* Laboratory Practice II (Artificial Intelligence) - Group B - Experiment No. - 4

Name:- Kaustubh Shrikant Kabra.
Class:- Third Year Engineering.
Div:- A
ERP Number:- 38
Department:- Computer Department
College:- AISSMS's IOIT.

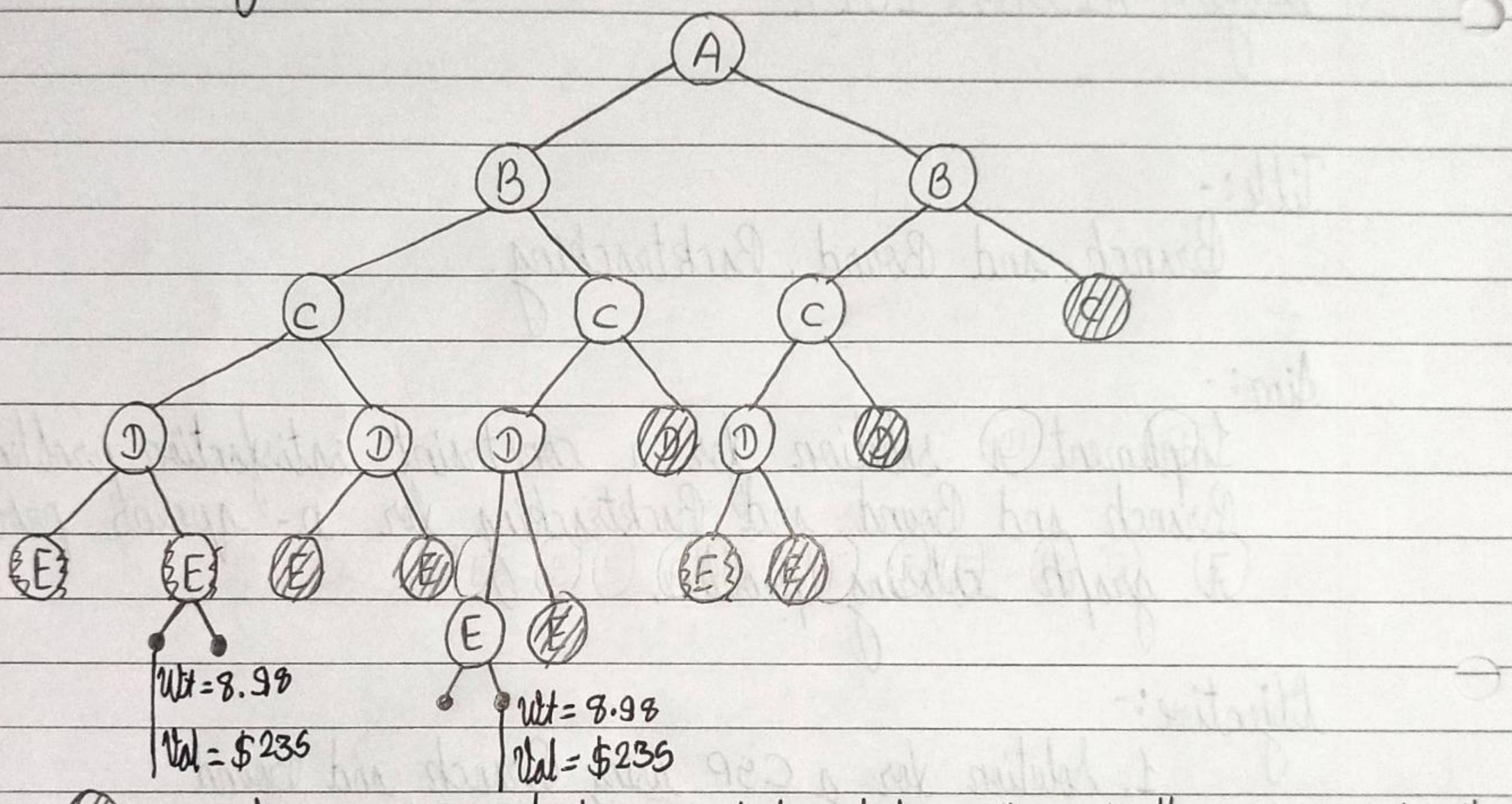
Title:-
Branch and Bound, Backtracking.

Aim:-
Implement a solution for a constraint satisfaction problem using Branch and Bound and Backtracking for n-queens problem / or a graph coloring problem.

Objective:-
1. Solution for a CSP using Branch and Bound.
2. Backtracking for n-queens problem.

Theory:-
Constraint Satisfaction Problem (CSP) are mathematical questions defined as a set of objects whose state must satisfy a number of constraints or limitations. CSP's represent the entities in a problem as a homogenous collection of finite constraints over variables, which is solved by constraint satisfaction methods.

Branch and Bound is an algorithm design paradigm which is generally used for solving combinational optimization problems. These problems are typically exponential in terms of time complexity and may require exponential in terms of time complexity and may require exploring all possible permutations in worst case. The branch and bound algorithm technique solve these problem relatively quickly.



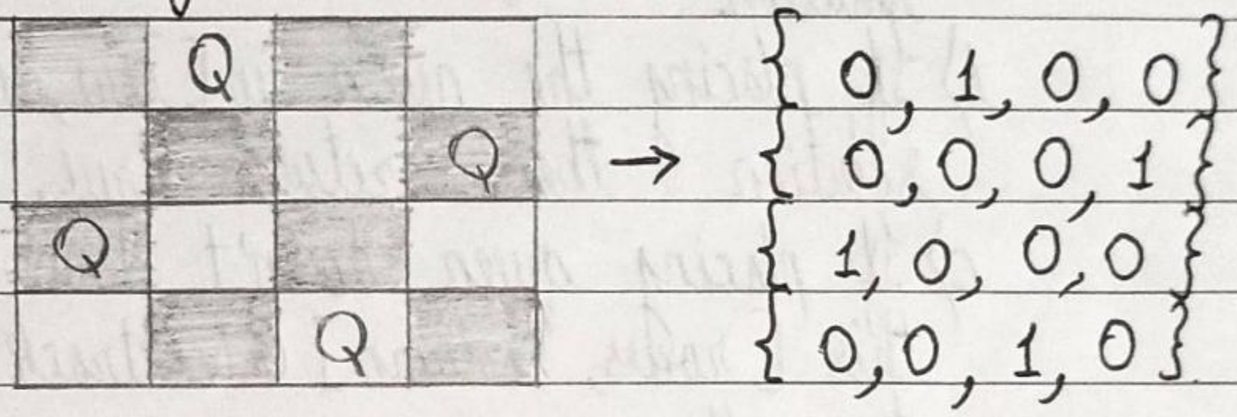
⊘ → nodes are ignored because best solution through them is worst than the current solution.

⊖ → nodes are ignored because of infeasible solution through them.

Backtracking -

Backtracking is an algorithm for solving problems recursively by trying to build a solution incrementally, one piece at a time, removing those solution that fail to satisfy the constraints of the problem at any point of time.

The N- Queen is the problem of placing N chess queens on an N×N chessboard so that no two queens attack each other. For example following is a solution for 4 queens problem.



Algorithm :-

1. Branch and Bound Algorithm -

```
Data : Input cost matrix M[][]
Result : Assign of jobs to each worker according to optimal cost.
Function Min Cost (M[][])
    While true do
    if E is a leaf node then
        print()
        return;
    end
    for each child S of E do
        Add (S);
        S → parent = E;
    end
end
```

2. Backtracking Algorithm:

Start in the leftmost column
If all queens are placed return true.
Try all rows in the current column.
Do following for every tried row.
a) If the queen can be replaced safely in this row then mark this [row, column] as part of solution and recursively check if placing queen leads to solution
b) If placing the queen in [row, column] leads to a solution then return true.
c) If placing queen doesn't lead to a solution then unmark this [rows, column] (backtrack) and go to step(a) to try other rows.

If all rows have been tried and nothing worked, return false to trigger backtracking.

Conclusion:-

Thus we solved CSP problem using Branch and Bound also n-queens problem using backtracking.