Date 01/09/24
Page 01

Algorithm for Working of Vaccum Cleaner :-

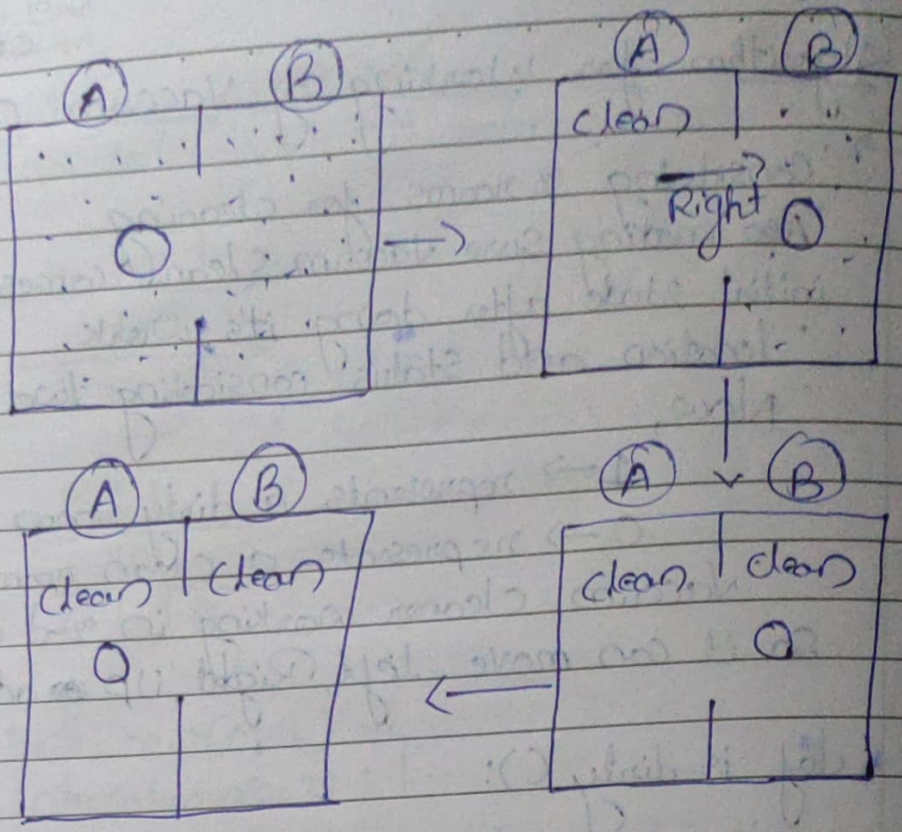
→ Considering 2 rooms for cleaning
Also making sure vacuum cleaner comes to initial state after doing it's work
location and status considering two precepts
Now,
1 → represents a dirty room
0 → represents a clean room
Vaccum cleaner working in 2-d grid, so it can move left, Right, up and down.

```
def is_dirty():
    return self.rooms[self.position] == 1

def clean():
    if self.is_dirty():
        self.rooms[self.position] = 0
        self.cleaned_rooms += 1

def move():
    self.position = 1 - self.position

def run(steps):
    for step in range(steps):
        clean()
        move()
rooms = [1, 0]
```



=> Percept Sequence

Check: RoomA, Dirty
Action: clean RoomA & move

check: RoomB, Dirty
Action: clean RoomB & move

I : → (Room1, Dirty)
II : → (Room2, clean)
III : → (Room1, clean)
IV : → (Room2, clean)
V : → (Room1, clean).

Date / /
Page 09

// program code

```
class VacuumCleaner:

    def __init__(self, rooms, start_position):

        self.rooms = rooms
        self.position = start_position
        self.cleaned_rooms = 0
        self.percept_sequence = []

    def is_dirty(self):
        return self.rooms[self.position] == 1

    def clean(self):

        if self.is_dirty():
            print(f"Cleaning room {self.position + 1}")
            self.rooms[self.position] = 0
            self.cleaned_rooms += 1

    def move(self):

        if self.position < len(self.rooms) - 1:
            self.position += 1

        else
            self.position = 0
        print(f"moved to room {self.position + 1}")

    def perceive(self):

    room_state = "Dirty" if self.is_dirty()
```

```
        else "clean"
        percept = (f"Room {self.position+1}", room.status)
        self.percept_sequence.append(percept)
        print(f"perception: {percept}")

    def run(self, steps):
        for step in range(steps):
            print(f"Step {step+1}:")
            self.perceive()
            self.clean()
            self.move()
            print(f"rooms status: {self.rooms}\n")

        print(f"Total cleaned rooms: {self.
                cleaned_rooms}")
        print("percept sequence:", self.percept_
                sequence)

rooms = [1,0,1,1]

vacuum.run(steps=8)
```

=> Output :—

Step 1:
Perception : ('Room1', 'Dirty')

Step 1:
Perception : ('Room1', 'Clean').

moved to room 2

Step2:
Perception: ('Room 2', Dirty')
cleaning room2
Moved to room 3

Step3:
Perception: ('Room 3', Dirty')
Cleaning room3
moved to room4

Step4:
Perception: ('Room 4', Dirty')
Cleaning room4
moved to room1

Step5:
Perception: ('Room1', 'Clean')
moved to room2

Step6:
Perception: ('Room2', 'Clean')
Moved to Room3

Step7:
Perception: ('Room 3', 'Clean')
Moved to Room4

Step8:
Perception: ('Room4', 'Clean')
Moved to room1

Total cleaned rooms: 3

Date / /
Page

Prompt Sequence : [('Room1', 'clean'),
('Room2', 'Dirty'),
('Room3', 'Dirty'),
('Room4', 'Dirty'),
('Room1', 'clean'),
('Room2', 'clean'),
('Room3', 'clean'),
('Room4', 'clean')] //.

Sathish
1/10/24