

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.

- | | |
|--|--|
| <p>1. Stabiliți ce se afișează în urma executării secvenței de instrucțiuni C/C++ alăturate, dacă y este o variabilă reală, iar x o variabilă întreagă. (4p.)</p> | <pre>y=10.1234;
x=(int)(y*100)/100;
printf("%d",x); cout<<x;</pre> |
| <p>a. 1012.34 b. 10.12 c. 0.12 d. 10</p> | |

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

- | | |
|---|---|
| <p>2. Se consideră algoritmul alăturat, reprezentat în pseudocod.</p> <p>S-a notat cu x y relația „x divide pe y” sau „y este divizibil cu x” și cu [z] partea întreagă a numărului real z.</p> <p>a) Scrieți valoarea care se va afișa dacă se citește numărul 245. (6p.)</p> <p>b) Scrieți două numere naturale care pot fi citite pentru variabila a astfel încât valoarea afișată în fiecare caz să fie 10. (4p.)</p> <p>c) Scrieți programul C/C++ corespunzător algoritmului dat. (10p.)</p> <p>d) Scrieți în pseudocod un algoritm echivalent cu cel dat, algoritm în care să se înlocuiască fiecare structură cât timp...execută cu câte o structură repetitivă de alt tip. (6p.)</p> | <pre>citește a (număr natural)
x←2
k←0
cât timp a>1 execută
 c←0
 cât timp x a execută
 c←x
 a←[a/x]
 ■
 dacă c≠0 atunci
 k←k+x
 ■
 x←x+1
 ■
scrie k</pre> |
|---|---|

Subiectul II (30 de puncte)

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

1. Pentru o matrice **t** cu 8 linii și 8 coloane, numerotate de la 0 la 7, cu elemente numere întregi, secvența de program alăturată, în care variabilele **z**, **i**, și **j** sunt de tip întreg, determină, în urma executării ei, memorarea în **z** a sumei tuturor elementelor situate:

```
z=0;
for(i=0;i<8;i++)
    for(j=7-i;j<8;j++)
        z=z+t[i][j];
```

(4p.)

- a. sub diagonala secundară, inclusiv diagonala secundară
b. deasupra diagonalei principale, inclusiv diagonala principală
c. strict sub diagonala principală
d. strict deasupra diagonalei secundare

2. Se consideră lista simplu înlănțuită, alocată dinamic, în care fiecare nod memorează în câmpul **nr** o valoare întreagă și în câmpul **urm** adresa nodului următor sau **NULL** dacă este ultimul nod din listă.

```
p=prim;
k=0;
while(p->urm!=NULL && k>=0)
{if(p->nr%2==0) k=k+1;
 else k=k-1;
 p=p->urm;
}
printf("%d",p->nr); | cout<<p->nr;
```

În listă sunt memorate, în această ordine, valorile 4, 3, 2, 5, 7, 9, 6, 1, 8. Variabila **prim** reține adresa primului element din listă, variabila **p** este de același tip cu **prim**, iar variabila **k** este de tip întreg. Care este valoarea ce se va afișa în urma executării secvenței alăturate ?

(4p.)

- a. 9 b. 4 c. 3 d. 8

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

3. Se consideră arborele cu 13 noduri, numerotate de la 1 la 13, și mulțimea muchiilor {[1,4], [2,5], [3,8], [4,7], [4,9], [4,11], [6,3], [6,10], [6,12], [5,6], [13,2], [2,9]}. Dacă se alege nodul numerotat cu 2 drept rădăcină, care este vectorul de „tați” pentru acest arbore? (6p.)
4. Fie graful neorientat cu 6 noduri, numerotate de la 1 la 6, și muchiile [1,2], [1,3], [1,4], [2,3], [2,4], [3,4], [3,5], [4,5], [4,6], [5,6]. Care este numărul maxim de muchii ce pot fi eliminate astfel încât graful parțial obținut să-și păstreze proprietatea de graf hamiltonian? (6p.)
5. Scrieți programul **C/C++** care citește de la tastatură un text cu cel mult 100 de caractere (doar litere ale alfabetului englez și spații), construiește în memorie și apoi afișează pe ecran șirul de caractere obținut din șirul inițial, în care se inserează după fiecare vocală caracterul *. Se consideră vocale literele a, e, i, o, u, A, E, I, O, U. Dacă textul citit nu conține vocale, se va afișa mesajul **FARA VOCALE**.

Exemplu: dacă se citește de la tastatură textul **Examenul de bacalaureat** se va obține și afișa: **E*xa*me*nu*1 de* ba*ca*la*u*re*a*t.** (10p.)

Subiectul III (30 de puncte)

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

1. Problema generării tuturor codurilor formate din exact **4** cifre nenule, cu toate cifrele distincte două câte două, este similară cu generarea tuturor: **(4p.)**
- a. aranjamentelor de **9** elemente luate câte **4** b. permutărilor elementelor unei mulțimi cu **4** elemente
- c. elementelor produsului cartezian **AxAxAxA** unde **A** este o mulțime cu **9** elemente d. submulțimilor cu **4** elemente ale mulțimii **{1, 2, 3, 4, 5, 6, 7, 8, 9}**

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

2. Pentru definiția alăturată a subprogramului **f**, scrieți ce valoare are **f(8)**. Dar **f(1209986)**? **(6p.)**
- ```
int f(long x)
{ int y,z;
 if (x==0) return x;
 else {y=x%10;
 z=f(x/10);
 if(y>z) return y;
 else return z;
 }
}
```
3. Scrieți definiția completă a subprogram **max**, cu trei parametri, **a**, **b**, **c**, care primește prin intermediul parametrilor **a** și **b** două numere reale cu exact două cifre la partea întreagă și exact două zecimale fiecare. Subprogramul determină cel mai mare număr real dintre următoarele patru valori: **a**, **b** și numerele reale obținute din **a** și **b** prin interschimbarea părții întregi cu partea fracționară în cadrul aceluiași număr. Această valoare este furnizată prin intermediul parametrului real **c**.  
**Exemplu:** dacă **a=33.17** și **b=15.40**, **c** va avea valoarea **40.15** (cea mai mare valoare dintre **33.17**, **15.40**, **17.33** și **40.15**) **(10p.)**
4. Se citește de pe prima linie a fișierului **numere.in** un număr natural **n** (**0<n<10000**) și, de pe a doua linie a fișierului, **n** numere naturale din intervalul **[1,100]** și se cere să se afișeze pe ecran, despărțite prin câte un spațiu, numărul sau numerele întregi din intervalul **[1,100]** care **nu** apar printre numerele citite. Dacă pe a doua linie a fișierului apar toate numerele din intervalul precizat, se va afișa mesajul **NU LIPSESTE NICIUN NUMAR**. Alegeți un algoritm de rezolvare eficient din punctul de vedere al timpului de executare.  
**Exemplu:** pentru fișierul **numere.in** cu următorul conținut
- 12**  
**4 2 3 1 6 5 7 8 9 11 10 100**  
se vor afișa valorile **12 13 ... 99**.
- a) Explicați în limbaj natural metoda utilizată, justificând eficiența acesteia (4-6 rânduri). **(4p.)**
- b) Scrieți programul **C/C++** ce rezolvă problema enunțată, corespunzător metodei descrise la punctul a). **(6p.)**