LAB - 2

```
def reflex(loc, status, cost):
    s = status  # Track the current status of the
location
    if status == 1:  # If the location is dirty
        cost += 1
        print(f"SUCK at {loc}")
        s = 0  # The location is now clean
    if loc == "A":
        print("Move RIGHT to B")
        loc = "B"  # Move to B
    elif loc == "B":
        print("Move LEFT to A")
        loc = "A"  # Move to A
    return cost, loc, s  # Return updated cost, location,
and status

def goal(a_status, b_status):
    if a_status == 0 and b_status == 0:
        print("Goal reached")
    else:
        print("Goal not reached")

loc = input("Enter the starting location of the vacuum (A
or B): ")
cost = 0
a_status = int(input("Enter the status of location A (0
for clean, 1 for dirty): "))
b_status = int(input("Enter the status of location B (0
for clean, 1 for dirty): "))

cost, loc, a_status = reflex("A", a_status, cost)

cost, loc, b_status = reflex("B", b_status, cost)

print(f"Total cost: {cost}")

goal(a_status, b_status)
```

Output:

```
Enter the starting location of the vacuum (A or B): A
Enter the status of location A (0 for clean, 1 for dirty): 0
Enter the status of location B (0 for clean, 1 for dirty): 0
Move RIGHT to B
Move LEFT to A
Total cost: 0
Goal reached
```

```
Enter the starting location of the vacuum (A or B): B
Enter the status of location A (0 for clean, 1 for dirty): 1
Enter the status of location B (0 for clean, 1 for dirty): 1
SUCK at A
Move RIGHT to B
SUCK at B
Move LEFT to A
Total cost: 2
Goal reached
```

```
Enter the starting location of the vacuum (A or B): A
Enter the status of location A (0 for clean, 1 for dirty): 1
Enter the status of location B (0 for clean, 1 for dirty): 0
SUCK at A
Move RIGHT to B
Move LEFT to A
Total cost: 1
Goal reached
```

```
Enter the starting location of the vacuum (A or B): A
Enter the status of location A (0 for clean, 1 for dirty): 0
Enter the status of location B (0 for clean, 1 for dirty): 1
Move RIGHT to B
SUCK at B
Move LEFT to A
Total cost: 1
Goal reached
```