

SUBIECTUL al II-lea

(30 de puncte)

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

1. Indicați o expresie C/C++ care are valoarea 1 oricare ar fi numărul strict pozitiv memorat în variabila reală x . (4p.)
 - a. $x - \text{floor}(x) \neq 0$
 - b. $x + \text{floor}(x) == 0$
 - c. $x - \text{floor}(x) >= 0$
 - d. $x + \text{floor}(x) <= 0$
2. Se consideră secvența de mai jos, în care toate variabilele sunt de tip întreg și memorează numere naturale nenule.

```
y=1;  
while ((y+1)*(y+1)<=x) y=y+1;  
cout<<.....; | printf("%d", .....);
```

Pentru ca executarea secvenței să determine afișarea pe ecran a celui mai mare număr natural pătrat perfect din intervalul $[1, x]$, zona punctată poate fi înlocuită cu: (4p.)

 - a. y
 - b. $y*y$
 - c. $(y-1)*(y-1)$
 - d. $(y+1)*(y+1)$

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

3. Variabilele reale x_A și y_A memorează abscisa, respectiv ordonata unui punct în sistemul de coordonate xOy , iar variabilele reale x_B și y_B memorează abscisa, respectiv ordonata unui alt punct în același sistem de coordonate.
Scrieți o secvență de instrucțiuni C/C++ care afișează pe ecran pătratul lungimii segmentului cu extremitățile în cele două puncte. (6p.)
4. Se citește un număr natural n și se cere să se scrie cea mai mică valoare naturală x ($x \geq n$) care se poate obține ca produs de două numere naturale consecutive.
Exemplu: dacă $n=10$ atunci $x=12$ ($12=3*4$).
 - a) Scrieți, în pseudocod, algoritmul de rezolvare pentru problema enunțată. (10p.)
 - b) Menționați rolul tuturor variabilelor care au intervenit în algoritmul realizat 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. În secvența de instrucțiuni de mai jos toate variabilele sunt de tip întreg.

```
for (i=1; i<=5; i++)  
{ for (j=1; j<=5; j++)  
    cout<<.....<<' '; | printf("%d ", .....);  
    cout<<endl; | printf("\n");  
}
```

| | | | | |
|---|---|---|---|---|
| 2 | 3 | 4 | 0 | 1 |
| 3 | 4 | 0 | 1 | 2 |
| 4 | 0 | 1 | 2 | 3 |
| 0 | 1 | 2 | 3 | 4 |
| 1 | 2 | 3 | 4 | 0 |

Indicați o expresie care poate înlocui punctele de suspensie astfel încât, în urma executării secvenței obținute, să se afișeze pe ecran valorile din figura de mai sus, în această ordine.

(4p.)

- a. $i+j\%5$ b. $i\%5+j$ c. $(i+j)\%5$ d. $i\%5+j\%5$

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

2. Se consideră tablourile unidimensionale $a=(100, 89, 9, 5, 3)$ și $b=(89, 10, 9, 8, 2, 1)$. Tabloul c este obținut prin interclasarea descrescătoare a elementelor din tablourile a și b . Scrieți elementele tabloului c , în ordinea în care ele apar în acesta. **(6p.)**

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 întregi cu cel mult 4 cifre. Cel puțin unul dintre elementele tabloului este strict pozitiv. Programul modifică tabloul, inserând după fiecare element strict pozitiv al acestuia, un nou element cu aceeași valoare. Programul afișează pe ecran valoarea actualizată a lui n și apoi, pe o linie nouă, elementele tabloului modificat, separate prin câte un spațiu.

Exemplu: pentru $n=6$ și tabloul $(4, -5, 0, 9, 9, -2)$,

se obțin $n=9$ și tabloul $(4, 4, -5, 0, 9, 9, 9, 9, -2)$.

(10p.)

4. Un număr natural cu cel puțin două cifre se numește **2-ordonat** dacă toate cifrele sale sunt în ordine crescătoare și valoarea absolută a diferenței dintre oricare două cifre aflate pe poziții consecutive este egală cu 2.

Exemplu: numărul **2468** este 2-ordonat, dar numărul **131** nu este 2-ordonat.

Se cere scrierea în fișierul **BAC.TXT** a tuturor numerelor naturale **2-ordonate**. Fiecare număr este scris pe câte o linie a fișierului.

Pentru determinarea numerelor cerute se utilizează un algoritm eficient din punctul de vedere al timpului de executare.

a) Descrieți în limbaj natural algoritmul utilizat, justificând eficiența acestuia. **(4p.)**

b) Scrieți programul C/C++ corespunzător algoritmului descris. **(6p.)**