

**EXAMENUL DE BACALAUREAT – 2008**  
**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. În secvența alăturată de program, instrucțiunea de afișare se va executa de un număr de ori egal cu: (4p.)
- |       |       |      |       |
|-------|-------|------|-------|
| a. 24 | b. 21 | c. 3 | d. 30 |
|-------|-------|------|-------|
- ```
for (i=1; i<=3; i++)
for (j=10; j>=i+1; j--)
    cout<<j; | printf("%d", j);
```

**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 care este valoarea afișată dacă se citesc valorile  $x=148$  și  $y=203$ . (6p.)
- b) Scrieți un set de valori care pot fi citite pentru variabilele  $x$  și  $y$  astfel încât, după executarea algoritmului alăturat, să se afișeze exact 5 valori. (4p.)
- c) Scrieți programul C/C++ corespunzător algoritmului dat. (10p.)
- d) Scrieți în pseudocod un algoritm echivalent cu cel dat, în care să se înlocuiască structura **cât timp...execută** cu o structură **pentru...execută**. (4p.)

```
citește x, y
    (numere naturale)
x ← x % 10
y ← y % 10
dacă y < x atunci
    aux ← y
    y ← x
    x ← aux
■
cât timp x ≤ y execută
    scrie x * 10 + y
    x ← x + 1
    y ← y - 1
■
```

**Subiectul II (30 de puncte)**

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

1. Graful neorientat cu 60 de noduri, numerotate de la 1 la 60, are numai muchiile [1, 60], [60, 20], [2, 30] și [4, 30]. Numărul componentelor conexe ale grafului este egal cu: (4p.)  
a. 3                                      b. 56                                      c. 54                                      d. 0
2. Într-un arbore cu rădăcină cu 10 noduri, numerotate de la 1 la 10, nodul 10 este rădăcină, iar între celelalte noduri există relația: nodul cu numărul  $i+1$  este tatăl celui cu numărul  $i$ , pentru  $i \in \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$ . Vectorul de tați al arborelui astfel definit, este: (4p.)  
a. (0, 1, 2, 3, 4, 5, 6, 7, 8, 9)                                      b. (1, 2, 3, 4, 5, 6, 7, 8, 9, 0)  
c. (2, 3, 4, 5, 6, 7, 8, 9, 10, 0)                                      d. (9, 8, 7, 6, 5, 4, 3, 2, 1, 0)

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

3. Într-o listă simplu înlănțuită alocată dinamic sunt memorate în ordine, următoarele valori:  

|   |    |   |    |   |    |   |    |   |    |   |    |    |
|---|----|---|----|---|----|---|----|---|----|---|----|----|
| 2 | -> | 3 | -> | 4 | -> | 7 | -> | 5 | -> | 9 | -> | 14 |
|---|----|---|----|---|----|---|----|---|----|---|----|----|

Dacă  $p$  este adresa primului element al listei și fiecare element reține în câmpul  $urm$  adresa elementului următor, care este informația din elementul a cărui adresă o va reține  $p$  în urma executării secvenței alăturate: (6p.)

```
p=p->urm;
while (p->urm->urm!=0)
    p=p->urm->urm;
```
4. Ce se va afișa în urma executării secvenței alăturate, în care variabila  $c$  memorează un șir cu cel mult 20 de caractere, iar  $i$  este o variabilă de tip întreg? (6p.)  

```
char c[20]="tamara", *p;
for(i=0; i<strlen(c); i=i+1)
    {p=strchr(c, 'a'); cout<<p-c; }
```
5. Scrieți programul C/C++ care citește de la tastatură un număr natural  $n$  ( $n \leq 20$ ), construiește în memorie și afișează pe ecran, matricea cu  $n$  linii și  $n$  coloane, în care se vor memora în ordinea crescătoare a valorii, pe linii și coloane, primele  $n^2$  numere naturale nenule, pare, care nu sunt divizibile cu 3. Fiecare linie a matricei se va afișa pe câte o linie a ecranului, cu elementele de pe aceeași linie separate prin câte un spațiu. **Exemplu:** pentru  $n=4$  se va construi și afișa matricea alăturată. (10p.)  

|             |             |
|-------------|-------------|
| 1 2 4 8     | 1 2 4 8     |
| 10 14 16 20 | 10 14 16 20 |
| 22 26 28 32 | 22 26 28 32 |
| 34 38 40 44 | 34 38 40 44 |

**Subiectul III (30 de puncte)**

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

1. Completarea unui bilet de LOTO presupune colorarea a 6 numere din cele 49 înscrise pe bilet. O situație statistică pe o anumită perioadă de timp arată că cele mai frecvente numere care au fost extrase la LOTO sunt: 2, 20, 18, 38, 36, 42, 46, 48. Câte bilete de 6 numere se pot completa folosind doar aceste valori știind că numărul 42 va fi colorat pe fiecare bilet. (4p.)
- a. 21                      b. 6!                      c. 42                      d. 56

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

2. Se consideră definit subprogramul  $f$ .

Scrieți două valori naturale,  $x_1$  și  $x_2$ , ( $x_1 < 12$  și  $x_2 < 12$ ) pentru care  $f(x_1) = f(x_2)$ . (6p.)

```
int f(int i)
{
    if (i > 12) return 1;
    else return 1 + f(i + 2);
}
```

3. a) Subprogramul **max** primește ca parametru un tablou unidimensional  $x$  cu cel mult 100 de elemente numere întregi, care sunt, în ordine, termenii unei progresii aritmetice și un număr natural  $n$ , care reprezintă dimensiunea tabloului. Scrieți definiția completă a subprogramului **max** care returnează cel mai mare termen al progresiei aritmetice. Alegeți un algoritm de rezolvare eficient din punct de vedere al timpului de executare și al spațiului de memorie utilizat. (6p.)

b) Explicați în limbaj natural metoda utilizată justificând eficiența acesteia. (4p.)

c) În fișierul **numere.txt**, se află memorat pe prima linie un număr natural  $n$  ( $n < 100$ ), iar pe fiecare dintre următoarele  $n$  linii, câte  $n$  numere întregi cu cel mult 4 cifre fiecare. Scrieți programul **C/C++** care citește din fișier datele existente, determină liniile din fișier pe care s-au memorat în ordine termenii unei progresii aritmetice și afișează pe ecran, folosind apeluri ale subprogramului **max** cel mai mare număr (diferit de cel situat pe prima linie) din fișier, care în plus este termenul unei progresii aritmetice. (10p.)

**Exemplu:** dacă fișierul **numere.txt** are conținutul alăturat, se va afișa 50, deoarece progresiile aritmetice sunt:

(-9 -7 -5 -3 -1),  
(50 40 30 20 10) și  
(18 17 16 15 14)

```
5
5 7 3 1 9
-9 -7 -5 -3 -1
2 5 8 14 11
50 40 30 20 10
18 17 16 15 14
```