Homework - 002

Design a 4 Tile Vacuum Cleaner Robot (Roomba V 0.5) based on Model Based Reflex Agent schema.

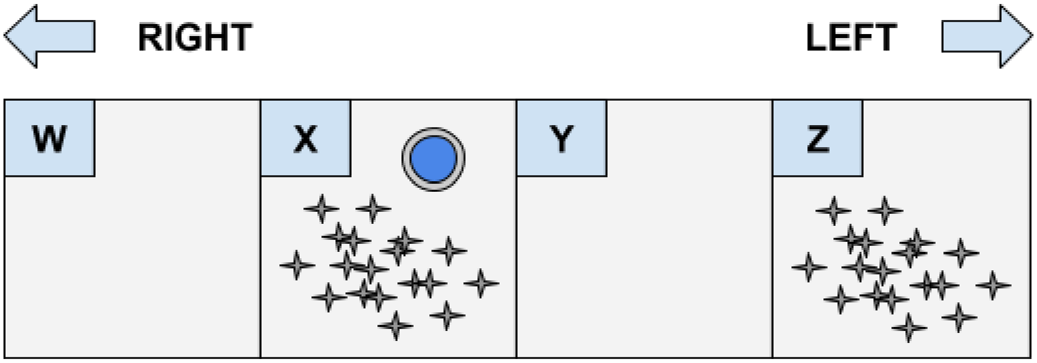


Fig : 4 Tile Environment

Deliverables

1. Write down the Percepts and Actions needed for 4 tile vacuum cleaner.
2. Draw the Decision Tree (or State Tree) for each step until all the tiles cleaned and show the performance measure to choose the right Action. Additionally write pseudo-code for the Action based on percepts and states on each step.
3. Finally write the Action function for 4 tile Vacuum Cleaner Robot.

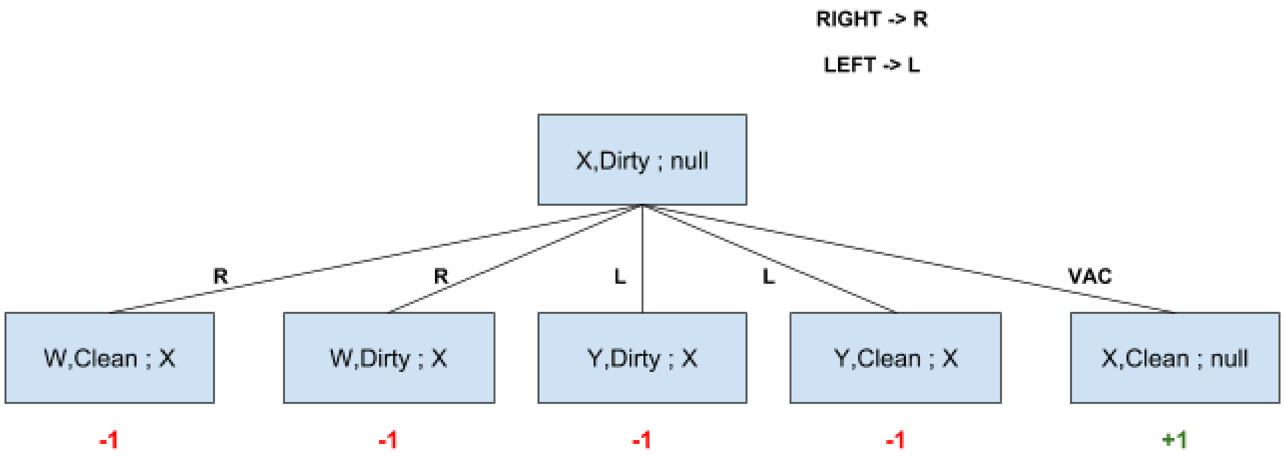
Submission Format

- Hard copy, according to previously shared template.

Answer to Question 1

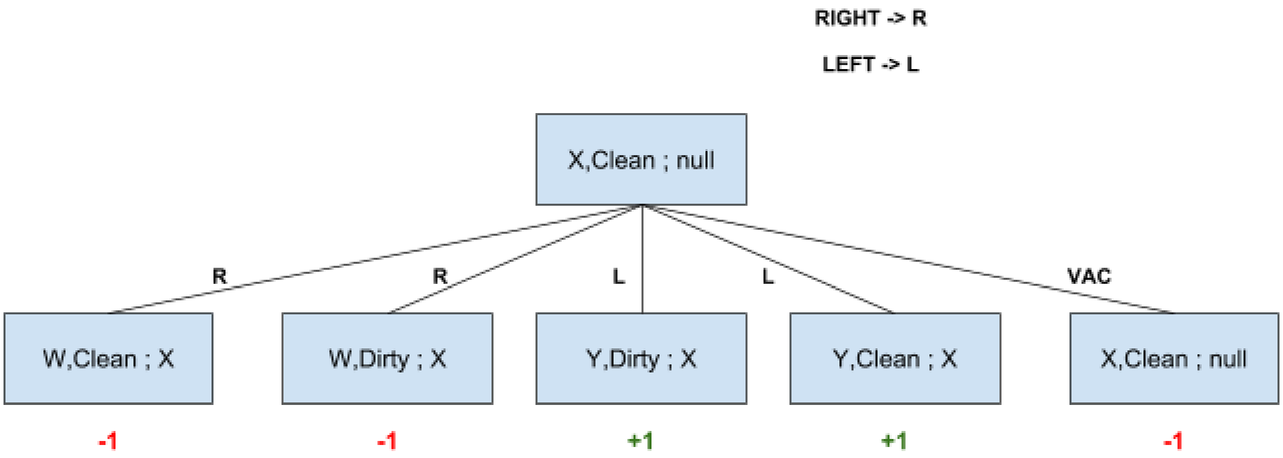
Percepts : **Loc { W, X, Y, Z }**,**Tile_Status {Dirty, Clean}**
Actions : **VAC**,**LEFT**,**RIGHT**
State : **PreLoc { null, W, X, Y, Z }**

Answer to Question 2



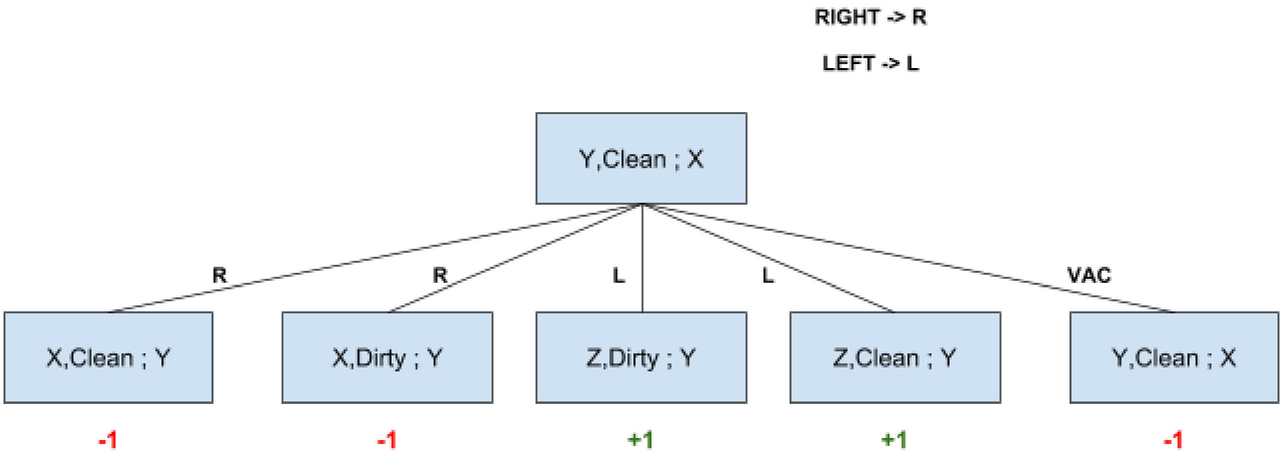
State-1

IF (Percept = **Dirty** && Loc = **X** && PreLoc = **null**) then Action := **VAC**;



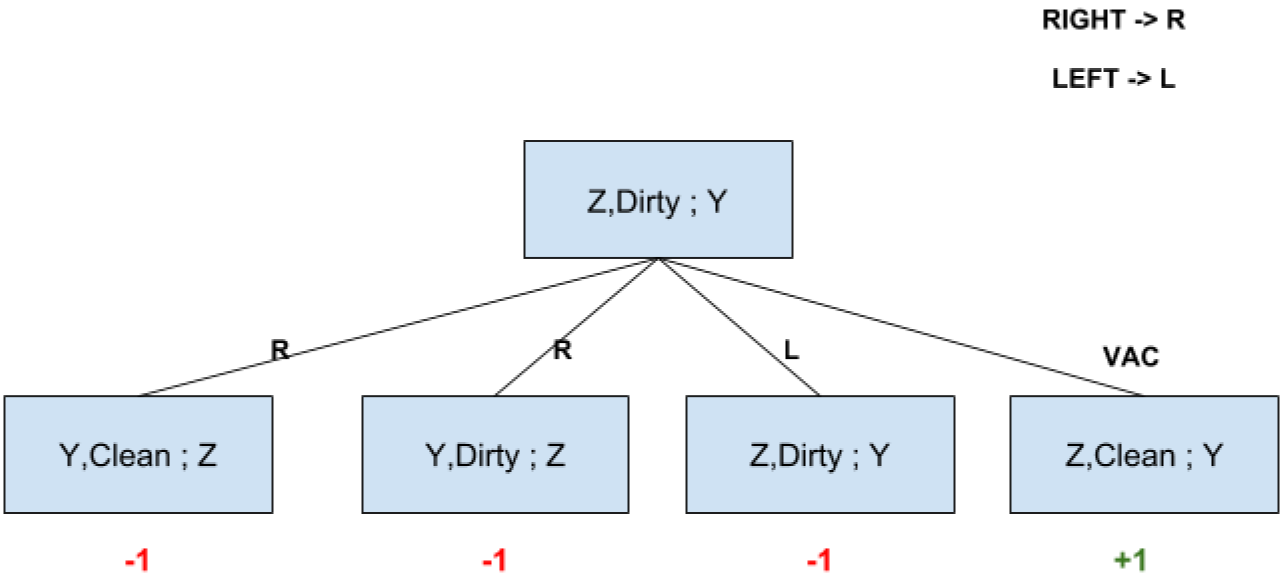
Stage-2 (We choose go Left strategy)

IF (Percept = **Clean** && Loc = **X** && PreLoc = **null**) then Action := **LEFT**;
IF (Action = **RIGHT** || Action = **LEFT**) then PreLoc := Loc;



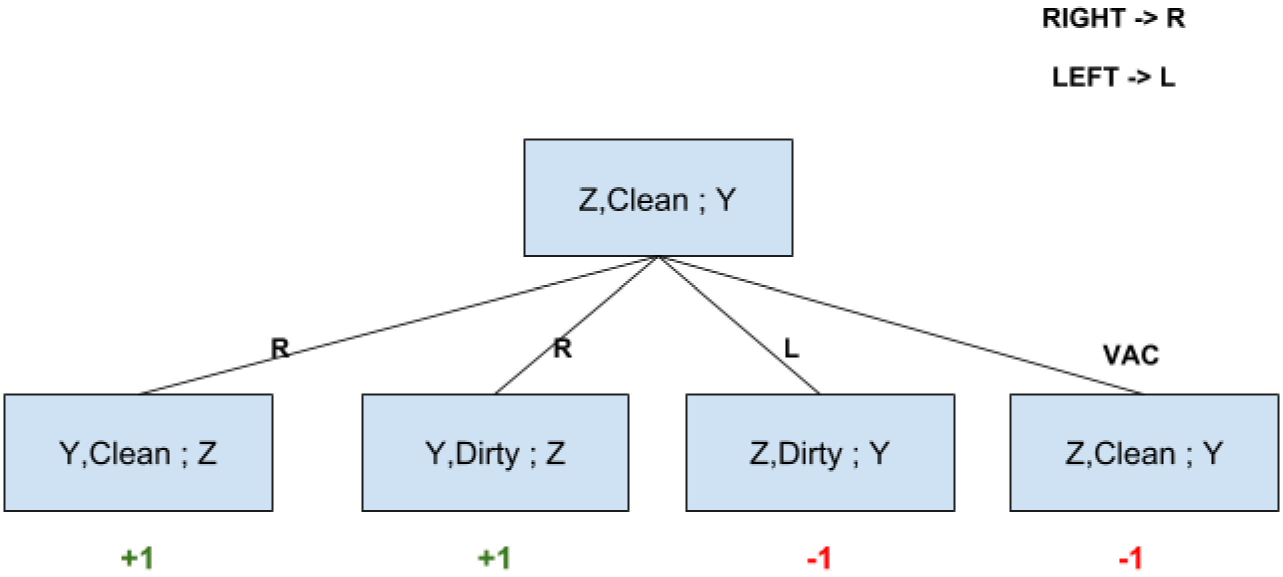
Stage-3

IF (Percept = **Clean** && Loc = **Y** && PreLoc = **X**) then Action := **LEFT**;
IF (Action = **RIGHT** || Action = **LEFT**) then PreLoc := Loc;



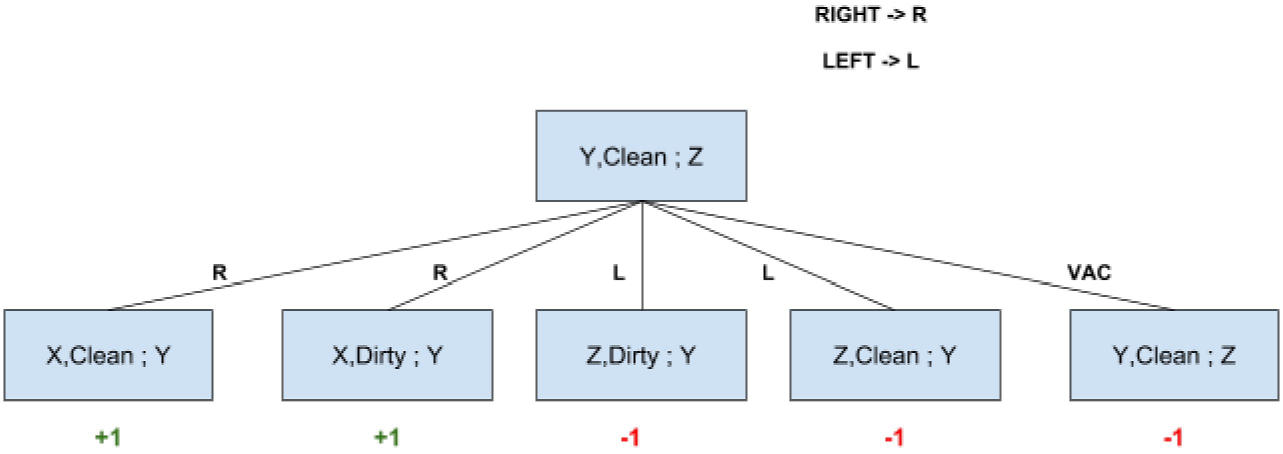
Stage-4

IF (Percept = **Dirty** && Loc = **Z** && PreLoc = **Y**) then Action := **VAC**;



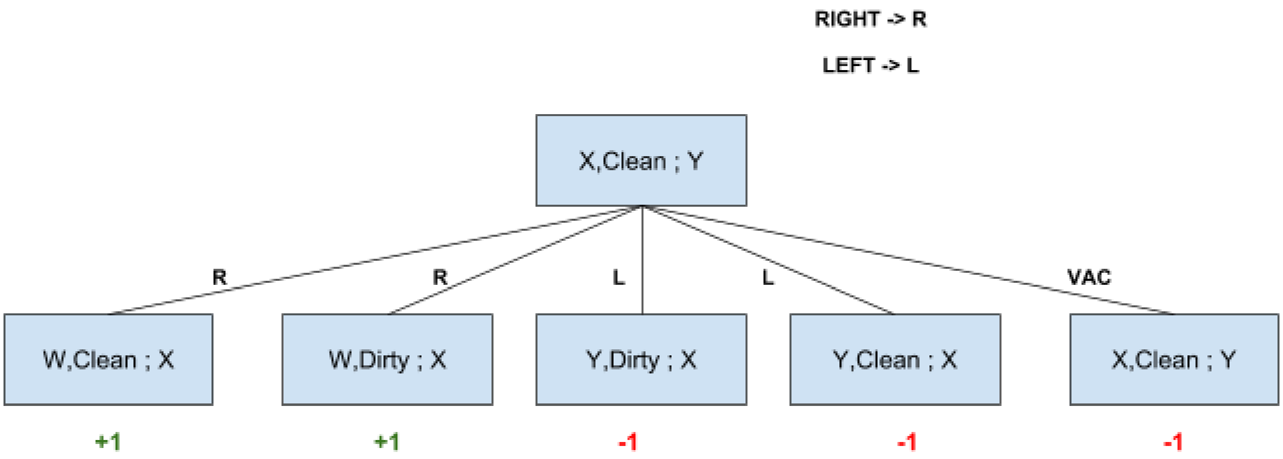
Stage-5

IF (Percept = **Clean** && Loc = **Z** && PreLoc = **Y**) then Action := **RIGHT**;
IF (Action = **RIGHT** || Action = **LEFT**) then PreLoc := Loc;



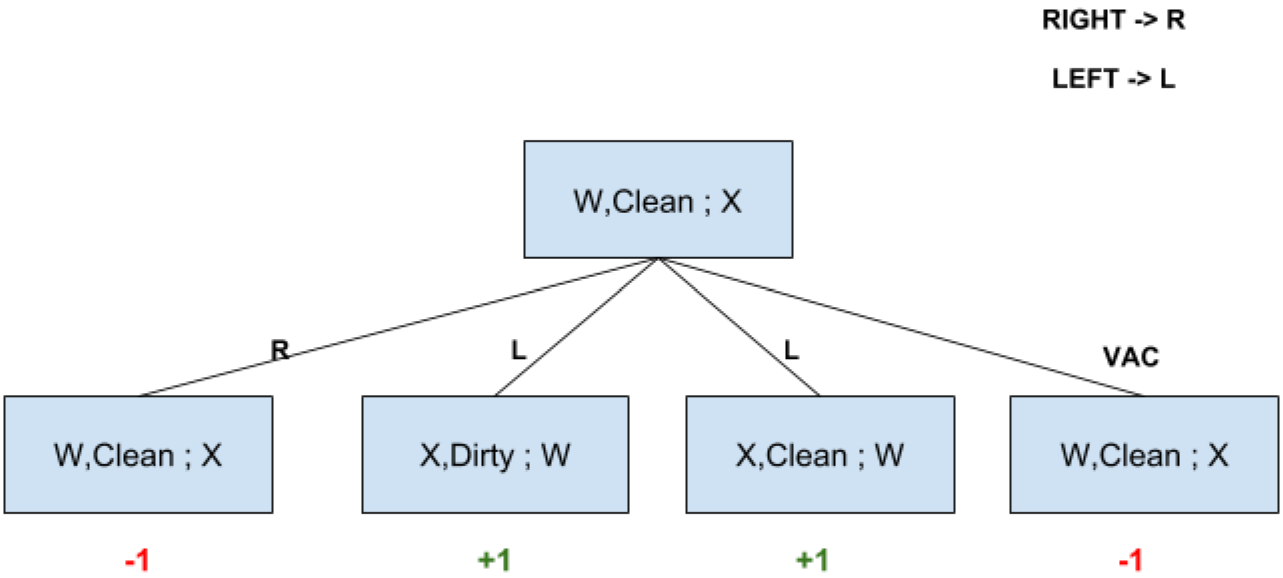
Stage-6

IF (Percept = **Clean** && Loc = **Y** && PreLoc = **Z**) then Action := **RIGHT**;
IF (Action = **RIGHT** || Action = **LEFT**) then PreLoc := Loc;



Stage-7

IF (Percept = **Clean** && Loc = **X** && PreLoc = **Y**) then Action := **RIGHT**;
IF (Action = **RIGHT** || Action = **LEFT**) then PreLoc := Loc;



Stage-8 (Final Stage as Vacuum Cleaner Robot traverse all the tiles)

IF (Percept = **Clean** && Loc = **W** && PreLoc = **X**) then Action := **LEFT**;
IF (Action = **RIGHT** || Action = **LEFT**) then PreLoc := Loc;

Summary: If we want to summarize all the condition-action rules, then we can write like below,

Condition 1

IF Percept = **Dirty** Then Action := **VAC**; // Loc and PerLoc has no impact as it is the most desired work which we accept from our Vacuum Cleaner Robot.

Condition 2

If **Condition 1** becomes not true then we have to consider Loc and PreLoc for any further decision making.

```
IF (Loc = W) {
        IF (PreLoc = null || PreLoc = X)) Then Action := LEFT;
}
ELSE IF (Loc = X) {
        IF (PreLoc = null) Then Action := LEFT; // if we choose go Left policy
        ELSE IF (PreLoc = W) Then Action := LEFT;
        ELSE IF (PreLoc = Y) Then Action := RIGHT;
}
ELSE IF (Loc = Y) {
        IF (PreLoc = null) Then Action := LEFT; // if we choose go Left policy
        ELSE IF (PreLoc = X) Then Action := LEFT;
        ELSE IF (PreLoc = Z) Then Action := RIGHT;
}
ELSE IF (Loc = Z) {
        IF (PreLoc = null || PreLoc = Y)) Then Action := RIGHT;
}
ELSE {
        Action := NO_ACTION; // Just for completeness of code it will never happen for our case
}
```

Condition 3

We have to update PerLoc each time when the Action either Right or Left,

```
IF (Action = LEFT || Action = RIGHT) Then PreLoc := Loc;
```

Alternatively we also can write,

```
IF (Action != VAC) Then PreLoc := Loc;
```

Answer to Question 3

Action Function Pseudocode

```
function getActionForFourTileRobot(Percept, Loc, PreLoc) : returns Action {
    IF (Percept = Dirty) Then Action := VAC;
    ELSE {
        IF (Loc = W) {
            ELSE IF (PreLoc = null || PreLoc = X)) Then Action := LEFT;
        }
        ELSE IF (Loc = X) {
            IF (PreLoc = null) Then Action := LEFT; // if we choose go Left policy
            ELSE IF (PreLoc = W) Then Action := LEFT;
            ELSE IF (PreLoc = Y) Then Action := RIGHT;
        }
        ELSE IF (Loc = Y) {
            IF (PreLoc = null) Then Action := LEFT; // if we choose go Left policy
            ELSE IF (PreLoc = X) Then Action := LEFT;
            ELSE IF (PreLoc = Z) Then Action := RIGHT;
        }
        ELSE IF (Loc = Z) {
            IF (PreLoc = null || PreLoc =Y)) Then Action := RIGHT;
        }
        ELSE {
            Action := NO_ACTION; // Just for completeness of code it will never happen for our case
        }
    }

    IF (Action = LEFT || Action = RIGHT) Then PreLoc := Loc;

    return Action;
}
```

The End…………