $f_1=1$, $f_2=1$ iar al k-lea termen ($k > 2$) se calculează prin formula: $f_k=f_{k-1} + f_{k-2}$.

În secvența de mai jos, toate variabilele sunt de tip int. Scrieți instrucțiunile care pot înlocui zona punctată astfel încât, în urma executării secvenței, să se afișeze pe ecran primii 20 termeni ai șirului Fibonacci, separați prin câte un spațiu. **(6p.)**

```
f1=1; f2=1;
cout<<"1 1 "; | printf("1 1 ");
for(i=3;i<=20; i++)
{
f3=f1+f2;
...
}
```

2. Instrucțiunea C/C++ a cărei execuție determină memorarea în variabila reală m a mediei aritmetice a valorilor variabilelor reale a, b, c este: **(4p.)**

- a. $m=(x+y+z)/2.0$
- b. $m=x+y+z / 3$
- c. $m=x/1/3+y/1/3+z/1/3$
- d. $m=x/3+(y+z)/2$

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

3. Variabilele întregi x și y , respectiv z și u memorează abscisa și ordonata colțului stânga sus și dreapta jos ale unui dreptunghi. Fie a și b abscisa și ordonata unui punct P din planul de coordonate xOy. Scrieți instrucțiunile C/C++ care să afișeze pe ecran mesajul **DA** dacă punctul P se află în interiorul dreptunghiului menționat anterior sau **NU** în caz contrar. **(6p.)**

4. Se citesc pe rând n numere naturale de maxim 6 cifre. Să se afișeze numărul cifrelor de 0 de la sfârșitul produsului numerelor citite.

Exemplu: dacă se citesc $n=8$ și apoi valorile 30, 42, 60, 66, 7, 75, 84, 90 se va afișa **4**.

- a) Scrieți în pseudocod, algoritmul de rezolvare pentru problema enunțată. **(10p.)**
- b) Precizați rolul tuturor variabilelor care au intervenit în algoritmul realizat la punctul a) și indicați datele de intrare, datele de ieșire și datele de manevră (intermediare) ale problemei enunțate. **(4p.)**

1. Fie vectorul $v = (-2, 3, 4, 5, 7, 3)$ cu $n=6$ elemente indexate de la 0. Care din următoarele secvențe de instrucțiuni transformă vectorul v astfel: $v = (3, 7, 5, 4, 3, -2)$ **(4p.)**

- a) `for (k=0; k<n; k++) {
 aux=a[k]; a[k]=a[n-1-k];
 a[n-1-k]=aux; }`
- b) `for (k=0; k<=n/2; k++) {
 aux=a[k]; a[k]=a[n+1-k];
 a[n+1-k]=aux; }`
- c) `for (k=0; k<n/2; k++) {
 aux=a[k]; a[k]=a[n-1-k];
 a[n-1-k]=aux; }`
- d) `for (k=n-1; k>=0; k--) {
 aux=a[k]; a[k]=a[n-1-k];
 a[n-1-k]=aux; }`

2. În secvența alăturată variabilele i , j și aux sunt întregi, iar v memorează un tablou unidimensional ale cărui elemente au următoarele valori: $v[1]=2$, $v[2]=13$, $v[3]=1$, $v[4]=5$, $v[5]=-4$. Care este numărul de interschimbări care se efectuează în urma executării secvenței alăturate? Dați un exemplu de 5 valori distincte 2 câte 2 pentru elementele tabloului astfel încât, în urma executării secvenței de mai jos, să nu se efectueze nici o interschimbare. **(6p.)**

```
for (i=1; i<5; i++)
  for (j=5; j>i; j--)
    if (v[i]<v[j])
    {
      aux=v[i];
      v[i]=v[j];
      v[j]=aux;
    }
```

3. Scrieți un program C/C++ care citește de la tastatură un număr natural n ($2 < n < 50$) și cele n elemente ale unui tablou unidimensional, numere naturale cu cel mult nouă cifre. Programul modifică tabloul în memorie, eliminând toate elementele pătrate perfect ale acestuia, ca în exemplu, apoi afișează pe ecran, separate prin câte un spațiu, elementele tabloului obținut câte 5 pe linie (ultima linie poate avea mai puțin de 5 valori afișate). Dacă asupra vectorului nu se face nici o modificare se va afișa mesajul NESCHIMBAT

Exemplu: pentru $n=6$ și tabloul $\{36, 19, 125, 81, 1002, 73\}$,
programul afișează pe ecran: 19 125 1002 73 **(10p.)**

4. Fișierul text **SIR.IN** conține un număr natural n de cel mult 6 cifre.

Se cere afișarea în fișierul **SIR.OUT** a termenului de rang n al șirului 1, 2, 1, 3, 2, 1, 4, 3, 2, 1, 5, 4, 3, 2, 1,

Se va utiliza un algoritm eficient din punctul de vedere al memoriei utilizate și al timpului de executare.

Exemplu: dacă fișierul **SIR.IN** conține valoarea 9 atunci în fișierul de ieșire se va afișa valoarea 2.

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

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