Unit - V

Backtracking & Branch & Bound.

Backtracking.

To construct a solution one component at a time and evaluate such partially constructed candidate as follows. if a partially constructed solution can be developed further without violating the problem's constraints.

State space tree

root represents an initial state before the search for a solution begins. The nodes of the first level in the tree represent the choices made for the first component of a solution

Promising Node.

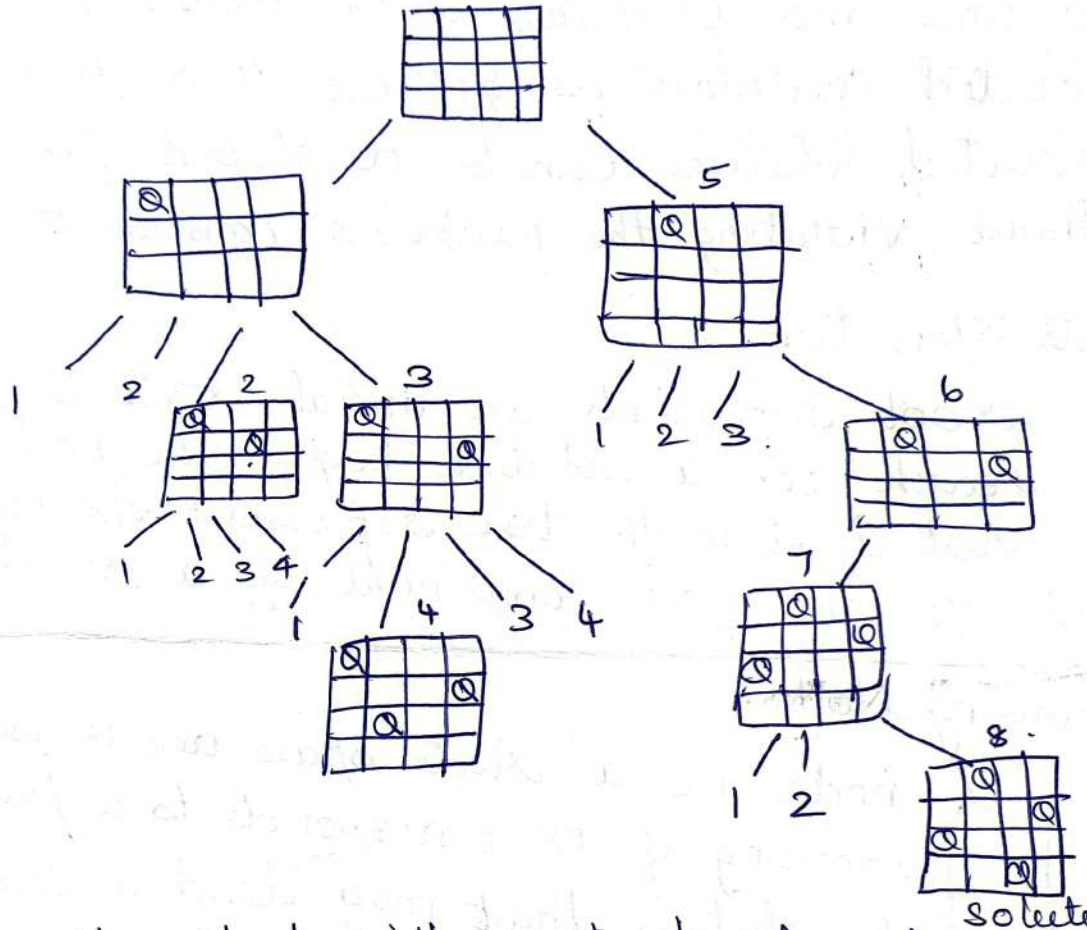
A node in a state space tree is said to be promising if it corresponds to a partially constructed solution that may lead a complete solution. otherwise it is called non-promising.

n - Queens problem.

Algorithm Backtract(x[1--i])
// input x[1--i] specifies first i promising component of a solution
// output All the tuples representing the pbm soln.
if x[1--i] is a solution write x[1--i]
else
for each element x ∈ S_{i+1} Consistent with x[1--i] and the constraints do.
x[i+1] ← x
Backtrack [x[1--i+1]]

n - Queens Problems.

To place n queens on an n-by-n che
so that no two queens attack each other by
in the same row or in the same column or o
the same diagonal.



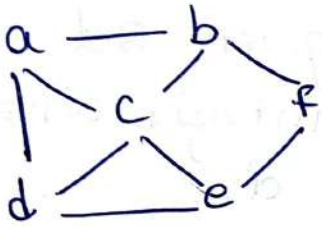
solution

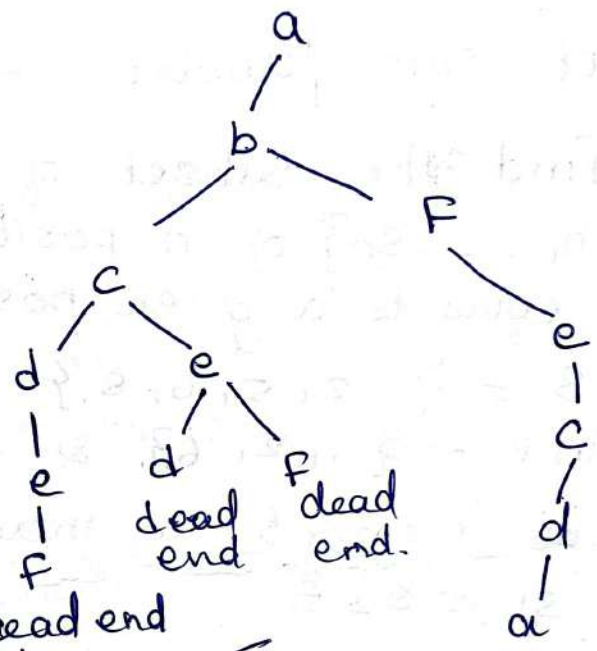
We start with empty board and place the queen in the first possible position of its row, which is in column 1 & 2 in the first acceptable position for it, which is square (2, 3) the square in row 2 and column 3. This proves to be a dead end because there is no acceptable position for queen 3. So the algorithm backtracks and puts queen 2 in the next possible position at (2, 4). The queen 3 is placed at (3, 2) which proves to be another dead end. The algorithm then backtracks to all the way to queen 1 and moves it to (1, 2) Queen 2 goes to (2, 4) queen 3 to (3, 1) and queen 4 to (4, 3) which is the solution to the problem.

Hamiltanion circuit problem.

Hamiltanion path in an undirected graph is a path that visits each vertex exactly once. A Hamiltanion cycle is a Hamiltanion path such that there is an edge from the last vertex to the first vertex of the hamiltanion path.

Example.

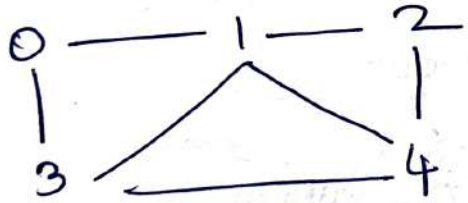


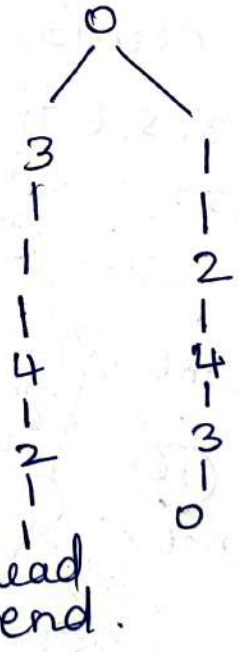


Solution.

a – b – f – e – c – d – a

Ex: 2

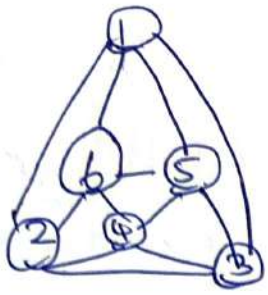




ex: 3

Solution

Ex - 3



Soln. 1 – 3 – 5 – 4 – 6 – 2 – 1 soln.
1 – 6 – 4 – 5 – 3 – 1 dead end.
1 – 2 – 6 – 4 – 5 – 3 – 1 – soln.

Subset sum problem :-

find the subset of a given set $S = \{s_1 \dots s_n\}$ of n positive integer whose sum is equal to a given positive d.

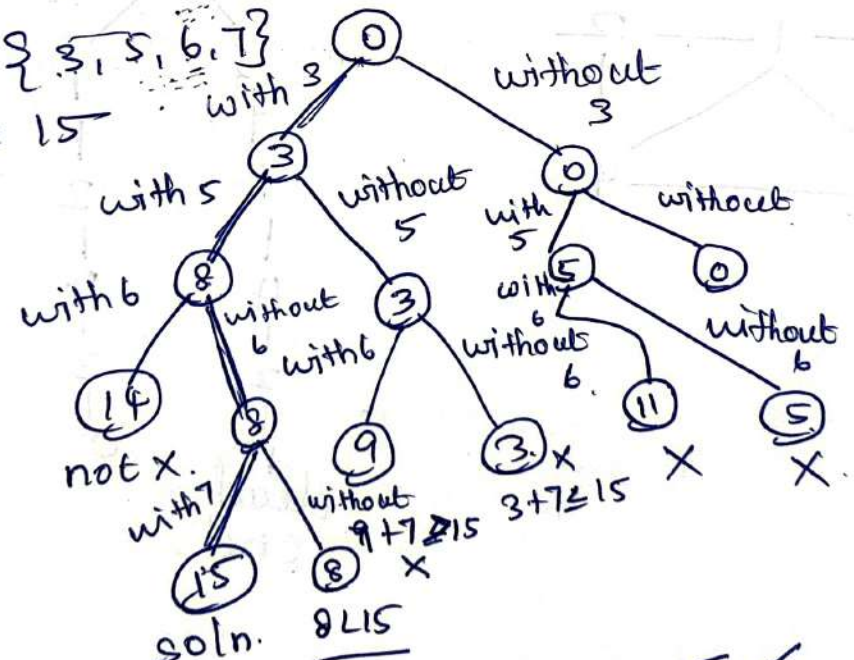
ex $S = \{1, 2, 5, 6, 8\}$ $d = 9$.
soln 1 = {1, 2, 6}. & {1, 8}

Set of elements in increasing order.
$s_1 \leq s_2 \leq \dots s_n$.

* root represents the starting point with no decision about the given element made.

i.e

$S = \{3, 5, 6, 7\}$
$d = 15$
$3+5+7$



Nitish

solution $3 + 5 + 7 = 15$ ✓

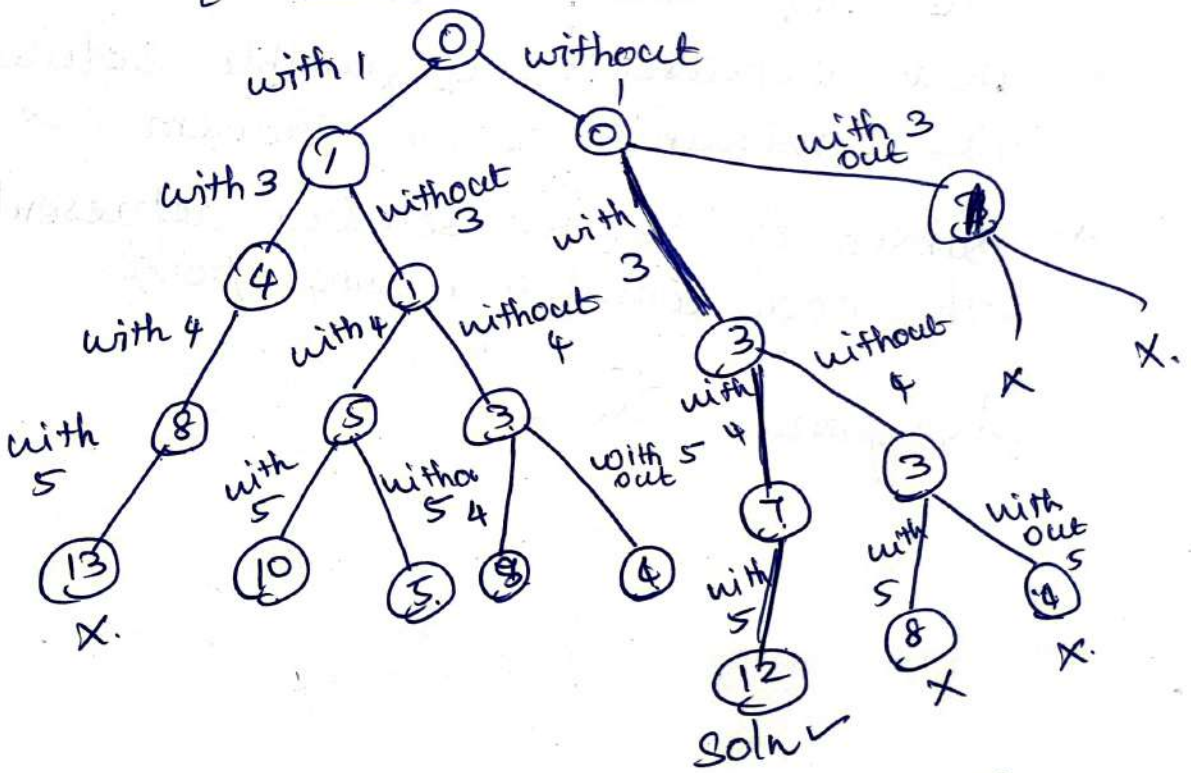
root of the tree represents the starting i.e. with no decisions about the given elements & left & right represent inclusion & exclusion of s1 & s2. Thus a path from root to a node on ith level of the tree indicates which of the first i numbers have been included in the subsets.

We record the value of s' the sum of these numbers, in the node. if s' is equal to d, we have a solution to the problem. or if all the solution need to be found, continue by backtracking to the node's Parent. if s' not equal to d.

$s' + s_{i+1} > d$ (sum s' is too long

$s' + \sum_{j=i+1}^{n} s_j < d$ (sum s' is too small.

ex: 2 $S = \{1, 3, 4, 5\}$ $d = 12$



$3+4+5 = 12$

State Space tree