

Specializarea Matematică-informatică intensiv 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. Variabilele **x**, **y** și **z** memorează valori reale, iar **x<y**. Care dintre expresiile de mai jos, scrise în limbajul C/C++, are valoarea 1 dacă valoarea variabilei **z** este situată în intervalul închis determinat de valorile memorate în variabilele **x** și **y**? **(4p.)**
- a. (**z>=x**) && (**z<=y**) b. (**z<=x**) && (**z<=y**)
c. (**y<x**) || (**z<=x**) d. (**z<x**) || (**z>y**)

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

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

S-a notat cu $\mathbf{x} \% \mathbf{y}$ restul împărțirii numărului natural \mathbf{x} la numărul natural nenul \mathbf{y} .

- Scrieți valoarea afișată în urma executării algoritmului dacă se citesc, în ordine, valorile: 5, 14, 6, 15, 16, 90. (6p.)
- Știind că valoarea citită pentru x este 7, scrieți un șir de numere distincte care pot fi citite în continuare astfel încât valoarea afișată să fie 0. (4p.)
- Scrieți programul C/C++ corespunzător algoritmului dat. (10p.)
- Scrieți în pseudocod un algoritm echivalent cu cel dat în care să se înlocuiască structura **pentru...execută** cu o structură repetitivă **cât timp...execută**. (6p.)

```

citește x
      (număr natural nenul)
nr ← 0
┌ pentru i ← 1, x execută
│   citește n (număr întreg)
│   ┌ dacă n % x = 0 atunci
│   │   nr ← nr + 1
│   └─┐
│     └─┐
└─┐
    └─┐
scrie nr

```


Subiectul III (30 de puncte)

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

1. Se utilizează metoda backtracking pentru a genera toate submulțimile cu 4 elemente ale mulțimii $\{1, 2, 3, 4, 5, 6\}$. Numărul de submulțimi generate este: **(4p.)**
- a. 30 b. 35 c. 5 d. 15

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

2. Se consideră subprogramul recursiv definit alăturat. Ce valori vor fi afișate pe ecran în urma apelului `gama(6);`? **(6p.)**
- ```
void gama(int n)
{
 int i;
 if(n >= 3)
 {
 for(i = 3; i <= n; i++)
 printf("%d ", n); | cout << n << " ";
 gama(n - 3);
 }
}
```
3. a) Să se scrie definiția completă a subprogramului `calcul`, care primește prin intermediul celor doi parametri ai săi două numere întregi,  $n$  și  $k$  ( $1 \leq n \leq 1000000000$  și  $1 \leq k \leq 9$ ), și returnează cifra de rang  $k$  a numărului  $n$ . Rangul unei cifre este numărul său de ordine, numerotând cifrele de la dreapta la stânga; cifra unităților având rangul 1. Dacă numărul  $k$  este mai mare decât numărul de cifre ale lui  $n$ , atunci funcția returnează valoarea -1.  
**Exemplu:** dacă  $n=9243$  și  $k=3$ , în urma apelului se va returna 2. **(5p.)**
- b) Scrieți un program C/C++ care citește de la tastatură un număr natural  $n$  cu cel mult 8 cifre. Programul va verifica, utilizând apeluri ale subprogramului `calcul`, dacă orice cifră a lui  $n$  are rangul cifrei mai mare sau egal cu valoarea cifrei respective și va afișa mesajul **Da** în caz afirmativ și mesajul **Nu** în caz contrar.  
**Exemplu :** pentru  $n=4160$  se va afișa **Nu**. **(5p.)**
- |       |   |   |   |   |
|-------|---|---|---|---|
| rang  | 4 | 3 | 2 | 1 |
| cifră | 4 | 1 | 6 | 0 |
4. Fișierul text `SIR.TXT` conține pe prima linie un număr natural  $n$  ( $1 \leq n \leq 10000$ ) și pe a doua linie, separate prin spații, un șir **crescător** de  $n$  numere naturale cu cel mult 9 cifre fiecare.  
Numim platou într-un șir de valori o secvență de elemente identice situate pe poziții alăturate. Lungimea unui platou este egală cu numărul de elemente care îl formează.
- a) Scrieți un program C/C++ care citește valorile din fișier și, printr-o metodă eficientă din punct de vedere al timpului de executare și al spațiului de memorie utilizat afișează pe ecran, separate printr-un spațiu, lungimea maximă a unui platou, precum și valoarea care formează platoul. În cazul în care sunt mai multe platouri de aceeași lungime se va afișa valoarea cea mai mare care formează unul dintre aceste platouri. **(6p.)**
- Exemplu:** dacă fișierul `SIR.TXT` are conținutul alăturat,
- |                                           |
|-------------------------------------------|
| 10                                        |
| 11 211 211 211 328 400 400 1201 1201 1201 |
- atunci programul va afișa pe ecran **3 1201**.
- b) Descrieți succint, în limbaj natural, metoda utilizată la punctul a, justificând eficiența acesteia. **(4p.)**