**Problema birot - Descrierea soluției**

Autor: *prof. Ionel-Vasile Piț-Rada*

*Colegiul Național “Traian”*

**Soluția 1 – 30 puncte**

Backtracking.

La fiecare nivel **k**, se ajunge cu rotițele în pozițiile **p1** și respectiv **p2** și costul cost. Caracterul **t[k]** poate fi obținut prin deplasarea rotiței **1** la poziția **r1** cu costul **c1** sau deplasând rotița 2 la poziția **r2** cu costul **c2**. Astfel din starea (**k**,**p1**,**p2**,**cost**) se vor încerca, pe rând, trecerile în stările (**k+1**,**r1**,**p2**,**cost+c1**) și respectiv (**k+1**,**p1**,**r2**,**cost+c2**).

Complexitatea timp este **O(2n)**

Complexitatea spațiu este **O(n+m)**

**Soluția 2 – 80-85 puncte**

Programare dinamică.

Lungimea șirului care trebuie construit este **n** , iar lungimile șirurilor rotițelor sunt egale cu **m**. Se păstrează pentru fiecare pereche de poziții ale celor două rotițe costul optim până la momentul **k**

**cost[k-1][i][j]** și pe baza lor se calculează **cost[k][i][j]**.

Caracterul **t[k]** se află pe rotița 1 la poziția **r1** și pe rotița 2 la poziția **r2** . Deplasarea rotiței 1 de la poziția **i** la poziția **r1** costă **c1(i,r1)** , iar pentru deplasarea rotiței 2 de la j la poziția **r2** costul este **c2(j,r2)**. Astfel **cost[k][r1][j]=cost[k-1][i][j]+c1(i,r1)**

și respectiv **cost[k][i][r2]=cost[k-1][i][j]+c2(j,r2)**

Se observă că se poate renunța la indicele **k** , lucrându-se pe două tablouri bidimensionale.

Complexitate timp **O(n·m2)**

Complexitate spatiu **O(n+m2)**

**Soluția 3 – 100 puncte**

Se observă că la soluția 2 este suficientă utilizarea doar a două tablouri unidimensionale , unul pentru costurile specifice utilizării rotiței 1 , iar al doilea pentru costurile utilizării rotiței 2. Observația se bazează pe faptul că după construirea caracterului **t[k]** se știu pozițiile unde se află rotița1 , respectiv rotița 2.

Complexitate timp **O(n·m)**

Complexitate spatiu **O(n+m)**