

**Subiectul III (30 de puncte)**

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

1. Se generează în ordine crescătoare toate numerele de 4 cifre, cu cifre distincte, astfel încât diferența în valoare absolută dintre ultimele două cifre ale fiecărui număr generat este egală cu 2. Primele opt soluții generate sunt, în ordine: 1024, 1035, 1042, 1046, 1053, 1057, 1064, 1068. Care dintre următoarele numere se va genera imediat după numărul 8975? **(4p.)**
- a. 8979                      b. 9013                      c. 8957                      d. 9024

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

2. Ce se va afișa în urma executării subprogramului alăturat, la apelul `F(56);`? **(6p.)**
- ```
void F(int x)
{
    if(x)
    {
        F(x/2);
        cout << x%10; | printf("%d",x%10);
    }
}
```
3. **a)** Scrieți definiția completă a subprogramului **Cautare**, cu trei parametri, **n**, **x** și **v**, care primește prin parametrul **n** un număr natural ( $1 \leq n \leq 1000$ ), prin parametrul **x** un tablou unidimensional format din **n** componente (numere întregi de cel mult patru cifre fiecare:  $x_1, x_2, \dots, x_n$ ) memorate în ordine crescătoare și prin parametrul **v** un număr întreg de cel mult patru cifre, diferit de oricare dintre elementele tabloului unidimensional **x**. Subprogramul va căuta, în mod eficient din punct de vedere al timpului de executare, poziția pe care ar trebui inserată valoarea **v** în șirul **x** astfel încât să se obțină tot un șir ordonat crescător și returnează această poziție. **(6p.)**
- b)** Descrieți succint, în limbaj natural, metoda de rezolvare folosită, explicând în ce constă eficiența ei (3 – 4 rânduri). **(4p.)**
- c)** Fișierul text **sir.in** conține cel mult 1000 numere întregi de maximum patru cifre fiecare, numerele fiind diferite două câte două și despărțite prin câte un spațiu. Scrieți un program C/C++ care citește numerele din fișierul **sir.in** și, folosind apeluri utile ale subprogramului **Cautare**, construiește în memorie un tablou unidimensional care va conține toate numerele din fișierul **sir.in** ordonate crescător. Programul scrie în fișierul text **sir.out** șirul obținut, câte 10 elemente pe un rând, elementele de pe același rând fiind despărțite printr-un singur spațiu. **Exemplu:** dacă fișierul **sir.in** conține numerele: 7 -5 635 -456 0 8 587 -98 65 3 -8 atunci după executarea programului fișierul **sir.out** va conține:  
-456 -98 -8 -5 0 3 7 8 65 587  
635 **(10p.)**