1. 1/0 Knapsack Problem

Avem *n* obiecte dintre care putem alege si un rucsac de capacitate W. Fiecare obiect este caracterizat printr-o valoare si greutate (numere intregi pozitive). Trebuie sa alegem in mod optim acele obiecte care maximizeaza profitul total, in același timp nedepasind greutatea totala W.

OBS: fiecare obiect ori este integral selectat ori este neselectat.

Input:  
n - numarul de obiecte

W - capacitatea rucsacului

w[] - greutatile obiectelor

val[]-valorile al obiectelor.  
  
Output:  
S - profitul maxim obtinut;

x[] - un vector de *n* elemente de 1 si 0 cu semnificatia ca daca x[i]=1 atunci obiectul *i* a fost selectat.

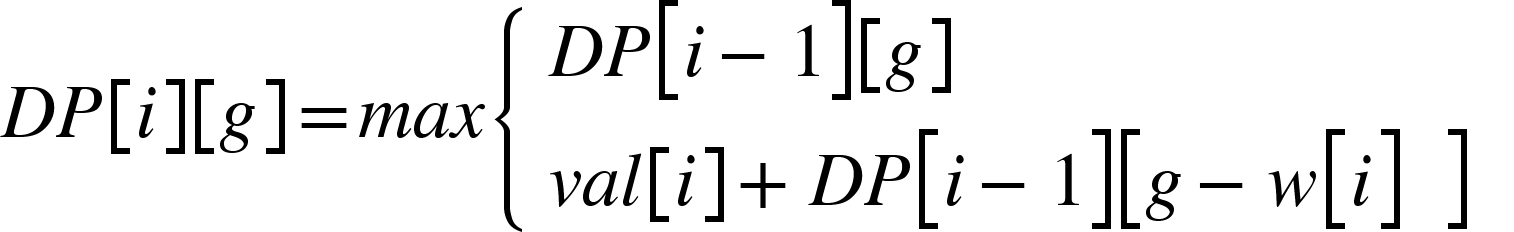
Ex: W=50; w={10,20,30} val={60,100,120}

Output: S=220 x: [0,1,1]

-----------------------------------------

Rezolvare:  
DP[i][g]=valoae maxima obtinuta alegand dintre primele *i* obiecte fara sa depasim greutatea *g*

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

  
  
2) Joc   
Fie urmatoarele reguli de joc:

2 jucatori au in fata un sir de numere S. La fiecare mutare jucatorul curent alege din S unul dintre cele doua capete, il “rupe” din S, iar valoarea numerica o adauga la punctajul obtinut. Apoi celălalt jucator va trebui sa mute. Jocul se termina atunci cand S este nul iar castigatatul este jucatorul care a strans un punctaj maxim.

1. Daca S este de lungime para, exista o tactica pentru Player 1 astfel incat sa obtina mereu macar o remiza? OBS: se poate lega de paritate.
2. Cum poate primul jucator sa obtina un punctaj maxim indiferent de numarul de elemente din S?

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | ... | i | i+1 | i+2 | ... | j-2 | j-1 | j | ... | n |

T[i][j] = punctajul maxim ce poate fi (garantat) obtinut de catre jucatorul aflat la mutare si avand in fata sirul S[i…..j]  
  
Solutia pt problema T[1][n]

Cum se calculeaza T?  
T[i][i]=S[i]

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

