

EXAMENUL DE BACALAUREAT – 2009
Proba scrisă la INFORMATICĂ
PROBA E, limbajul C/C++
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. Care dintre următoarele instrucțiuni **C/C++** atribuie variabilei întregi **t** valoarea **-1** dacă și numai dacă variabilele întregi **a** și **b** sunt nenule și au semne diferite? (4p.)
- a. **if ((a>0)|| (b<0)) t=-1;** b. **if ((a>0)&&(b<0)) t=-1;**
c. **if (a*b<0) t=-1;** d. **if (a*b>0) t=-1;**

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

2. Se consideră algoritmul alăturat, descris în pseudocod.

S-a notat cu **x%y** restul împărțirii numărului natural **x** la numărul natural nenul **y**.

- a) Scrieți valorile care se vor afișa în urma executării algoritmului dacă se citesc numerele **a=105**, **b=118** și **k=7**. (6p.)
- b) Dacă pentru **k** se citește valoarea 7, iar pentru **a** valoarea 2009, scrieți cea mai mare valoare care se poate citi pentru variabila **b**, astfel încât numărul afișat să fie **-1**. (4p.)
- c) Scrieți programul **C/C++** corespunzător algoritmului dat. (10p.)

```
citește a, b, k
      (numere naturale)

t ← a
p ← 0
cât timp t ≤ b execută
    dacă k = t%10 atunci
        scrie t
        p ← 1
    t ← t+1
dacă p = 0 atunci
    scrie -1
```

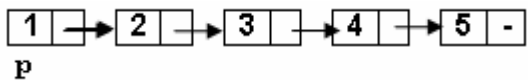
- d) Scrieți în pseudocod un algoritm echivalent cu cel dat în care să se înlocuiască structura **cât timp...execută** cu o structură repetitivă cu test final. (6p.)

Subiectul II (30 de puncte)

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

1. În declarația alăturată, variabila **p** memorează în câmpul **x** abscisa, iar în câmpul **y** ordonata unui punct din planul **xOy**. Dacă punctul se află în semiplanul din dreapta axei **Oy** (dar nu pe această axă), care dintre expresiile de mai jos are valoarea 1? (4p.)
- ```
struct
{
 float x;
 float y;
}p;
```
- a.  $p.x > 0$                       b.  $p.y > 0$                       c.  $x.p + y.p > 0$                       d.  $p(x) + p(y) > 0$
2. Ce memorează variabila **s**, de tip șir de caractere, după executarea instrucțiunilor de mai jos? (4p.)
- ```
strncpy(s, "informatica", strlen("2009"));
s[strlen("2009")] = '\0';
strcat(s, "BAC");
```
- a. **info** b. **infoBAC** c. **BACinfo** d. **InformaticaBAC**

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

3. Se consideră un arbore cu 6 noduri, numerotate de la 1 la 6, reprezentat prin matricea de adiacență dată alăturat. Scrieți toate nodurile care pot fi alese ca rădăcină a arborelui astfel încât acesta să aibă un număr par de frunze. (6p.)
- | | | | | | |
|---|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 |
4. Fiecare element al unei liste simplu înlănțuite, alocată dinamic, reține în câmpul **nr** un număr întreg, iar în câmpul **urm** adresa următorului element din listă sau **NULL** dacă nu există un element următor. Ce valori au variabilele întregi **a** și **b** după executarea secvenței alăturate, dacă variabila **p** reține adresa primului element al listei de mai jos, iar variabila **q** este de același tip cu **p**? (6p.)
- ```
q=p;
a=p->urm->nr;
while (q->urm!=NULL)
{
 q->urm->nr=q->nr*a;
 q=q->urm;
}
b=q->nr;
```
- 
5. Scrieți un program **C/C++** care citește de la tastatură un număr natural **n** ( $1 \leq n \leq 20$ ), elementele unei matrice cu **n** linii și **n** coloane, numere întregi din intervalul  $[-100, 100]$  și afișează pe ecran diferența **m1-m2**, unde **m1** este media aritmetică a elementelor strict pozitive ale matricei, situate deasupra diagonalei principale, iar **m2** este media aritmetică a elementelor strict pozitive ale matricei, situate sub diagonala principală, ca în exemplu. Cele două medii se consideră egale cu 0 dacă nu există valori strict pozitive în zonele corespunzătoare. Exemplu: pentru **n=4** și matricea alăturată se afișează valoarea **0.25** (**m1=2.75**, calculată din elementele aflate deasupra diagonalei principale, marcate cu chenar, și **m2=2.5**, calculată din elementele subliniate). (10p.)
- |          |          |          |          |
|----------|----------|----------|----------|
| -1       | <u>2</u> | -4       | <u>5</u> |
| 0        | 6        | <u>3</u> | <u>1</u> |
| <u>2</u> | <u>4</u> | 2        | 0        |
| <u>3</u> | -5       | <u>1</u> | -3       |

**Subiectul III (30 de puncte)**

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

1. La un bal mascat, magazia școlii pune la dispoziția elevilor **10** pelerine, **10** măști și **10** pălării divers colorate. Algoritmul de generare a tuturor posibilităților de a obține un costum format dintr-o pălărie, o mască și o pelerină este similar cu algoritmul de generare a : **(4p.)**
- a. elementelor produsului cartezian                      b. aranjamentelor  
c. permutărilor                                                      d. submulțimilor

**Scrieți pe foaia de examen răspunsul la următoarea întrebare:**

2. Se consideră subprogramul **f**, definit alăturat. Ce se afișează la apelul **f('a');**?
- (6p.)**

```
void f(char c)
{
 if (c != 'e')
 {
 f(c+1);
 cout<<c; | printf("%c",c);
 }
}
```

3. Funcția **verif** primește prin intermediul parametrului **n** un număr natural format din cel mult 9 cifre, și prin intermediul parametrului **a**, un număr natural nenul ( $2 \leq a \leq 9$ ). Funcția returnează valoarea **1** dacă **n** este un număr format din cifre aparținând intervalului închis **[0, a]** și valoarea **0** în caz contrar.

a) Scrieți definiția completă a funcției **verif**. **(4p.)**

b) Spunem că **n** poate fi o reprezentare în baza **b** ( $1 < b \leq 10$ ), dacă toate cifrele lui **n** sunt strict mai mici decât **b**. Scrieți un program care citește de la tastatură o valoare naturală **n** cu cel mult 9 cifre și, utilizând apeluri ale funcției **verif**, afișează pe ecran, în ordine crescătoare, cu spații între ele, toate valorile lui **b** pentru care valoarea citită **nu** poate fi o reprezentare în baza **b**. **(6p.)**

**Exemplu:** Pentru **n=4101**, se afișează **2 3 4**.

4. Fișierul text **bac.txt** conține cel mult **1000** de numere întregi de cel mult **9** cifre fiecare, numerele fiind separate prin câte un spațiu; printre numerele din fișier există cel puțin două numere pozitive, aflate pe poziții consecutive.

a) Scrieți un program **C/C++** care afișează două numere pozitive, aflate unul după altul în fișier, a căror sumă este maximă, utilizând un algoritm eficient din punct de vedere al timpului de executare și al spațiului de memorie utilizat. Dacă există mai multe soluții, se afișează doar acea pereche pentru care diferența dintre cele două numere este maximă. Numerele vor fi afișate pe ecran, în ordinea din fișier, separate printr-un spațiu.

**Exemplu:** dacă fișierul conține numerele: **-2 2 16 4 -1 25 -2 8 12 7 13** se vor afișa numerele **16 4**, în această ordine, cu un spațiu între ele. **(6p.)**

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