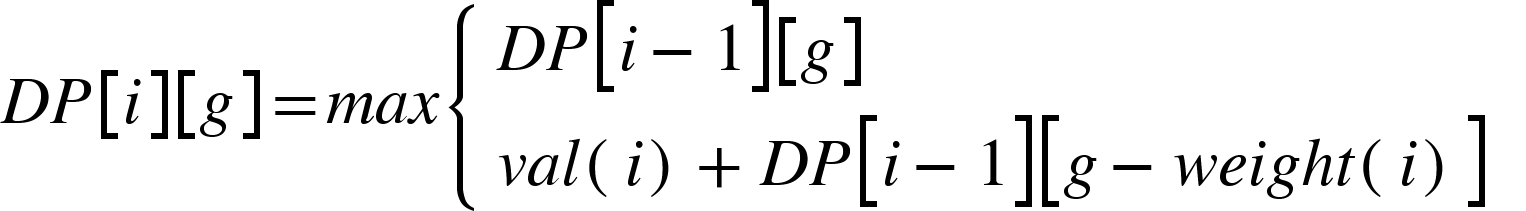
Algoritmul pseudopolinomial pt 1/0 knapsack problem:

DP[i][g] - profitul maxim ce poate fi obtinut alegand dintre primele *i* obiecte, fara a depasi greutatea totala *g*

Cum calculam DP?

DP[0][g]=DP[i][0]=0



Se citesc din fisier: n, W -capacitatea rucsacului, val si w - 2 vectori de *n* elemente reprezentand vectorul de valori si de greutati pt cele *n* obiecte. Sa Se afiseze suma maxima ce poate fi obtinuta din incarcarea unor obiecte in rucsac. Sa se afiseze (cumva) si obiectele care au fost selectate

Input

W= 50

val: 60, 100, 120

w: 10, 20, 30

OUTPUT: S=220; Obiectele [0,1,1]

Problema 2:

Joc:

Avem un joc in 2 persoane. Cei doi jucatori au un fata lor un sir S de numere pozitive. Jocul este alcatuit dintr-o succesiune de mutari astfel: jucatorul aflat la mutare trebuie sa aleaga unul dintre capetele sirului, rupe din sir valoarea din acel capat, si-o insuseste si este adaugata punctajului sau. Jocul se termina cand S devine vid. Castigatorul este jucatorul cu punctaj maxim.

1. Daca sirul S este de lungime para. Poate jucatorul 1 sa isi asigure mereu cel putin o remiza? Se calculeaza suma de pozitiile pare si impare.
2. Indiferent de lungimea lui S, cum se poate calcula mutarea optima a unui jucator / punctajul maxim garantat ce poate fi obtinut?

[1…….n] - configuratia de start -> vreau sa aflu mutarea optima

[i……...j] -configuratia tablei intr-un moment oarecare > mutarea optima

T]i][j] = punctajul maxim ce poate fi obtinut de catre jucatorul aflat la mutare si care are tabla S[i….j]

T[1][n] - raspunsul cautat de noi

T[i][i]=S[i]

T[i][i+1]=max(S[i],S[i+1])

