

## Q. Design State Space Search for Water Jug Problem and Implement same problem in prolog.

Problem: There are two jugs of volume A litre and B litre. Neither has any measuring mark on it. There is a pump that can be used to fill the jugs with water. How can you get exactly of water into the Assuming that we have unlimited supply of water.

For e.g., one having the capacity to hold 3 gallons of water and the other has the capacity to hold 4 gallons of water. There is no other measuring equipment available and the jugs also do not have any kind of marking on them. So, the agent's task here is to fill the 4-gallon jug with 2 gallons of water by using only these two jugs and no other material. Initially, both our jugs are empty.

```
solve(X,Y) :- X>4, Y<3, write('Jug A overflowed\n').
solve(X,Y) :- X<4, Y>3, write('Jug B overflowed\n').
solve(X,Y) :- X>4, Y>3, write('Both jugs overflowed\n').
```

```
solve(X,Y) :- X==2, Y==0, write('Goal acheived\n').
```

```
solve(X,Y) :-
X==0, Y==0, write('Jug A: 0, Jug B: 3 -> Filled jug B\n'), solve(0,3);
X==0, Y==3, write('Jug A: 3, Jug B: 0 -> Poured water from jug B to jug A\n'), solve(3,0);
X==3, Y==0, write('Jug A: 3, Jug B: 3 -> Filled jug B\n'), solve(3,3);
X==3, Y==3, write('Jug A: 4, Jug B: 2 -> Poured water from jug B to jug A\n'), solve(4,2);
X==4, Y==2, write('Jug A: 0, Jug B: 2 -> Emptied jug A\n'), solve(0,2);
X==0, Y==2, write('Jug A: 2, Jug B: 0 -> Poured water from jug B to jug A\n'), solve(2,0);
X==4, Y==0, write('Jug A: 1, Jug B: 3 -> Poured water in Jug B from Jug A\n'), solve(1,3);
X==1, Y==3, write('Jug A: 1, Jug B: 0 -> Emptied Jug B\n'), solve(1,0);
X==1, Y==0, write('Jug A: 0, Jug B: 1 -> Poured water in Jug B from Jug A\n'), solve(0,1);
X==0, Y==1, write('Jug A: 4, Jug B: 1 -> Filled Jug A\n'), solve(4,1);
X==4, Y==1, write('Jug A: 2, Jug B: 3 -> Poured water in Jug B from Jug A\n'), solve(2,3);
X==2, Y==3, write('Jug A: 2, Jug B: 0 -> Emptied Jug B\n'), solve(2,0).
```

```
solve(X,Y):-
X==2, Y>0, write('Jug A: 2, Jug B: 0 -> Emptied Jug B\n'), solve(2,0);
X==1, Y==1, write('Jug A: 2, Jug B: 0 -> Poured water in Jug A from Jug B\n'), solve(2,0);
X==2, Y>0, write('Jug A: 2, Jug B: 0 -> Emptied Jug B\n'), solve(2,0);
X>0, Y==2, write('Jug A: 0, Jug B: 2 -> Emptied Jug A\n'), solve(0,2);
X>0, Y==3, write('Jug A: 0, Jug B: 3 -> Emptied Jug A\n'), solve(0,3);
X>0, Y==1, write('Jug A: 4, Jug B: 1 -> Filled Jug A\n'), solve(4,1).
```

```
:- write('Enter initial state (X,Y)\n'), write('\nX: '), read(X), write('\nY: '), read(Y), write('Jug A: '), write(X), write(', Jug B: '), write(Y), write('-> Initial state\n'), solve(X,Y).
```

```
maxmax@madmax:~/Desktop/u19cs019_sem6/Artificial_Intelligence/lab1$ swipl -s program.pl
Enter initial state (X,Y)

X: 0.

Y: |: 0.
Jug A: 0, Jug B: 0 -> Initial state
Jug A: 0, Jug B: 3 -> Filled jug B
Jug A: 3, Jug B: 0 -> Poured water from jug B to jug A
Jug A: 3, Jug B: 3 -> Filled jug B
Jug A: 4, Jug B: 2 -> Poured water from jug B to jug A
Jug A: 0, Jug B: 2 -> Emptied jug A
Jug A: 2, Jug B: 0 -> Poured water from jug B to jug A
Goal acheived
```