

Program Title: vacuum cleaner agent

code:

```
# Define the grid environment
class Environment:
    def __init__(self, width, height):
        self.width = width
        self.height = height
        self.grid = [[random.choice([0, 1]) for _ in range(width)] for _ in range(height)] # 0: Clean,
1: Dirty
        self.agent_pos = [random.randint(0, height - 1), random.randint(0, width - 1)]

    def is_dirty(self, x, y):
        return self.grid[x][y] == 1

    def clean(self, x, y):
        self.grid[x][y] = 0

    def display(self):
        for row in self.grid:
            print(row)
        print(f"Agent position: {self.agent_pos}\n")

# Define the Vacuum Cleaner Agent
class VacuumCleanerAgent:
    def __init__(self, environment):
        self.env = environment

    def move(self):
        # Move agent randomly in one of the four directions
        direction = random.choice(["UP", "DOWN", "LEFT", "RIGHT"])
        x, y = self.env.agent_pos

        if direction == "UP" and x > 0:
            self.env.agent_pos[0] -= 1
        elif direction == "DOWN" and x < self.env.height - 1:
            self.env.agent_pos[0] += 1
        elif direction == "LEFT" and y > 0:
            self.env.agent_pos[1] -= 1
        elif direction == "RIGHT" and y < self.env.width - 1:
            self.env.agent_pos[1] += 1

    def clean(self):
        x, y = self.env.agent_pos
```

```
if self.env.is_dirty(x, y):
    self.env.clean(x, y)
    print(f"Cleaned dirt at position: {x}, {y}")

def step(self):
    self.clean() # Clean the current location if dirty
    self.move() # Move to a new location

# Initialize the environment and agent
width, height = 5, 5 # 5x5 grid
env = Environment(width, height)
agent = VacuumCleanerAgent(env)

# Run the simulation
steps = 10
for step in range(steps):
    print(f"Step {step + 1}:")
    agent.step()
    env.display()
```

② Implement vacuum cleaner agent

Algorithm:

1. Sense the environment:

- The agent perceives two things
 - its current location (A or B)
 - The status of that location (dirty or clean)

2. Decision process:

- If the current location is Dirty
 - The agent takes the action Suck to clean the location
- Else if the current location is A
 - The agent moves right to location B
- Else if the current location is B
 - The agent moves left to location A

3. Act:

- The agent performs the action based on its perception and the decision rules.

4. Repeat:

- After the action is taken the agent repeats the process when a new percept is received.

Output:

Percept : [A, 'clean'], Action: Right

Percept : [A, 'dirty'], Action: Suck

Percept : [B, 'clean'], Action: Left

Percept : [B, 'dirty'], Action: Suck

Percept : [A, 'clean'], Action: Right

Percept : [A, 'clean'], Action: Right

Percept Sequence: $\{[A, \text{'clean'}], [A, \text{'dirty'}]$
 $[B, \text{'clean'}], [B, \text{'dirty'}], [A, \text{'clean'}], [A, \text{'clean'}]\}$

Action Sequence: $\{\text{'right'}, \text{'suck'}, \text{'left'},$
 $\text{'suck'}, \text{'right'}, \text{'right'}\}$