DESIGN & ANALYSIS OF ALGORITHMS
(Prof. Parminder Kaur Wadhwa)

Using Backtracking Technique to solve

Sum of Subsets

Suppose we are given n distinct positive numbers (usually called weights) and we desire to find all combinations of these numbers whose sums are m.
This is called as the sum of subsets problem.

For example; if $n = 4$
Required sum $= m = 31$

Given weights are:-

$$(w_1, w_2, w_3, w_4)$$
$$= (11, 13, 24, 7)$$

Then, the desired subsets whose sum is $m = 31$ are:-

$$(11, 13, 7) \text{ and } (24, 7)$$

The number of nodes in solution space by using backtracking for sum of subsets is 2^n.

The worst-case time for the backtracking algorithm solving sum of subsets problem is $O(p(n)\,2^n)$, (where $p(n)$ is polynomial in n).

Considering a backtracking solution involving the fixed-tuple-size strategy.

In this case the element x_i of the solution vector is either one or zero depending on whether the weight w_i is included or not.

The solution for:- $n = 4$, $m = 31$

$$\begin{pmatrix} w_1, & w_2, & w_3, & w_4 \end{pmatrix}$$
$$\begin{pmatrix} 11, & 13, & 24, & 7 \end{pmatrix}$$

are:- $\begin{pmatrix} 1 & 1 & 0 & 1 \end{pmatrix}$, $\begin{pmatrix} 0 & 0 & 1 & 1 \end{pmatrix}$ fixed-tuple size of the solution

For a node at level i the left child corresponds to $x_i = 1$ and the right to $x_i = 0$.

a node

(means x_i is included) $x_i = 1$ — LEFT SUBTREE

$x_i = 0$ (means x_i is not included) — RIGHT SUBTREE

⋮

The bounding function for sum of subsets problem is :-

$$B_k(x_1, \cdots, x_k) = \text{true}$$

iff

$$\sum_{i=1}^{k} w_i x_i + \sum_{i=k+1}^{n} w_i \geq m$$

and

$$\sum_{i=1}^{k} w_i x_i + w_{k+1} \leq m$$

The first condition is :- that the sum computed till k i.e. $\sum_{i=1}^{k} w_i x_i$ when is added to the sum of remaining weights i.e. from $(k+1)$ ~~f~~ onwards i.e. $\sum_{i=k+1}^{n} w_i$

should be greater than or equal to m.

The second condition is :- that when new next weight i.e. w_{k+1} is added to the already computed weights i.e. $\sum_{i=1}^{k} w_i x_i$ should be less than or equal to m. It means it should not exceed m because m is the required sum.

When both first & second conditions are TRUE, then that {solution / state} is considered otherwise not.

Sum of Subsets -

for example :- $m = 31$ (required sum)

Elements :- x_1 x_2 x_3 x_4

11 13 24 7

$$\begin{array}{r} 11 \\ +13 \\ 24 \\ +07 \\ \hline 55 \end{array}$$