Implement Vacuum Cleaner Agent

```
def vacuum_world():

    goal_state = {'A': '0', 'B': '0'}
    cost = 0

    location_input = input("Enter Location of Vacuum: ")
    status_input = input("Enter status of " + location_input + " (0 for Clean, 1 for Dirty): ")
    status_input_complement = input("Enter status of other room (0 for Clean, 1 for Dirty): ")

    print("Initial Location Condition: " + str(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: " + str(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
                print("COST for moving RIGHT: " + str(cost))
                goal_state['B'] = '0'
                cost += 1
                print("COST for SUCK: " + str(cost))
                print("Location B has been Cleaned.")
            else:
                print("No action needed; Location B is already clean.")
        else:
            print("Location A is already clean.")

            if status_input_complement == '1':
                print("Location B is Dirty.")
                print("Moving RIGHT to Location B.")
                cost += 1
```

```
            print("COST for moving RIGHT: " + str(cost))
            goal_state['B'] = '0'
            cost += 1
            print("COST for SUCK: " + str(cost))
            print("Location B has been Cleaned.")
        else:
            print("No action needed; Location B is already clean.")

else:

    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: " + str(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
            print("COST for moving LEFT: " + str(cost))
            goal_state['A'] = '0'
            cost += 1
            print("COST for SUCK: " + str(cost))
            print("Location A has been Cleaned.")
        else:
            print("No action needed; 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
            print("COST for moving LEFT: " + str(cost))
            goal_state['A'] = '0'
            cost += 1
            print("COST for SUCK: " + str(cost))
            print("Location A has been Cleaned.")
        else:
            print("No action needed; Location A is already clean.")
```

```
    print("GOAL STATE: ")
    print(goal_state)
    print("Performance Measurement: " + str(cost))
    print("Vaibhav Urs A N")
    print("1BM22CS315")

vacuum_world()
```

OUTPUT

```
Enter Location of Vacuum: B
Enter status of B (0 for Clean, 1 for Dirty): 1
Enter status of other room (0 for Clean, 1 for Dirty): 1
Initial Location Condition: {'A': '0', 'B': '0'}
Vacuum is placed in Location B
Location B is Dirty.
COST for CLEANING B: 1
Location B has been Cleaned.
Location A is Dirty.
Moving LEFT to Location A.
COST for moving LEFT: 2
COST for SUCK: 3
Location A has been Cleaned.
GOAL STATE:
{'A': '0', 'B': '0'}
Performance Measurement: 3
Vaibhav Urs A N
1BM22CS315

=== Code Execution Successful ===
```