

Subiectul III (30 de puncte)

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

1. Aplicând metoda backtracking pentru a genera toate permutările celor n elemente ale unei mulțimi, o soluție se memorează sub forma unui tablou unidimensional x_1, x_2, \dots, x_n . Dacă sunt deja generate valori pentru componentele x_1, x_2, \dots, x_{k-1} , iar pentru componenta curentă, x_k ($1 < k < n$), a fost găsită o valoare convenabilă, atunci se încearcă alegerea **(4p.)**
- a. unei noi valori pentru componenta x_{k-1} b. unei valori pentru componenta x_{k+1}
c. unei noi valori pentru componenta x_k d. unei noi valori pentru componenta x_1

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

2. Ce afișează subprogramul **F**, descris alăturat, la apelul **F(5)**? **(6p.)**
- ```
void F(int x)
{
 cout<<x; | printf("%d",x);
 if(x>=3)
 F(x-2);
}
```
3. Scrieți definiția completă a subprogramului **divizor**, cu trei parametri, prin care primește 3 numere naturale nenule cu cel mult 9 cifre fiecare și returnează numărul divizorilor comuni tuturor celor 3 numere.  
**Exemplu:** dacă numerele primite ca parametri sunt 24, 20 și 12, subprogramul returnează valoarea 3 (divizorii comuni sunt 1, 2 și 4). **(10p.)**
4. Fișierul **BAC.TXT** conține 10000 de numere naturale (dintre care cel puțin două impare) cu cel mult 9 cifre fiecare. Numerele sunt separate prin câte un spațiu.
- a) Scrieți un program **C/C++** care, utilizând un algoritm eficient din punct de vedere al spațiului de memorare și al timpului de executare, determină și afișează pe ecran penultimul număr impar din fișier precum și numărul de ordine al acestuia. **(6p.)**  
**Exemplu:** dacă fișierul conține valorile alăturate, se vor afișa **10 45 0 ... 0 49 6 7**  
numerele **49 9998** (penultimul număr impar este **49** și are **9995** de 0)
- b) Descrieți în limbaj natural metoda utilizată și explicați în ce constă eficiența ei. **(4p.)**