

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 $A \times A \times A \times A$ 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.)**