

SUBIECTUL al II-lea

(30 de puncte)

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

1. O expresie C/C++ care are valoarea 1 este: **(4p.)**
- a. `floor(7)+1==ceil(7)` b. `floor(7.19)==ceil(7.19)`
c. `floor(7.19)==floor(7.91)` d. `floor(7.91)==ceil(7.19)`
2. Se consideră variabilele x , y , z și w care memorează câte un număr real, astfel încât expresia C/C++ alăturată are valoarea 1. **(4p.)**
- Variabila care are ca valoare cel mai mare dintre numerele menționate mai sus este:
- a. x b. y c. z d. w

`x<y && z<w && w<x`

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

3. Variabilele i și s sunt întregi. Rescrieți secvența de instrucțiuni alăturată, completând zona punctată astfel încât, în urma executării secvenței obținute, variabila s să memoreze suma divizorilor lui 2500 din intervalul $[25, 79]$. **(6p.)**
- ```
s=0;
for(i=79;i>=25;i--)
.....
```
4. Se citește un număr natural nenul și se cere să se scrie numărul de cifre distincte din scrierea acestuia.
- Exemplu:** dacă numărul citit este 1612325, valoarea obținută este 5.
- a) Scrieți, în pseudocod, algoritmul de rezolvare pentru problema enunțată. **(10p.)**
- b) Menționați rolul tuturor variabilelor care au intervenit în prelucrarea realizată la punctul a) și indicați datele de intrare, respectiv datele de ieșire ale problemei enunțate. **(6p.)**

**SUBIECTUL al III-lea**

**(30 de puncte)**

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

1. Pentru a calcula în mod eficient media aritmetică a elementelor unui tablou unidimensional cu  $n$  componente numere naturale, toate egale, este necesar și suficient să se execute: **(4p.)**
  - a. o singură instrucțiune de atribuire
  - b. o singură parcurgere a tabloului și o singură atribuire
  - c. o singură parcurgere a tabloului și două atribuiri
  - d. două parcurgeri ale tabloului

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

2. În secvența alăturată de instrucțiuni, variabilele  $i$  și  $j$  sunt de tip întreg.  

```
for (i=0; i<=4; i++) {
 for (.....)
 cout<<j<<' '; | printf("%d ", j);
 cout<<endl; | printf("\n");
}
```

Rescrieți secvența, completând zona punctată astfel încât, în urma executării secvenței obținute, să se afișeze pe ecran, în această ordine, numerele de mai jos.

0  
1 0  
2 1 0  
3 2 1 0  
4 3 2 1 0

**(6p.)**

3. Se consideră șirul  $s$ , construit după regula alăturată.

$$s_n = \begin{cases} 1 & \text{dacă } n \leq 2 \\ 3 \cdot s_{n-1} - s_{n-2} & \text{dacă } n > 3 \end{cases}$$

Scrieți un program C/C++ care citește de la tastatură un număr natural  $n$  ( $3 < n < 20$ ) și construiește în memorie un tablou unidimensional care memorează primii  $n$  termeni ai șirului  $s$ , definit după regula de mai sus, astfel încât numerele impare să ocupe primele poziții în tablou, iar cele pare să fie memorate în continuarea celor impare. Programul afișează pe ecran elementele tabloului construit, separate prin câte un spațiu.

**Exemplu:** dacă  $n=6$ , primii șase termeni ai șirului sunt 1, 1, 2, 5, 13, 34, iar tabloul construit poate fi (1, 1, 5, 13, 2, 34). **(10p.)**

4. Fișierul `bac.in` conține pe prima linie un număr natural  $n$  ( $3 < n < 1000$ ), iar pe următoarea linie, un șir de  $n$  numere naturale distincte, de cel mult nouă cifre fiecare. Numerele din șir sunt separate prin câte un spațiu și cel puțin două dintre ele au ultima cifră egală cu 5.

**a)** Scrieți un program C/C++ care citește toate numerele din fișier și, utilizând un algoritm eficient din punct de vedere al timpului de executare și al memoriei utilizate, determină și afișează pe ecran cele mai mari două numere din șir care au ultima cifră egală cu 5. Numerele determinate sunt afișate în ordine crescătoare, separate printr-un spațiu. **(6p.)**

**Exemplu:** dacă fișierul `bac.in` are conținutul  
10  
97 5 11 1 8 6 85 3 25 15  
alăturat, pe ecran se vor afișa, în această ordine, numerele: 25 85

**b)** Descrieți succint, în limbaj natural (3-4 rânduri), algoritmul utilizat la punctul **a)** și justificați eficiența acestuia. **(4p.)**