

## Problema 1 – CIFRU

Autor: prof. Alin Burța

### Descrierea soluției

Vom utiliza tabloul unidimensional **Apar**, cu 10 elemente, care memorează de câte ori se repetă fiecare cifră în cifrul inițial, deci:

$Apar[i]$  = numărul de apariții al cifrei  $i$  în cifrul inițial, unde  $i=0, 1, \dots, 9$

Valorile tabloului **Apar** pot fi calculate pe măsură ce citim datele de intrare, nefiind nevoie de memorarea celor  $N$  numere ce formează cifrul. Este suficient să aflăm câte discuri sunt poziționate inițial pe cifra 0, câte pe cifra 1, ..., câte pe cifra 9, iar apoi să calculăm numărul minim de mutări necesare pentru a poziționa cifrul pe o anumită cifră. Astfel:

- Calculăm pe rând, pentru fiecare cifră  $i=0,1,\dots,9$  care apare măcar o dată în cifru, numărul de mutări necesare pentru ca toate discurile cifrului să fie setate pe valoarea  $i$ , astfel:

**(C++)**

```
for (j=0; j<=9; j++)  
    if (Apar[j] && j!=i)  
        Nr += abs(j-i) <= 10 - abs(j-i) ? Apar[j]*abs(j-i) : Apar[j]*(10 - abs(j-i));
```

**(PASCAL)**

```
for j:=0 to 9 do  
    if (Apar[j]>0) and (j<>i) then  
        if abs(j-i) <= 10 - abs(j-i) then Nr := Nr+ Apar[j]*abs(j-i)  
        else Nr := Nr + Apar[j]*(10 - abs(j-i));
```

- Se ține cont de faptul că fiecare disc poate fi mutat în două direcții deci, dacă cifrul este poziționat pe cifra  $j$ , pentru a-l muta pe cifra  $i$  putem face  $abs(j-i)$  mutări în sus sau  $10 - abs(j-i)$  mutări în jos. La fiecare pas considerăm cea mai mică valoare dintre cele două.
- Când găsim un număr total de mutări mai mic actualizăm minimul curent și reținem cifra pentru care se obține acesta. Atunci când găsim un număr total de mutări egal cu minimul curent incrementăm numărul de soluții de valoare minimă.