LAB-02 01|R01|24

2. Vacuum Cleaner Agent

→ Algorithm:

1. Start at the intial room. chuck the left room first.
2. Percept if the room is dirty or clean.
3. Clean the room (if dirty), mark it clean
4. Move to the other room on the right
5. Clean the current room (if dirty) & mark it clean.
6. If room is missed go back.
7. If all clean, terminate the program.

→ PSEUDOCODE:

```
START-
    left_room is dirty:
        clean left_room
    move to right_room
    IF right_room is dirty:
        clean right_room
    IF both rooms are clean:
        TERMINATE
END
```

→ Percept Sequence:

Percept 1: Start in the left room, observe that it is dirty.
- Action: Clean the left room.

Percept 2: Move to the right room, observe that it is clean.
- Action: Do nothing

Percept 3: Return to the left room to confirm.
- Action: End /Terminate

→ PSEUDOCODE:

START

```
array: 0 = clean, 1 = dirty
rooms = [1, 0]  # [left_room, right_room],
# left is dirty & right is clean

current_room = 0   # start left (index 0)

WHILE rooms are not all clean:
    IF rooms[curren_room] == 1:  # is dirty.
    clean rooms[current_room]
    current_room.state = clean
    current_room. = current_room.next
IF current_room is null:
        current_room is left_room
```

END

→ Percept sequence:

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 1. | percept | | | 2. | | percept | |
| | Dirty | Dirty | | | Dirty | Dirty | |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 3. | percept | | | 4. | | percept | |
| | Dirty | clean | | | dirty | clean | |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 5. | clean | | | 6. | clean | percept | |
| | | dirty | | | | dirty | |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 7. | percept | clean | | 8. | clean | clean | |

terminate

→ CODE:

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

class VacuumCleaner:
    def __init__(self, location = 'A', status = 'Clean'):
        self.location = location
        self.status = {status_a: 'A', 'B': status_b}

    def percept(self):
        return self.status[self.location]

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

def table_driven_agent(percept):
    return agent_table.get(percept, 'NoOp')

if __name__ == "__main__":
    status_a = input("Is room A 'Clean' or 'Dirty'?").
                strip().capitalize()
    status_b = input(        B        ).strip().
                capitalize()
```

```
vacuum = VacuumCleaner(status_a = status_a,
                       status_b = status_b)

for _ in range(3):
    current_percept = vacuum.percept()
    action = table_driven_agent((current_percept,
                                 vacuum.location))
    print(f"percept: {current_percept}, Action: {
          action}")

    if action != 'NoOp':
        vacuum.act(action)

    print(f"location: {vacuum_location}
          status: {vacuum.status:}\n")
```

→ Output:

```
Is room A 'Clean' or 'Dirty'? :Dirty
Is room B 'Clean' or 'Dirty'? Dirty

Percept: Dirty, Action: Suck
Location: A, Status: {'A': 'Clean', 'B': 'Dirty'}

Percept: Dirty Clean, Action: MoveRight
Location: B, status: {'A': 'Clean', 'B': 'Dirty'}

Percept: Dirty, Action: ~~MoveR Clean~~ Suck
Location: B, status: {'A': 'Clean', 'B': 'Clean'}
```

1/10/24

```
Is room A 'Clean' or 'Dirty'? Dirty
Is room B 'Clean' or 'Dirty'? Dirty
Percept: Dirty, Action: Suck
Location: A, Status: {'A': 'Clean', 'B': 'Dirty'}

Percept: Clean, Action: MoveRight
Location: B, Status: {'A': 'Clean', 'B': 'Dirty'}

Percept: Dirty, Action: Suck
Location: B, Status: {'A': 'Clean', 'B': 'Clean'}
```

```
Is room A 'Clean' or 'Dirty'? dirty
Is room B 'Clean' or 'Dirty'? dirty
Is room C 'Clean' or 'Dirty'? dirty
Is room D 'Clean' or 'Dirty'? dirty
Percept: Dirty, Action: Suck
Location: A, Status: {'A': 'Clean', 'B': 'Dirty', 'C': 'Dirty', 'D': 'Dirty'}

Percept: Clean, Action: MoveRight
Location: B, Status: {'A': 'Clean', 'B': 'Dirty', 'C': 'Dirty', 'D': 'Dirty'}

Percept: Dirty, Action: Suck
Location: B, Status: {'A': 'Clean', 'B': 'Clean', 'C': 'Dirty', 'D': 'Dirty'}

Percept: Clean, Action: MoveRight
Location: C, Status: {'A': 'Clean', 'B': 'Clean', 'C': 'Dirty', 'D': 'Dirty'}

Percept: Dirty, Action: Suck
Location: C, Status: {'A': 'Clean', 'B': 'Clean', 'C': 'Clean', 'D': 'Dirty'}

Percept: Clean, Action: MoveRight
Location: D, Status: {'A': 'Clean', 'B': 'Clean', 'C': 'Clean', 'D': 'Dirty'}

Percept: Dirty, Action: Suck
Location: D, Status: {'A': 'Clean', 'B': 'Clean', 'C': 'Clean', 'D': 'Clean'}

All rooms are clean!
```