

EXAMENUL DE BACALAUREAT – 2007
Proba scrisă la INFORMATICĂ
PROBA E, limbajul C/C++

Varianta 79

- ♦ Toate subiectele sunt obligatorii. Se acordă 10 puncte din oficiu.
- ♦ Timpul efectiv de lucru este de 3 ore.
- ♦ În programele cerute la subiectele II și III, 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 (40 de puncte)

Pentru fiecare din itemii de la 1 la 8, scrieți pe foaia de examen litera corespunzătoare răspunsului corect. Fiecare răspuns corect se notează cu 5 puncte.

1. Într-un liceu sunt n clase iar în fiecare clasă sunt câte 25 de elevi. Problema determinării tuturor echipelor de n elevi, câte unul din fiecare clasa, este similară cu generarea tuturor:
 - a. elementelor produsului cartezian A^n , unde $A=\{1,2,\dots,25\}$
 - b. submulțimilor cu n elemente ale mulțimii $\{1,2,\dots,25\}$
 - c. permutărilor mulțimii $\{1,2,\dots,n\}$
 - d. partițiilor mulțimii $\{1,2,\dots,n\}$
2. Pentru o matrice a cu 10 linii și 10 coloane, stabiliți ce calculează secvența de program alăturată în variabila s :


```
s=0;
for(i=0;i<10;i++)
    for(j=i;j<10;j++)
        s=s+a[i][j];
```

 - a. suma elementelor situate strict deasupra diagonalei principale
 - b. suma elementelor situate strict deasupra diagonalei secundare
 - c. suma elementelor situate deasupra diagonalei principale, inclusiv diagonala principală
 - d. suma elementelor situate strict sub diagonala principală
3. Care din următoarele expresii este adevărată dacă și numai dacă numărul întreg a este un număr impar pozitiv?
 - a. $a\%2==0\&\&a>0$
 - b. $!(a\%2==0\&\&a<=0)$
 - c. $a>0||a\%2==1$
 - d. $!(a<=0||a\%2==0)$
4. Se consideră lista simplu înlănțuită în care fiecare nod memorează în câmpul nr o valoare întreagă și în câmpul urm adresa nodului următor. Un nod al listei este de tipul nod . În listă sunt memorate, în această ordine, valorile 1,2,3,4,5. Dacă variabila p reține adresa primului nod din listă, ce se va afișa în urma apelului $f(p)$?


```
void f(nod*p)
{
    if(p!=NULL)
    {
        f(p->urm);
        printf("%d",p->nr); | cout<<p->nr;
    }
}
```

 - a. 12345
 - b. 54321
 - c. 5
 - d. 1
5. Ce se va afișa pe ecran în urma executării următoarelor instrucțiuni, dacă pentru variabila întreagă a se citesc, în ordine, numerele: 1234, 234, 52, 25, 5432, 819 ?


```
for(i=1;i<=6;i++)
{
    scanf("%d",&a); | cin>>a;
    if(i%2==0)
        printf("%d",a/100%10); | cout<<a/100%10;
    else
        printf("%d",a/10%10); | cout<<a/10%10;
}
```

 - a. 230241
 - b. 432221
 - c. 220241
 - d. 325038
6. Pentru graful neorientat conex cu 7 noduri, în care toate nodurile au același grad, care dintre următoarele variante **nu** poate fi gradul unui nod?
 - a. 3
 - b. 2
 - c. 4
 - d. 6
7. Se consideră arborele cu rădăcină având 9 noduri numerotate de la 1 la 9, dat prin vectorul de tați $t=(5,5,2,2,0,5,9,9,5)$. Câte lanțuri distincte de lungime 3 care au ca extremități noduri terminale (frunze) există? Lanțul de lungime 3 (6,5,9,7) se consideră identic cu lanțul (7,9,5,6).
 - a. 8
 - b. 2
 - c. 5
 - d. 4

8. Se utilizează metoda backtracking pentru a determina toate modalitățile de a descompune pe 8 ca sumă de numere naturale nenule distincte (făcând abstracție de ordinea termenilor) și se obțin soluțiile 1+2+5, 1+3+4, 1+7, 2+6, 3+5, 8. Câte sume diferite, cu patru termeni, se obțin utilizând aceeași metodă, pentru descompunerea numărului 15?
- a. 10 b. 1 c. 6 d. 5

SUBIECTUL II (20 de puncte)

Se consideră programul pseudocod alăturat:

S-a notat cu $[x]$ partea întreagă a numărului real x .

1. Ce se va afișa pentru $a=12$? (5p.)
2. Scrieți o valoare pentru a astfel încât să se afișeze valorile 3 și 1 (2p.)
3. Scrieți programul C/C++ corespunzător algoritmului dat. (8p.)
4. Scrieți un algoritm echivalent cu algoritmul dat, dar care să utilizeze alt tip de structură repetitivă. (5p.)

citește a (a număr natural, $a > 1$)

$b \leftarrow 1/a$

$c \leftarrow 0$

cât timp $b < 1$ execută

$b \leftarrow b * 10$

$c \leftarrow c + 1$

$b \leftarrow [b]$

scrie c, b

SUBIECTUL III (30 de puncte)

1. Scrieți un subprogram **elimin** care elimină prima și ultima cifră a unui număr natural cu minimum 3 cifre și maximum 9 cifre. Subprogramul va avea 2 parametri: numărul inițial n și numărul obținut prin eliminarea celor două cifre, m . De exemplu, pentru $n=12345678$, m va avea valoarea 234567. (10p.)
2. Fie o listă simplu înălțuită circulară, în care fiecare nod reține în câmpul **inf** un caracter și în câmpul **next** adresa nodului următor. Se consideră subprogramele:
 - **creare** care primește prin parametrul **cuv** un cuvânt de cel mult 20 de litere mici ale alfabetului englez, creează o listă circulară cu toate literele cuvântului **cuv**, în ordinea în care sunt așezate în cuvânt și returnează adresa elementului din listă care reține prima literă din cuvânt.
 - **afisare** care afișează pe ecran caracterele din toate nodurile listei circulare pornind de la adresa nodului transmisă prin parametrul **p** și vizitând toate nodurile listei o singură dată.

Un nod al listei este de tipul **nod**.

 - a) Scrieți antetele pentru subprogramele **creare** și **afisare**. (2 p.)
 - b) Definiți tipurile de date necesare și scrieți programul C/C++ care citește de la tastatură un cuvânt de maxim 20 de litere, creează o listă circulară cu toate literele cuvântului și afișează, câte una pe linie, toate permutările circulare ale cuvântului, utilizând apeluri ale subprogramelor de la punctul a).
De exemplu, dacă se citește de la tastatură cuvântul **carte**, se vor afișa cuvintele **carte**, **artec**, **rteca**, **tecar**, **ecart**. (8 p.)
3. Pe prima linie a fișierului **NUMERE.TXT** se află un număr natural n ($0 < n < 10000$) iar pe următoarea linie exact n valori naturale de cel mult 2 cifre fiecare, valori despărțite prin câte un spațiu.
 - a) Scrieți programul C/C++ care citește din fișier numărul n și apoi cele n numere și afișează pe ecran, despărțite prin câte un spațiu, numărul sau numerele care apar de cele mai multe ori pe linia a doua a fișierului. Se va utiliza un algoritm eficient (ca timp de executare și gestionare a memoriei). De exemplu, dacă fișierul **NUMERE.TXT** are următorul conținut:
10
21 3 4 21 4 5 21 3 6 4
atunci programul va afișa numerele **21** și **4**, deoarece fiecare apare de câte 3 ori, iar celelalte numere apar de mai puțin de 3 ori. (8p.)
 - b) Descrieți pe scurt metoda folosită explicând eficiența acesteia (3-4 rânduri). (2p.)