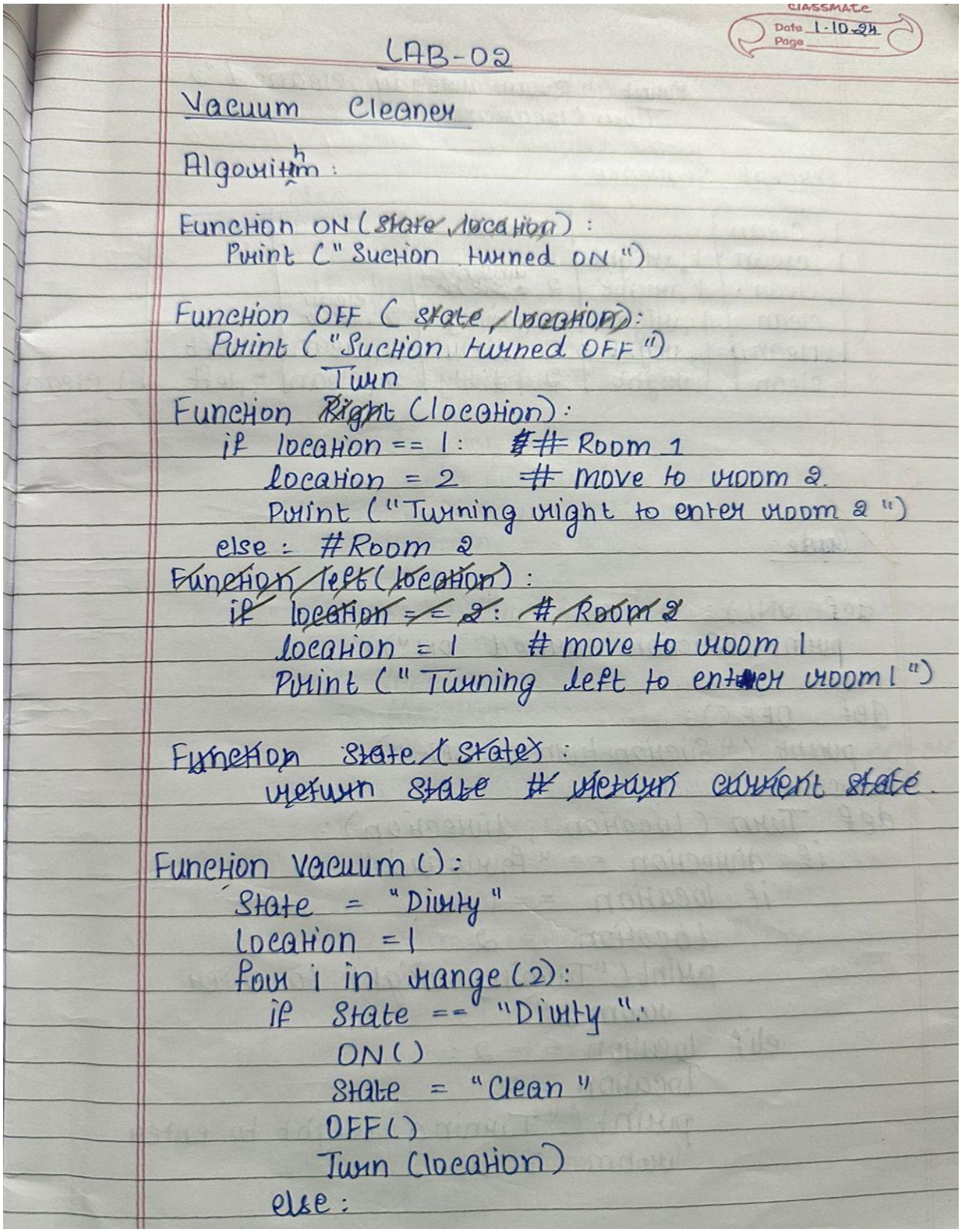
LAB:2 Vacuum cleaner

Observation book:



LAB-02

Vacuum Cleaner

Algorithm:

```
Function ON (state, location):
    Print ("Suction turned ON")

Function OFF (state, location):
    Print ("Suction turned OFF")
        Turn
Function Right (location):
    if location == 1:    ## Room 1
        location = 2     # move to room 2.
        Print ("Turning right to enter room 2")
    else: #Room 2
Function left (location):
    if location == 2: # Room 2
        location = 1     # move to room 1
        Print ("Turning left to enter room 1")

Function State (state):
    return State # return current state.

Function Vacuum():
    State = "Dirty"
    location = 1
    for i in range(2):
        if State == "Dirty":
            ON()
            State = "Clean"
            OFF()
            Turn (location)
        else:
```

print("Room already cleaned")
Turn(location).

Percept Sequence.

1, clean					
1, clean	1, right				
1, clean	1, right	2, dirty clean			
1, clean	1, right	2, dirty	2, clean		
1, clean	1, right	2, dirty	2, clean	2, left	
1, clean	1, right	2, dirty	2, clean	2, left.	1, Clea

Code:

```
def ON():
    print("Suction turned ON").

def OFF():
    print("Suction turned OFF")

def Turn(location, direction):
    if direction == "forward":
        if location == 1:
            location = 2.
            print("Turning right to enter
                room 2")
        elif location == 2:
            location = 3
            print("Turning right to enter
                room 3")
```

```
        elif location == 3:
            location = 4
            print("Turning right to enter
                room 4")
        else:
            pass
    else:
        if location == 2:
            location == 1
            print("Turning left to enter
                room 1").
        elif location == 3:
            location = 2
            print("Turning left to enter
                room 2")
        elif location == 4:
            location = 3
            print("Turning left to enter room
                3")
        elif location = 4
            location
            print("Turning right to enter
            room 4")
    return location
state = "Dirty"
location = 1
print("Starting at room 1, Dirty")
print
for i in range(4):
    if state == "Dirty":
        ON()
        state = "clean"
        print("Room is clean")
```

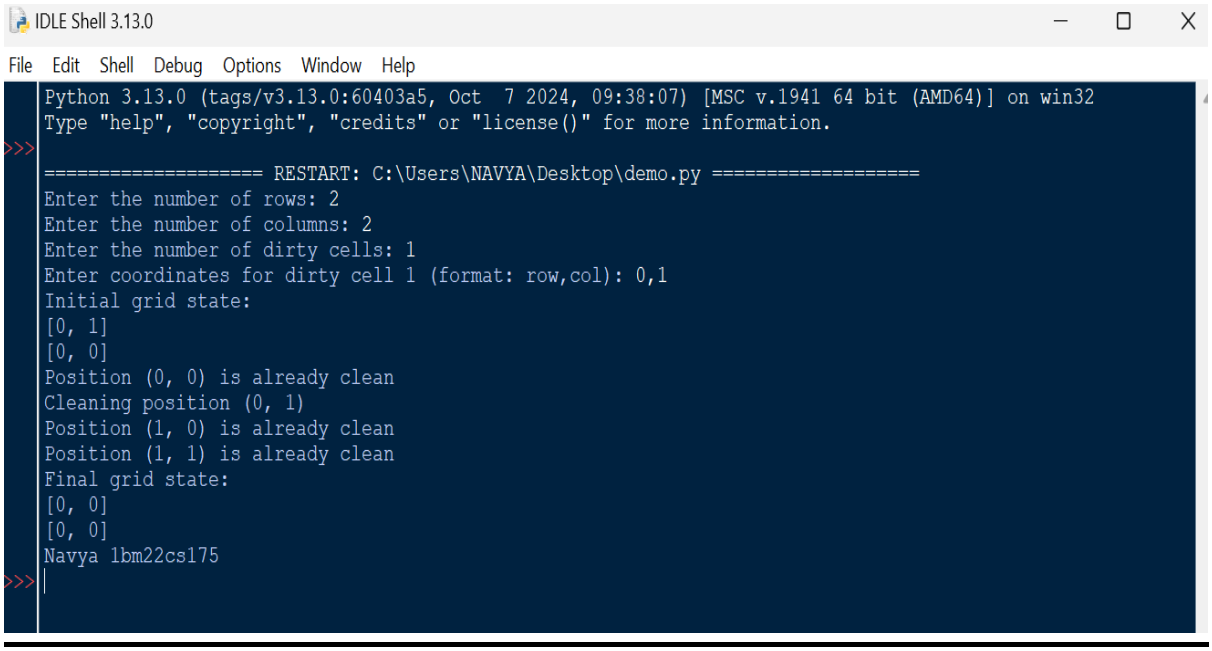
```
        location = Turn(location, "Forward")
    else:
        print("Room is already cleaned")

for i in range(3):
    if state == "Dirty":
        ON()
        state = "clean"
        print("Room is clean now")
        OFF()
        location = Turn(location, "reverse")
    else:
        print("Room is clean")
        location = Turn(location, "reverse
```

Output:

```
Starting room 1, Dirty
Suction turned ON
Room is clean
Suction turned OFF
Turning right to enter room 2.
room is already clean
    "          room 3
room is already clean
    "          room 4
room is already clean
Turning left to enter room 3
room is already cleaned
```

Output:


IDLE Shell 3.13.0
File Edit Shell Debug Options Window Help
Python 3.13.0 (tags/v3.13.0:60403a5, Oct 7 2024, 09:38:07) [MSC v.1941 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
========== RESTART: C:\Users\NAVYA\Desktop\demo.py ==========
Enter the number of rows: 2
Enter the number of columns: 2
Enter the number of dirty cells: 1
Enter coordinates for dirty cell 1 (format: row,col): 0,1
Initial grid state:
[0, 1]
[0, 0]
Position (0, 0) is already clean
Cleaning position (0, 1)
Position (1, 0) is already clean
Position (1, 1) is already clean
Final grid state:
[0, 0]
[0, 0]
Navya 1bm22cs175
>>>