



Problema Premii

Clasa a X-a

Autor Lect. Dr. Păcșas Csaba-György, Universitatea Babeș-Bolyai, Cluj-Napoca

Abordarea 1

Pentru valori mici ale lui n putem genera toate posibilitățile de partiționare a premiilor în trei mulțimi disjuncte: primul concurent, al doilea concurent, respectiv restul concurenților. Dintre soluțiile în care suma primei partiții este egală suma celei de-a doua partiții, alegem cea cu suma maximă. Algoritmul are timp de rulare de ordinul $O(3^n)$ și poate obține între 15 și 30 de puncte în funcție de implementare și optimizările folosite.

Această abordare se poate îmbunătăți, dacă observăm, că este de ajuns să generăm a treia partiție, care reprezintă premiile primite de ceilalți concurenți. Restul premiilor trebuie împărțit în mod egal între cei doi concurenți, ceea ce se poate realiza în timp pseudopolinomial aplicând metoda programării dinamice de tip rucsac. Dacă notăm cu S suma valorilor celor n premii, algoritmul rulează în timp $O(2^n \cdot n \cdot S)$ și obține între 30 și 75 de puncte în funcție de implementare și optimizările folosite.

Abordarea 2

Pentru 50% din punctaj putem aplica metoda programării dinamice în felul următor. Fie p_{i,s_1,s_2} adevărat, dacă folosind primele i premii, putem alocă premii în valoare s_1 primului concurent și premii în valoare s_2 celui de-al doilea concurent. Evident, $p_{0,0,0}$ este adevărat și soluția o vom găsi pe poziția $p_{n,m,m}$ cu valoare adevărată, pentru care m este maxim. Relația de recurență se deduce ușor:

$$p_{i,s_1,s_2} = p_{i-1,s_1,s_2} \text{ SAU } p_{i-1,s_1-a_i,s_2} \text{ SAU } p_{i-1,s_1,s_2-a_i}$$

unde a_i este valoarea premiului cu numărul de ordine i .

Dat fiind, că m este cel mult egal cu $S/2$, algoritmul are complexitatea timp $O(nS^2)$. Fiindcă p_i depinde doar de p_{i-1} , algoritmul se poate implementa folosind $O(S^2)$ memorie suplimentară.

Abordarea 3

Pentru obținerea punctajului maxim, observăm că putem scăpa de unul dintre parametrii folosiți în abordarea precedentă. Fie $sm_{i,d}$ suma maximă ce poate fi alocată celor doi concurenți folosind doar primele i premii, în așa fel, încât diferența dintre valoarea premiilor alocate primului concurent și valoarea premiilor alocate concurentului al doilea să fie d . Relație de recurență va fi:

$$sm_{i,d} = \max(sm_{i-1,d}, sm_{i-1,d-a_i} + a_i, sm_{i-1,d+a_i} + a_i)$$

Soluția se va găsi pe poziția $sm_{n,0}$.

Fiindcă d poate lua valori din intervalul $[-S, S]$, care de fapt se poate reduce la $[-S/2, S/2]$, complexitatea timp a algoritmului este $O(nS)$.

Menționăm că pe acest tip de problemă, diferite euristici determinați și nedeterminați se comportă foarte bine în practică și pot obține un punctaj între 40 și 90 de puncte.