

```

def vacuum_world():
    goal_state = {'A': '0', 'B': '0'}
    cost = 0

    location_input = input("Enter Location of Vacuum (A/B): ").upper()
    status_input = input(f"Enter status of {location_input} (0 for Clean, 1 for Dirty): ")
    other_location = 'B' if location_input == 'A' else 'A'
    status_input_complement = input(f"Enter status of {other_location} (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
            print("COST for moving RIGHT:", cost)
            goal_state['B'] = '0'
            cost += 1
            print("COST for SUCK:", cost)
            print("Location B has been Cleaned.")
        else:
            print("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:", 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:", cost)
            goal_state['A'] = '0'
            cost += 1
            print("COST for SUCK:", cost)
            print("Location A has been Cleaned.")
        else:
            print("Location A is already clean.")

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

vacuum_world()

```

```

➡ Enter Location of Vacuum (A/B): A
Enter status of A (0 for Clean, 1 for Dirty): 1
Enter status of B (0 for Clean, 1 for Dirty): 1
Initial Location Condition: {'A': '0', 'B': '0'}
Vacuum is placed in Location A
Location A is Dirty.
Cost for CLEANING A: 1
Location A has been Cleaned.
Location B is Dirty.
Moving right to Location B.
COST for moving RIGHT: 2
COST for SUCK: 3
Location B has been Cleaned.
GOAL STATE: {'A': '0', 'B': '0'}
Performance Measurement: 3

```

