7/10/24

Bafna Gold
Date: Page:

LAB 2 -

Vacuum Cleaner Agent Algorithm

Step 1 : Start

Step 2 : Take input of location of agent (A or B) and take the status whether rooms are clean or dirty.

Step 3 : Initialize status to the rooms.

Step 4 :

Declare dictionary for percept

```
agent_table = {
    (Clean, 'A') = MoveRight
    (Clean, 'B') = MoveLeft
    (Dirty, 'A') = Suck
    (Dirty, 'B') = Suck
}
```

Step 5 : Based on the status of rooms while loop runs until both the rooms are clean.

```
while (room_status != 'clean')
    call the action function
```

Step 6 :

```
action function ( )
    if status = dirty return Suck
    else
        if location = A return MoveRight
        if location = B return MoveLeft
```

Step 7 : Print the percept sequence and end the loop when both the rooms are clean.

Step 8 : END

1/10

CODE:

```
agent_table = {
    ('Clean', 'A') = 'MoveRight',
    ('Clean', 'B') = 'MoveLeft',
    ('Dirty', 'A') = 'Suck',
    ('Dirty', 'B') = 'Suck',
}

class VacuumCleaner:
    def __init__(self, loc = 'A', status = 'Clean'):
        self.loc = loc
        self.status = status

    def act(self, action):
        if action == 'MoveRight':
            self.loc = 'B'
        elif action == 'MoveLeft':
            self.loc = 'A'
        elif action == 'Suck':
            self.status = 'Clean'

if __name__ == "__main__":
    status_A = input("Enter status of room A:")
    status_B = input("Enter status of room B:")
    vacuum = VacuumCleaner(loc = 'A', status = status_A)
    while status_A == 'Dirty' or status_B == 'Dirty':
        action = agent_table.get((vacuum.status,
                                  vacuum.loc), 'NoOp')
        print(f" Percept: {vacuum.status}, Action: {action}")
```

```
if (action != 'NoOp'):
    vacuum.act(action)
    if action == 'Suck':
        if vacuum.loc == 'A':
            status_A = 'Clean'
        else:
            status_B = 'Clean'

vacuum.status = staus_A if vacuum_loc='A'
                else status_B
print(f"Location: {vacuum.loc}, StatusA:
    {status_A}, statusB: {status_B}")

print("Both rooms are clean!")
```

OUTPUT:

Enter the status of room A : clean
Enter status of room B : dirty
Percept : clean, Action : Moveright
Location : B , statusA : clean, statusB : Dirty
Percept : Dirty, Action : Suck
Location : B , statusA : clean, statusB : clean
Both rooms are clean!

OUTPUT:

Enter status of A : Dirty
Enter status of B : Clean
Enter status of C : Clean
Enter status of D : ~~Clean Dirty~~ Clean

Percept: Dirty Action: Suck
Location: A
Percept: Clean Action: MoveRight
Location: B
All rooms are clean!