2. Implement vacuum cleaner agent.

```
def vacuum_world():
goal_state = {'A': '0', 'B': '0'}
cost = 0
location_input = input("Enter Location of Vacuum (A or B): ")
status_input = input(f"Enter status of {location_input} (0 for clean, 1 for dirty): ")
status_input_complement = input("Enter status of the other room (0 for clean, 1 for dirty): ")
print("Initial Location Condition:", goal_state)
if location_input == 'A':
    print("Vacuum is placed in Location A."
    if status_input == '1':
        print("Location A is Dirty.")
        goal_state['A'] = '0'
        cost += 1
        print("Cost for CLEANING A:", cost)
        print("Location A has been Cleaned.")

    if status_input_complement == '1':
        print("Location B is Dirty.")
        print("Moving right to Location B.")
        cost += 1  # Cost for moving
        print("COST for moving RIGHT:", cost)
        goal_state['B'] = '0'
        cost += 1  # Cost for sucking
        print("COST for SUCK:", cost)
        print("Location B has been Cleaned.")
    else:
        print("Location B is already clean.")

elif location_input == 'B':
    print("Vacuum is placed in Location B.")

    if status_input == '1':
```

```
        print("Location B is Dirty.")
        goal_state['B'] = '0'
        cost += 1
        print("COST for CLEANING B:", cost)
        print("Location B has been Cleaned.")

        if status_input_complement == '1':
            print("Location A is Dirty.")
            print("Moving left to Location A.")
            cost += 1  # Cost for moving
            print("COST for moving LEFT:", cost)
            goal_state['A'] = '0'
            cost += 1  # Cost for sucking
            print("COST for SUCK:", cost)
            print("Location A has been Cleaned.")
        else:
            print("Location A is already clean.")
    else:
        print("Location B is already clean.")
        if status_input_complement == '1':
            print("Location A is Dirty.")
            print("Moving left to Location A.")
            cost += 1  # Cost for moving
            print("COST for moving LEFT:", cost)
            goal_state['A'] = '0'
            cost += 1  # Cost for sucking
            print("COST for SUCK:", cost)
            print("Location A has been Cleaned.")
        else:
            print("Location A is already clean.")

else:
    print("Invalid location input. Please enter A or B.")
```

```
    print("GOAL STATE:", goal_state)
    print("Performance Measurement:", cost)


# To run the function:
vacuum_world()
```

Output:

```
Enter Location of Vacuum (A or B):  A
Enter status of A (0 for clean, 1 for dirty):  0
Enter status of the other room (0 for clean, 1 for dirty):  1
Initial Location Condition: {'A': '0', 'B': '0'}
Vacuum is placed in Location A.
Location B is Dirty.
Moving right to Location B.
COST for moving RIGHT: 1
COST for SUCK: 2
Location B has been Cleaned.
GOAL STATE: {'A': '0', 'B': '0'}
Performance Measurement: 2
```