

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 este valoarea pe care poate să o aibă inițial variabila întreagă **x** dacă, la sfârșitul executării secvenței alăturate, variabila întreagă **y** are valoarea **2**? (4p.)

```
y=0;  
do  
{ x=x/10;  
  y=y+1;  
} while(x%100==0);
```

- a. 300 b. 5000 c. 120 d. 0

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

2. Se consideră algoritmul alăturat, descris în pseudocod.

- a) Scrieți succesiunea de caractere pe care le va afișa algoritmul dacă se citesc, în aceasta ordine, valorile **2**, respectiv **9**. (6p.)

- b) Scrieți numărul de perechi de valori aparținând intervalului **[1, 20]**, care pot fi citite pentru variabilele **x** și **y**, astfel încât rezultatul afișat să fie format din exact **12** caractere ? (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 repetitivă **cât timp...execută** cu o structură repetitivă de alt tip. (6p.)

```
citește x, y  
    (numere naturale)  
dacă x < y atunci  
    x ← x - y  
    y ← x + y  
    x ← y - x  
■  
cât timp x ≥ y execută  
    scrie 'A'  
    x ← x - y  
    scrie 'B'  
■
```

Subiectul II (30 de puncte)

Pentru fiecare dintre itemii următori, scrieți pe foaia de examen litera care corespunde răspunsului corect.

1. Într-o listă liniară simplu înlănțuită, alocată dinamic, fiecare element memorează în câmpul **nr** un număr întreg, iar în câmpul **urm** adresa elementului următor din listă sau valoarea **NULL** dacă nu există un element următor. Lista conține exact trei elemente ale căror adrese sunt memorate în variabilele **p**, **q** și **r**. Știind că **p->nr==1**, **q->nr==2**, **r->nr==3**, **p->urm!=NULL** și **r->urm==q**, care este ordinea numerelor din listă? (4p.)
- a. 1 3 2 b. 1 2 3 c. 2 1 3 d. 3 2 1
2. Care dintre următoarele variante reprezintă o declarație corectă pentru o variabilă **x** care memorează simultan codul de identificare al unui candidat la un examen, exprimat printr-un număr natural de cel mult 4 cifre și media obținută de acesta la examen, exprimată printr-un număr real? (4p.)
- a. `struct x { int cod;
 float media;};`                      b. `struct {int cod;
 float media;} x;`
- c. `int x.cod ;
 float x.media;`                      d. `struct candidat {int x.cod;
 float x.media;};`

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

3. Care este gradul maxim posibil și care este gradul minim posibil pentru un nod dintr-un arbore cu **n** noduri (**n>1**)? (6p.)
4. Ce se va afișa în urma executării secvenței de program alăturate știind că variabila **a** memorează un șir cu cel mult 100 de caractere, iar variabila **i** este de tip întreg ? (6p.)
- ```
strcpy(a,"bacalaureat");
cout<<strlen(a)<<endl; | printf("%d\n",strlen(a));
for(i=0;i<strlen(a);i++)
 if(strchr("aeiou",a[i])!=0)
 cout<<'*'; | printf('*');
```
5. Scrieți un program **C/C++** care citește de la tastatură un număr natural **n** (**2<n<20**), construiește în memorie și afișează pe ecran o matrice cu **n** linii și **n** coloane, în care fiecare element de pe diagonala secundară are valoarea **n**, fiecare element aflat deasupra diagonalei secundare este mai mic cu o unitate decât vecinul aflat pe aceeași linie în dreapta lui și fiecare element aflat sub diagonala secundară este mai mare cu o unitate decât vecinul aflat pe aceeași linie în stânga lui. Elementele matricei vor fi afișate pe ecran, câte o linie a matricei pe câte o linie a ecranului cu câte un spațiu între elementele fiecărei linii. (10p.)
- Exemplu:** pentru **n=5** se va afișa matricea alăturată.
- |   |   |   |   |   |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| 2 | 3 | 4 | 5 | 6 |
| 3 | 4 | 5 | 6 | 7 |
| 4 | 5 | 6 | 7 | 8 |
| 5 | 6 | 7 | 8 | 9 |

**Subiectul III (30 de puncte)**

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

- |                                                                                                                                                   |                                                                                                                             |
|---------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------|
| <p>1. Subprogramul <b>f</b> are definiția alăturată. Ce se va afișa în urma apelului <b>f(12345)</b>?</p> <p style="text-align: right;">(4p.)</p> | <pre>void f(long n) {     if (n&gt;9)     {         cout&lt;&lt;n/100;   printf("%d",n/100);         f(n/10);     } }</pre> |
| <p>a. 1231210                      b. 123121</p>                                                                                                  | <p>c. 1234123121                      d. 123</p>                                                                            |

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

2. Un algoritm generează în ordine descrescătoare toate numerele de **5** cifre, fiecare dintre ele având cifrele în ordine strict crescătoare. Știind că primele cinci soluții generate sunt **56789**, **46789**, **45789**, **45689**, **45679**, precizați care sunt ultimele **trei** soluții generate, în ordinea generării. (6p.)
3. Scrieți definiția completă a subprogramului **interval** care are doi parametri **a** și **n**, prin care primește un tablou unidimensional cu maximum **100** de numere naturale mai mici decât **1000** și respectiv numărul efectiv de elemente din tabloul unidimensional. Subprogramul returnează numărul de elemente din tabloul unidimensional care aparțin intervalului închis determinat de primul și respectiv ultimul element al tabloului.  
**Exemplu:** dacă tabloul are **6** elemente și este de forma (12, 27, 6, 8, 9, 2), subprogramul va returna valoarea **5**. (10p.)
4. În fișierul **numere.txt** pe prima linie este memorat un număr natural **n** ( $n \leq 10000$ ), iar pe linia următoare un șir de **n** numere naturale distincte două câte două, separate prin câte un spațiu, cu maximum **4** cifre fiecare. Se cere afișarea pe ecran a poziției pe care s-ar găsi primul element din șirul aflat pe linia a doua a fișierului, în cazul în care șirul ar fi ordonat crescător. Numerotarea pozițiilor elementelor în cadrul șirului este de la **1** la **n**. Alegeți un algoritm de rezolvare eficient din punct de vedere al memoriei utilizate și al timpului de executare.  
**Exemplu:** dacă fișierul **numere.txt** conține:  
**6**  
**267 13 45 628 7 79**  
se va afișa **5**, deoarece primul element din șirul inițial, **267**, s-ar găsi pe poziția a cincea în șirul ordonat crescător (**7 13 45 79 267 628**).  
**a)** Descrieți succint, în limbaj natural, strategia de rezolvare și justificați eficiența algoritmului ales. (4p.)  
**b)** Scrieți programul **C/C++** corespunzător algoritmului ales. (6p.)