

Subiectul III (30 de puncte)

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

1. Se generează, utilizând metoda backtracking, toate modalitățile de așezare a numerelor naturale de la 1 la 5 astfel încât oricare 2 numere de aceeași paritate să nu se afle pe poziții alăturate. Dacă primele 3 soluții generate sunt, în ordine: (1,2,3,4,5) și (1,2,5,4,3), (1,4,3,5,2) care este prima soluție generată ce începe cu 3? **(4p.)**
- a. (3, 2, 1, 4, 5) b. (3, 2, 5, 4, 1) c. (3, 4, 1, 2, 5) d. (3, 4, 5, 2, 1)

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

2. Subprogramul recursiv alăturat este definit incomplet. Care este expresia cu care se pot înlocui punctele de suspensie astfel încât **f(40,16)** să fie egal cu 8. **(6p.)**
- ```
int f(unsigned int a, unsigned int b)
{ if (...)
 return a;
 else
 if (a>b) return f(a-b,b);
 else return f(a,b-a);
}
```

3. Subprogram **sfx** primește prin singurul său parametru, **x**, un număr natural din intervalul [100,2000000000] și returnează valoarea 1 dacă ultimele trei cifre ale numărului sunt în ordine strict descrescătoare sau valoarea 0 în caz contrar.

**Exemplu:** la apelul **sfx(24973)** se va returna valoarea 1.

a) Scrieți definiția completă a subprogramului **sfx**. **(10p.)**

b) Fișierul text **date.in** conține cel mult 10000 de numere naturale de exact 6 cifre fiecare, separate prin câte un spațiu. Scrieți un program C/C++ care citește toate numerele din fișier, determină și afișează pe ecran câte dintre aceste numere au toate cifrele în ordine strict descrescătoare. Programul va folosi apeluri utile ale subprogramului **sfx**. Se va utiliza un algoritm eficient din punctul de vedere al memoriei utilizate. **(6p.)**

**Exemplu:** dacă fișierul **date.in** conține

236543

865210

976532

Pe ecran se afișează:

2

c) Descrieți succint, în limbaj natural, metoda de rezolvare folosită, explicând în ce constă eficiența ei (3 – 4 rânduri). **(4p.)**