

**TEST EXAMENUL DE BACALAUREAT NAȚIONAL DE INFORMATICĂ**

**SUBIECTUL I**

**(20 de puncte)**

**Pentru fiecare dintre itemii de la 1 la 5, scrieți pe foaia de examen litera corespunzătoare răspunsului corect. Fiecare răspuns corect se notează cu 4 puncte.**

1. Variabila  $x$  este de tip întreg și memorează un număr nenul. Indicați expresia C++ cu valoarea **1** dacă și numai dacă **30** este multiplu al lui  $x$ .

- a.  $30/(30/x)==0$     b.  $30/(30\%x)==0$     c.  $30\%(30x)==0$     d.  $30\%(30\%x)==0$

2. Subprogramul  $f$  este definit alăturat. Indicați ce se afișează în urma apelului  $f(12123)$ ;

```
void f (int x) { cout<<"*"; if(x>0) { f(x/100); cout<<x;} cout<<"+";}
```

- a. **\*\*\*\*+1+121+12123+**                      b. **\*\*\*\*+11212123++++**  
 c. **+12123+121+1+\*\*\*\***                      d. **\*+\*1+\*121+12123+\***

3. Utilizând metoda backtracking se generează toate posibilitățile de a forma șiruri de câte **4** animale din mulțimea {**cal, câine, miel, zebra, ponei**}, astfel încât pe oricare două poziții alăturate să nu se afle animale din submulțimea {**cal, miel, ponei**}. Primele opt soluții generate sunt, în această ordine: (**cal, câine, cal, câine**), (**cal, câine, cal, zebra**), (**cal, câine, câine, cal**), (**cal, câine, câine, câine**), (**cal, câine, câine, miel**), (**cal, câine, câine, zebra**), (**cal, câine, câine, ponei**), (**cal, câine, miel, câine**). Indicați numărul de soluții generate care au pe prima poziție un **câine** și pe ultima poziție o **zebra**.

- a. 18                      b. 20                      c. 14.                      d. 16

4. Un graf orientat cu **5** vârfuri este reprezentat prin matricea de adiacență alăturată. Indicați numărul vârfurilor cu **gradul exterior 3**.

<b>0</b>	<b>1</b>	<b>0</b>	<b>0</b>	<b>0</b>
<b>0</b>	<b>0</b>	<b>1</b>	<b>0</b>	<b>0</b>
<b>1</b>	<b>0</b>	<b>0</b>	<b>1</b>	<b>0</b>
<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>1</b>
<b>1</b>	<b>0</b>	<b>1</b>	<b>1</b>	<b>0</b>

5. Un arbore cu rădăcină are **8** noduri, numerotate de la **1** la **8**, și este reprezentat prin vectorul de „tați” (**0,1,1,1,2,2,6,7**). Indicați frunzele arborelui.

- a. **1,2,8**                      b. **3,4,7**                      c. **3,4,5,8**                      d. **1,5,8**

## SUBIECTUL II

(40 de puncte)

### 1. Algoritmul alăturat este reprezentat în pseudocod.

S-a notat cu  $a\%b$  restul împărțirii numărului natural  $a$  la numărul natural nenul  $b$  și cu  $[a]$  partea întreagă a numărului real  $a$ .

a). Scrieți numărul afișat dacă se citește valoarea 1234. (6p.)

b). Scrieți trei numere distincte din intervalul  $[10,10^3]$  care pot fi citite astfel încât, în urma executării algoritmului, pentru fiecare dintre acestea, să se afișeze valoarea 126. (6p.)

c.) Scrieți în pseudocod un algoritm echivalent cu cel dat, înlocuind adecvat structura **pentru...execută** cu o atribuire. (6p.)

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

```
citește n
  (număr întreg nenul)
dacă n<0 atunci
  n←-n
  ■
s←0
repetă
  x←n%10
  pentru i←1,x execută
  | s←s+i
  | ■
  n← [n/10]
până când n=0
scrie s
```

2. Variabila  $s$  memorează simultan următoarele date despre fiecare dintre cele 25 de spectacole din cadrul unei stagiuni: un cod, numărul de actori și vârstele acestora. La un spectacol sunt maximum 8 actori. Expresiile C++ de mai jos au ca valori numere naturale și reprezintă codul celui de al treilea spectacol, numărul de actori care joacă în acest spectacol, respectiv vârsta celui de al 5-lea astfel de actor din acest spectacol. Scrieți definiția unei structuri cu eticheta **spectacol**, care permite memorarea datelor despre un spectacol, și declarați corespunzător variabila  $s$ .

`s[2].cod s[2].nrActori s[2].varsta[4]` (6p.)

3. Variabila  $s$  poate memora un șir de cel mult 20 de caractere. Scrieți ce se afișează în urma executării secvenței de mai jos.. (6p.)

```
strcpy(s,"cartof");
cout<<s+4<<endl;
s[0]=s[0]-1;
s[1]=s[0]-1;
s[2]=s[0]+2;
s[3]=s[0]+3;
s[4]='\0';
cout<<s;
```

## SUBIECTUL II

(30 de puncte)

1. Subprogramul **produsimp** are doi parametri:

- **n**, prin care primește un număr natural ( $n \in [0, 10^9]$ );
- **p**, prin care furnizează produsul cifrelor impare distincte din scrierea acestuia, sau **-1** dacă nu există astfel de cifre.

Scrieți definiția completă a subprogramului.

**Exemplu:**

dacă **n=1325325**, după apel **p=15** ( $15=1 \cdot 3 \cdot 5$ ), iar dacă **n=22420**, după apel **p=-1**. (10p.)

2. Scrieți un program C++ care citește de la tastatură un număr natural, **n** ( $n \in [2, 20]$ ), și construiește în memorie un tablou bidimensional cu **n** linii și **n** coloane, având proprietățile:

- toate elementele situate pe diagonala principală sunt nule;
- fiecare linie conține, începând cu diagonala principală, de la stânga la dreapta, un șir strict descrescător de numere consecutive, iar începând cu diagonala principală, de la dreapta la stânga, tot un șir strict descrescător de numere consecutive.

Programul afișează pe ecran tabloul construit, fiecare linie a tabloului pe câte o linie a ecranului, cu elementele aflate pe aceeași linie separate prin câte un spațiu .

**Exemplu:** dacă **n=5** se afișează pe ecran tabloul alăturat. (10p.)

0 4 3 2 1

4 0 4 3 2

3 4 0 4 3

2 3 4 0 4

1 2 3 4 0

3. Fișierul **bac.txt** conține, **în ordine crescătoare**, cel puțin trei și cel mult  $10^6$  numere naturale din intervalul  $[0, 10^9]$ , separate prin câte un spațiu. Se cere să se afișeze pe ecran, în ordine strict crescătoare, separate prin câte un spațiu, numai numerele care apar în fișier de exact trei ori. Dacă nu există niciun astfel de număr, se afișează pe ecran mesajul **nu există**. Proiectați un algoritm eficient din punctul de vedere al memoriei utilizate și al timpului de executare.

**Exemplu:** dacă fișierul conține numerele **11 12 12 12 50 50 50 66 66 78 90 100 100 100**

pe ecran se afișează, în această ordine, numerele **12 50 100**

a. Scrieți programul C++ corespunzător algoritmului proiectat. (8p.)

b. Descrieți în limbaj natural algoritmul proiectat, justificând eficiența acestuia. (2p.)