# Artificial Intelligence (CS308)

## Assignment - 6

## U19CS012

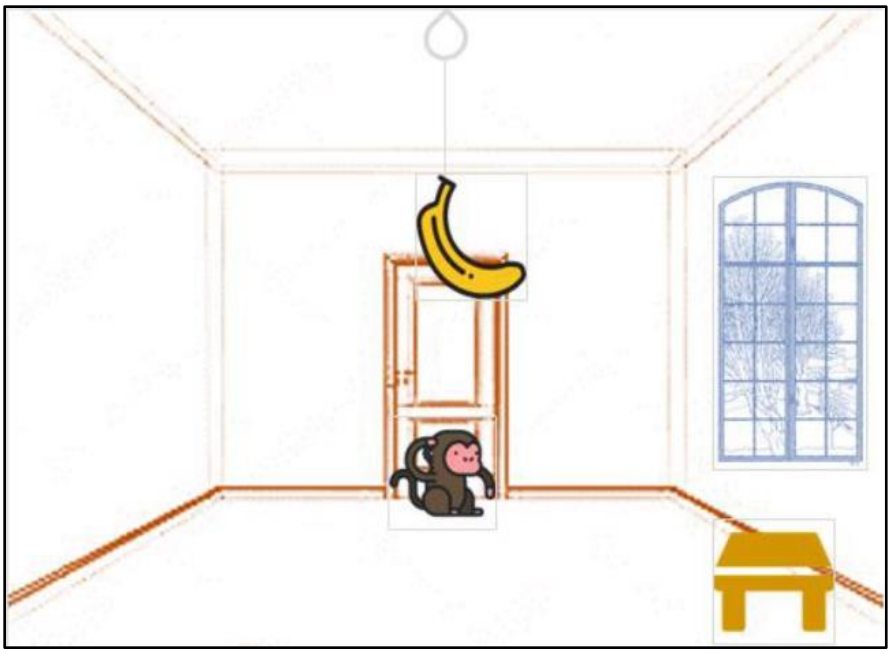
**Monkey Banana Problem**

- There is a **monkey** at the door into a room.
- In the middle of the room a **banana** is hanging from the **ceiling**.
- The monkey is **hungry** and wants to get the banana, but he cannot stretch high enough from the floor.
- At the window of the room there is a box the monkey may use.

The monkey can perform the following **actions**:

- ✓ walk on the floor
- ✓ climb box
- ✓ push box
- ✓ grasp the banana (if standing on the box directly under the banana.)

**Can the Monkey get the Banana?**



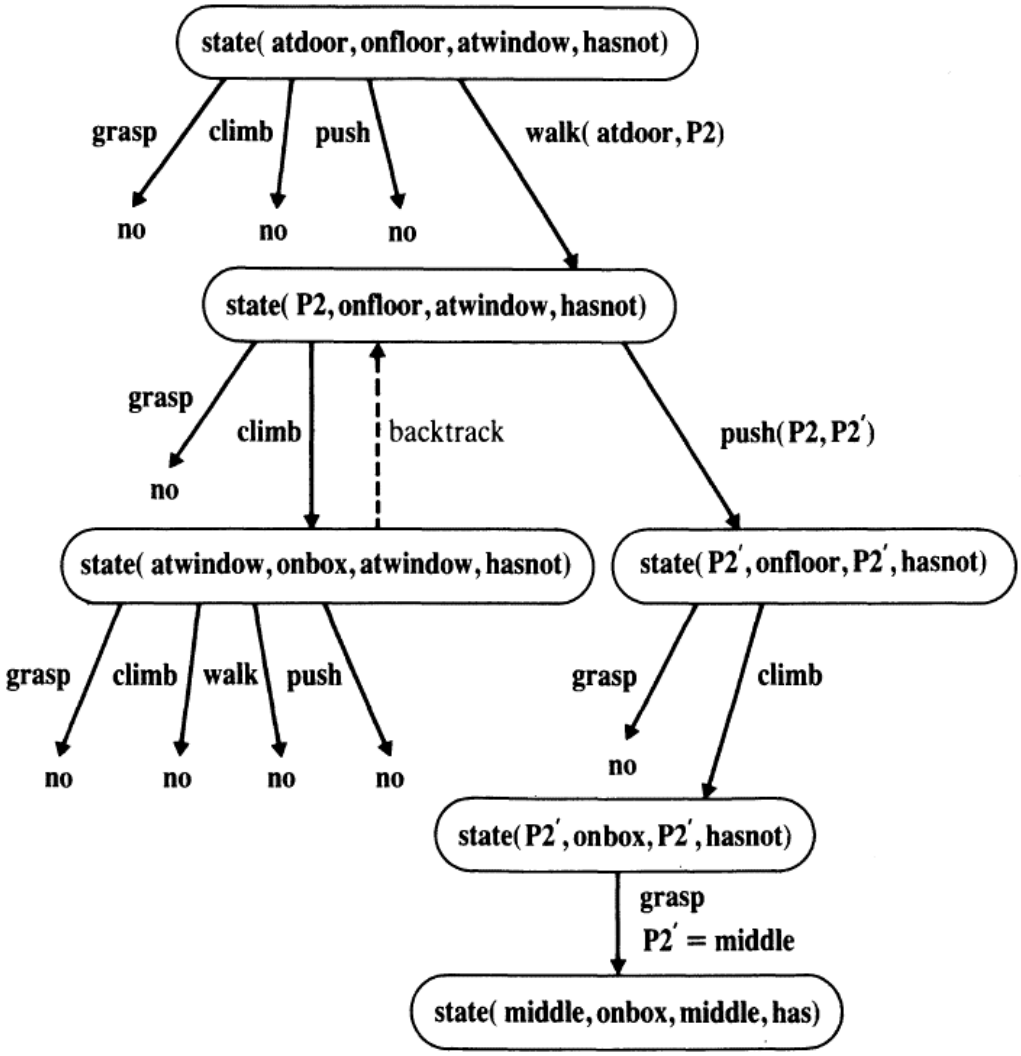
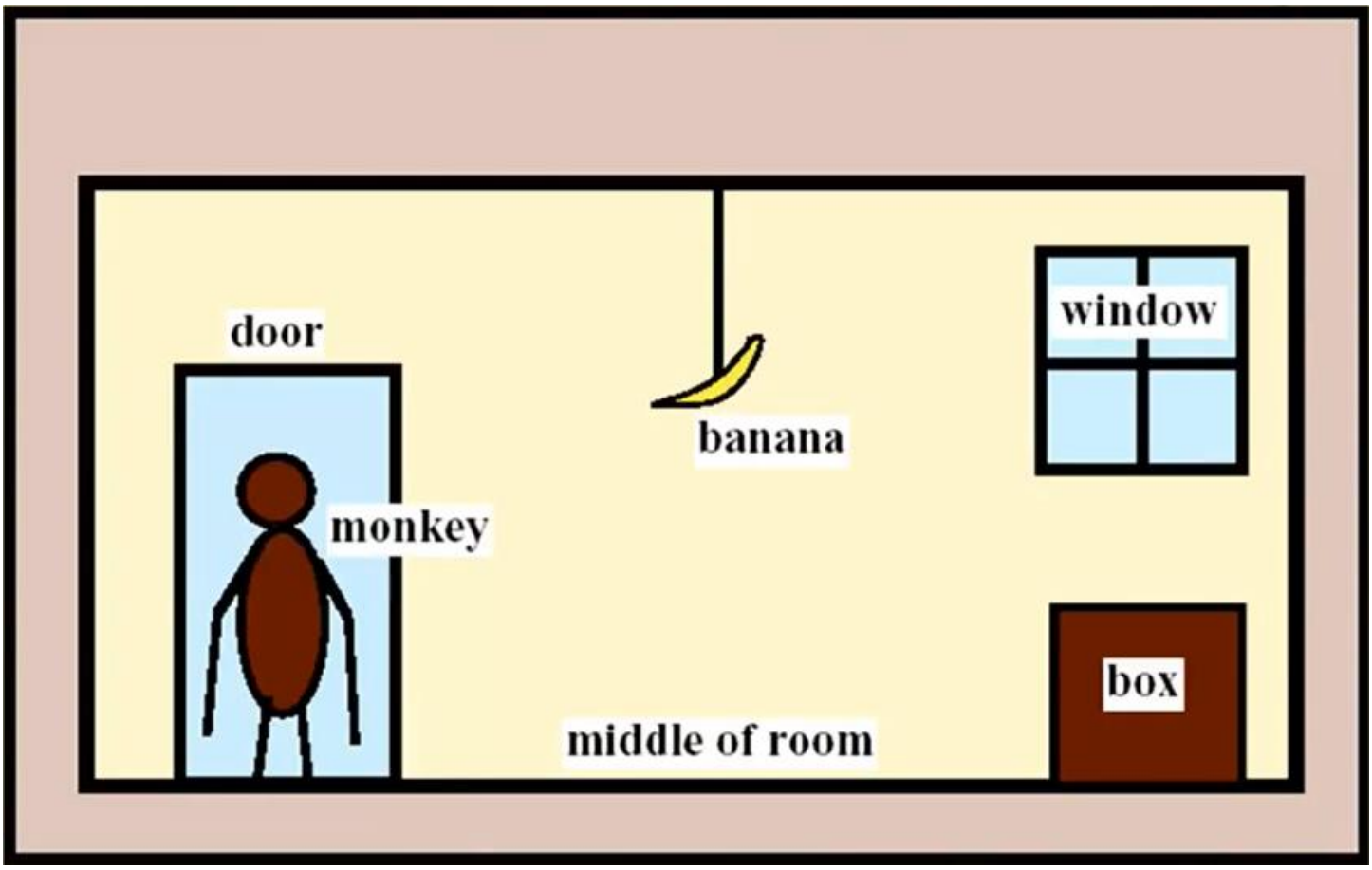



**Figure 2.15** The monkey's search for the banana. The search starts at the top node and proceeds downwards, as indicated. Alternative moves are tried in the left-to-right order. Backtracking occurred once only.

# Prolog Code

```
% Monkey-Banana Problem {[U19CS012] BHAGYA RANA}

% initial state: Monkey is at door, Monkey is on floor, Box is at window, Monkey doesn't have
banana.

% state(Monkey location in the room, Monkey onbox/onfloor, box location, has/hasnot banana)


% Legal Actions Moves

% grab banana
do( state(middle, onbox, middle, hasnot),
    grab,
    state(middle, onbox, middle, has) ).

% climb box
do( state(L, onfloor, L, Banana),
    climb,
    state(L, onbox, L, Banana) ).

% push box from L1 to L2
do( state(L1, onfloor, L1, Banana),
    push(L1, L2),
    state(L2, onfloor, L2, Banana) ).

% walk from L1 to L2
do( state(L1, onfloor, Box, Banana),
    walk(L1, L2),
    state(L2, onfloor, Box, Banana) ).

% canget(State): monkey can get banana in State

% Monkey already has it, goal state
canget(state(_, _, _, has)).

canget(State1):-                    % not goal state, do some work to get it
      do(State1, Action, State2),   % do something (grab, climb, push, walk)
      canget(State2).               % canget from State2

% get plan = list of actions
canget(state(_, _, _, has), []).    % Monkey already has it, goal state

canget(State1, Plan) :-               % not goal state, do some work to get it
      do(State1, Action, State2),     % do something (grab, climb, push, walk)
      canget(State2, PartialPlan),    % canget from State2
      add(Action, PartialPlan, Plan). % add action to Plan

add(X,L,[X|L]).
```




**Output**

```
% c:/users/admin/desktop/ai_lab_6/monkey_banana compiled 0.00 sec, 0 clauses
?- canget(state(atdoor, onfloor, atwindow, hasnot), Plan).
Plan = [walk(atdoor, atwindow), push(atwindow, middle), climb, grab] .

?- canget(state(atwindow, onbox, atwindow, hasnot), Plan).
false.
```

In case 2, Since Monkey can Only Climb once on the Box [There is no Action of Climbing Down the Box], Therefore it returns **False**.

**SUBMITTED BY**: U19CS012

BHAGYA VINOD RANA