2.IMPLEMENTATION OF VACUUM CLEANER

```
import random

class VacuumCleaner:
    def __init__(self, grid_size):
        self.grid_size = grid_size
        self.grid = [['dirty' for _ in range(grid_size)] for _ in range(grid_size)]
        self.position = (0, 0)  # Start position
        self.cleaned_count = 0

    def display_grid(self):
        for row in self.grid:
            print(" ".join(row))
        print()

    def clean(self):
        x, y = self.position
        if self.grid[x][y] == 'dirty':
            self.grid[x][y] = 'clean'
            self.cleaned_count += 1
            print(f"Cleaned position {self.position}. Total cleaned: {self.cleaned_count}")
        else:
            print(f"Position {self.position} is already clean.")

    def move(self):
        x, y = self.position
        # Randomly choose a direction to move (up, down, left, right)
        direction = random.choice(['up', 'down', 'left', 'right'])

        if direction == 'up' and x > 0:
            self.position = (x - 1, y)
        elif direction == 'down' and x < self.grid_size - 1:
            self.position = (x + 1, y)
        elif direction == 'left' and y > 0:
            self.position = (x, y - 1)
        elif direction == 'right' and y < self.grid_size - 1:
            self.position = (x, y + 1)

        print(f"Moved to position {self.position}.")
        self.clean()

    def run(self, steps):
        for _ in range(steps):
```

```
            self.move()
            self.display_grid()


# Example usage
if __name__ == "__main__":
    vacuum = VacuumCleaner(grid_size=5)
    vacuum.run(10)
```

② Impliment 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

Bafna Gold
Date: Page: (9)

Percept Sequence : [['A', 'clean'], ['A', 'Dirty'], ['B', 'clean'], ['B', 'Dirty'], ['A', 'clean'], ['A', 'clean']]
Action Sequence : ['Right', 'Suck', 'Left', 'Suck', 'Right', 'Right']