

**Subiectul III (30 de puncte)**

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

1. Utilizând metoda backtracking se generează în ordine lexicografică toate anagramele cuvântului **caiet** (cuvinte formate din aceleași litere, eventual în altă ordine). Care este a șasea soluție? (4p.)

a. <b>catei</b> c. <b>actei</b>		b. <b>actie</b> d. <b>catie</b>
------------------------------------	--	------------------------------------

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

2. Se consideră subprogramul **f**,  
definit alăturat. Ce valoare are **f(97,2)**? Dar **f(175,2)**? (6p.)

```
int f(int n, int x)
{
    if (n<=1) return 0;
    else if (x<=n/2)
        if n%x=0 return 0;
        else return f(n,x+1);
    else return 1;
}
```

3. Se consideră subprogramul **pal** care primește care primește prin intermediul primului parametru **a** un număr natural, cu **minimum** 2 cifre și **maximum** 8 cifre, și returnează prin intermediul celui de-al doilea parametru **b** cel mai apropiat număr palindrom de **a**. În cazul în care există 2 astfel de numere, subprogramul va returna numărul mai mic. Un număr natural **x** este palindrom dacă este egal cu numărul obținut prin scrierea cifrelor lui **x** în ordine inversă.

**Exemplu:** **pal(16)** va returna valoarea 11; **pal(126)** va returna 121; **pal(33)** va returna 33

a) Scrieți definiția completă a subprogramului **pal**. (4p.)

b) Fișierul text **date.in** conține pe prima linie un număr natural nenul **n** ( $n \leq 100$ ) iar pe a doua linie **n** numere naturale nenule, separate prin câte un spațiu, fiecare număr având **minimum** 2 cifre și **maximum** 8 cifre. Scrieți un program C/C++ care citește toate numerele din fișierul text **date.in** și afișează pe ecran despărțite printr-un spațiu, pentru fiecare dintre cele **n** numere cel mai apropiat număr palindrom, folosind apeluri utile ale subprogramului **pal**.

**Exemplu:** dacă fișierul **date.in** are conținutul alăturat, atunci  
se vor afișa numerele: 11 1771 333 191 (6p.)

```
4
16 1775 333 190
```

4. Scrieți un program C/C++ care citește de la tastatură un număr natural nenul **n** ( $n \leq 100$ ) și apoi cele **n** numere naturale nenule, de maximum 4 cifre, reprezentând elementele unui tabou unidimensional **v** (cu indici de la 1 la **n**) și afișează câte dintre elementele **v[i]** ( $2 \leq i \leq n-1$ ) sunt egale cu suma celor două elemente vecine. În cazul în care nu există niciun astfel de element în tabloul **v** se va afișa valoarea 0.

**Exemplu:** dacă **n=4** și tabloul unidimensional  
**v** are conținutul alăturat atunci se va afișa  
valoarea 2 (deoarece  $25=10+15$ ,  
 $45=15+30$ ) (10p.)

```
10 25 15 45 30 2 1
```