

Subiectul III (30 de puncte)

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

1. Utilizând metoda backtracking se generează numerele formate din câte 3 cifre distincte din mulțimea $\{1, 3, 5, 7\}$. Dacă primele trei numere generate sunt, în această ordine: 135, 137, 153 care este cel de-al patrulea număr generat? (4p.)
- a. 315 b. 173 c. 157 d. 357

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

2. Subprogramul **f** are definiția alăturată. Ce valoare are **f(3)**? Dar **f(10)**? (6p.)
- ```
int f(int x)
{
 if(x==0) return 0;
 else return f(x-1)+2;
}
```

```
int f(int x)
{
 if(x==0) return 0;
 else return f(x-1)+2;
}
```
3. Scrieți definiția completă a unui subprogram **P** cu doi parametri, care primește prin intermediul primului parametru, **n**, un număr natural nenul ( $1 \leq n \leq 100$ ) și prin intermediul celui de-al doilea parametru, **a**, un tablou unidimensional cu elementele numerotate de la 1 la **n**, numere întregi de cel mult 4 cifre fiecare. Subprogramul returnează suma tuturor numerelor **impare** aflate pe poziții **pare** din tablou.
- Exemplu:** dacă **n=6**, iar șirul **a** este format din elementele (3, 12, 7, 1, 4, 3), atunci la apel se va returna 4. (10p.)
4. Fișierul text **numere.txt** conține pe prima linie un număr natural **n** ( $0 < n < 100000$ ) iar pe a doua linie **n** cifre, separate prin câte un spațiu.
- a) Scrieți un program C/C++ care determină în mod eficient, din punct de vedere al timpului de executare, cel mai mare număr ce se poate forma cu toate cifrele conținute de a doua linie a fișierului **numere.txt**. Numărul determinat se va afișa pe ecran.
- Exemplu:** dacă fișierul **numere.txt** are următorul conținut:
- ```
7
2 5 3 1 5 8 9
```
- atunci pe ecran se va afișa: 9855321. (6p.)
- b) Descrieți succint, în limbaj natural, metoda de rezolvare folosită, explicând în ce constă eficiența ei (3 – 4 rânduri). (4p.)