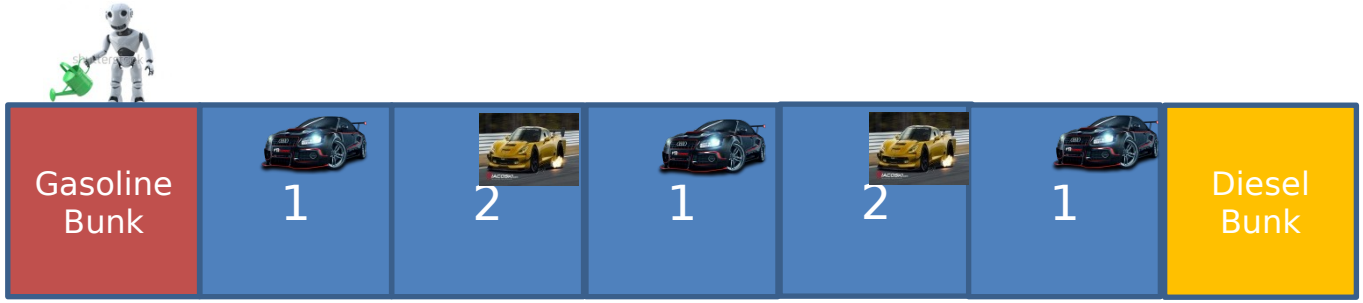
Robot Car Fueling

21-Sep-2016 Advance Problem

Chandru Byadgi & Rohit Bendre

Problem statement:

There are N cars parked in a row in a parking lot of the newly constructed club. as it is demonstrated in the picture below.



There is a gasoline and diesel fueling station installed.at the left and right side of the park.
An automatic fueling robot carries the fuel from station and fill up the parked car with fuel.
The cars are divided into 2 types depending on whether it is a gasoline or diesel car.
1 is denoted as gasoline cars and 2 is denoted as diesel cars.

The automatic robot will be used to provide a cost free fueling service which is filling up all cars with 1 litre of each corresponding fuel.
The robot will move in between the 2 fuelling stations as below :

1) The robot carries 2 litre of gasoline at the gasoline station and starts moving from there.
2) The robot can fill up the cars of the same type of gas it carries 1 litre each.
3) The robot can go back to the fuelling station at any time, Independent from the current amount of fuel it carries.
4) When the robot arrives at the fuelling station, it gets 2 litre of supply of the corresponding fuel. (If the robot has some remaining fuel it will be discarded).

Problem statement:

5) There is an equal distance of 1 between each fueling station and the cars.

The fuel type of N Cars parked in the parking lot will be given.
Find the minimum moving distance of the automated fueling robot after it has filled up all the cars with 1 litre of fuel each.

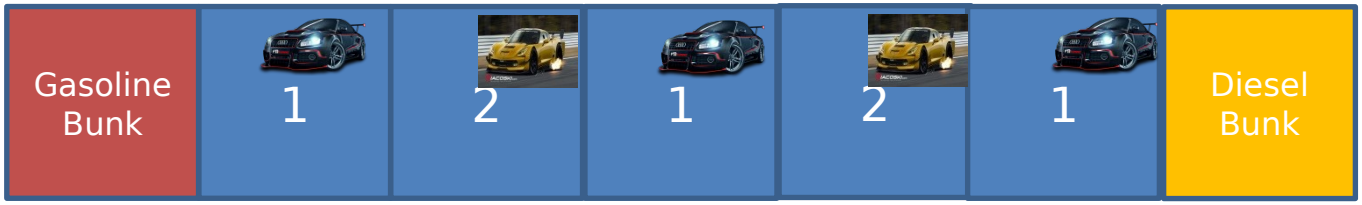
Time limit: C/C++/Java: 3 seconds.
Test cases: 50
2<=N<=8
I/P format:
2 ✉ Total number of test cases
5 ✉ N(Number of cars between gasoline and Diesel stations)
1 2 1 2 1(1✉ Gasoline car, 2✉Diesel cars)
5
2 1 1 2 1

O/P:
#1 12
#2 14

Example 1) Given the total number of cars N = 5 and the order of the parked cars such as G - D - G - D - G (PS: G-> Gasoline, D->Diesel)
the process of finding the minimum moving distance for fueling the car is as follows :

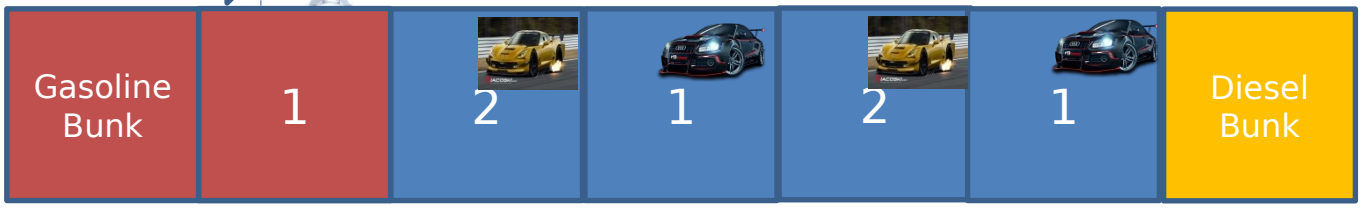
Initial positions of cars and Robot

Example:1

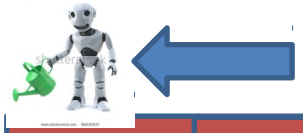


Initial position of cars and robot

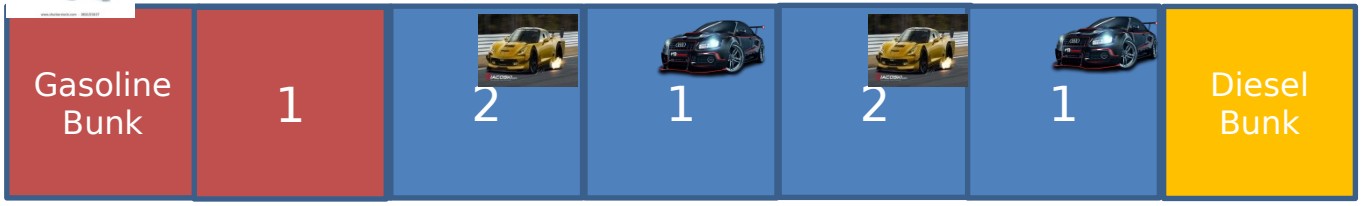
MOVE-1



Distance covered **=1** (gasoline bunk to 1st car)



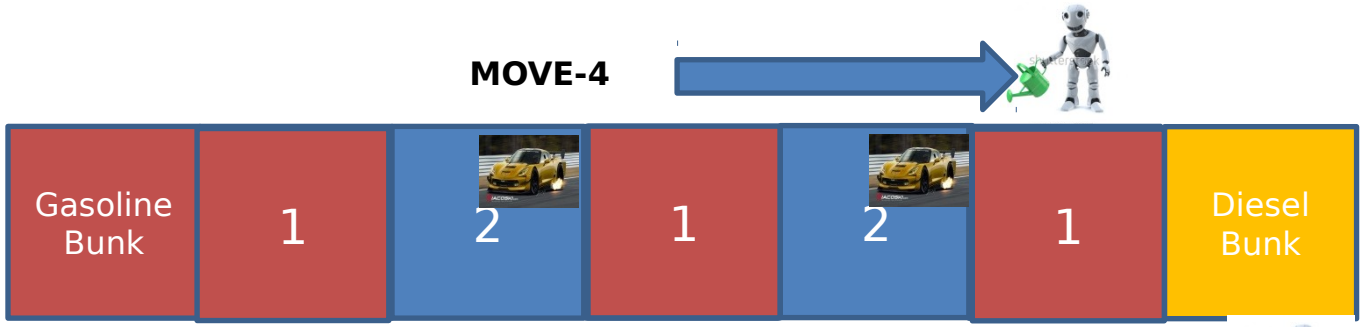
MOVE-2



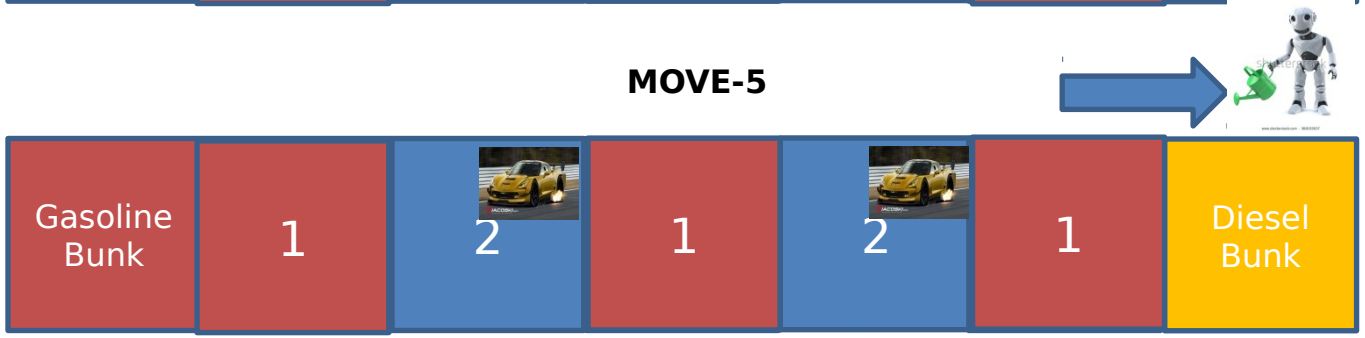
Distance covered **=1+1 = 2** (1st car to Gasoline bunk)

MOVE-3

Distance covered **=2+3 =5** (Gasoline bunk to 3rd car)

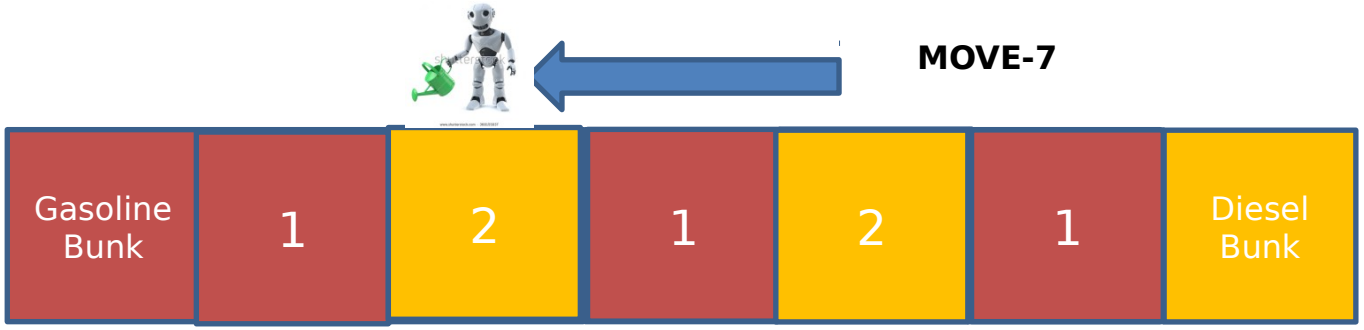


Distance covered **5+2 = 7** (3rd car to 5th car)

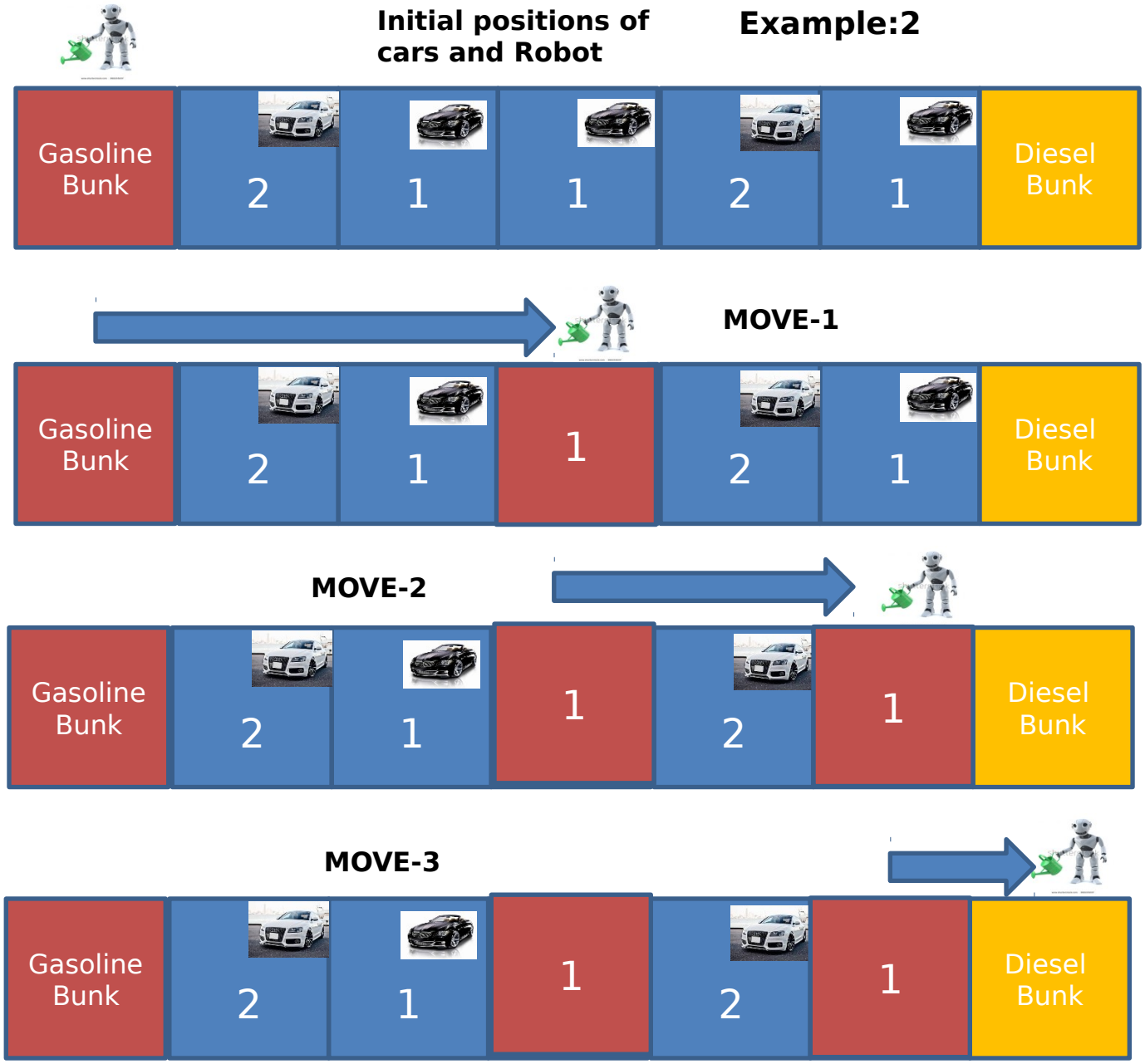


Distance covered **7+1=8** (5th car to Diesel bunk)

Distance covered **8+2=10** (Diesel bunk to 4th car)



Distance covered **10 +2=12** (4th car to 2nd car). So **12** is the shortest distance among all the available options possible



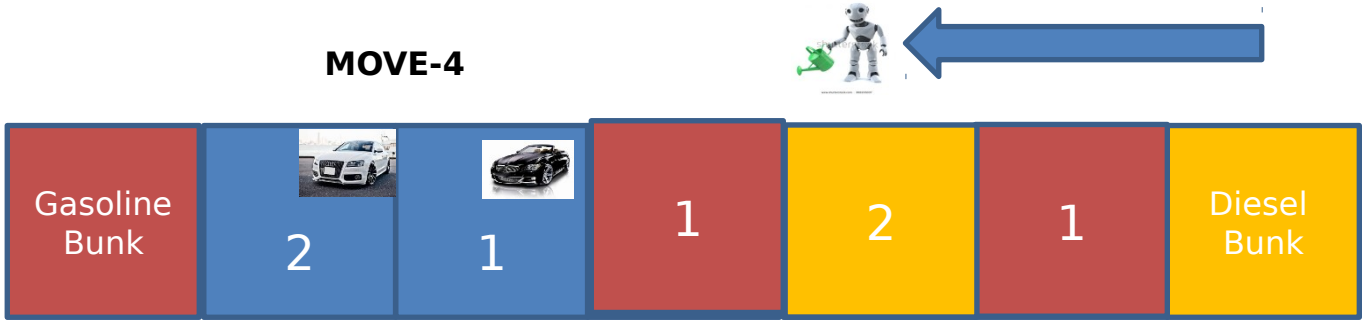
Initial position of cars and robot

Distance covered **=3** (gasoline bunk to 3rd car)

Distance covered **=3+2=5** (3rd car to 5th car)

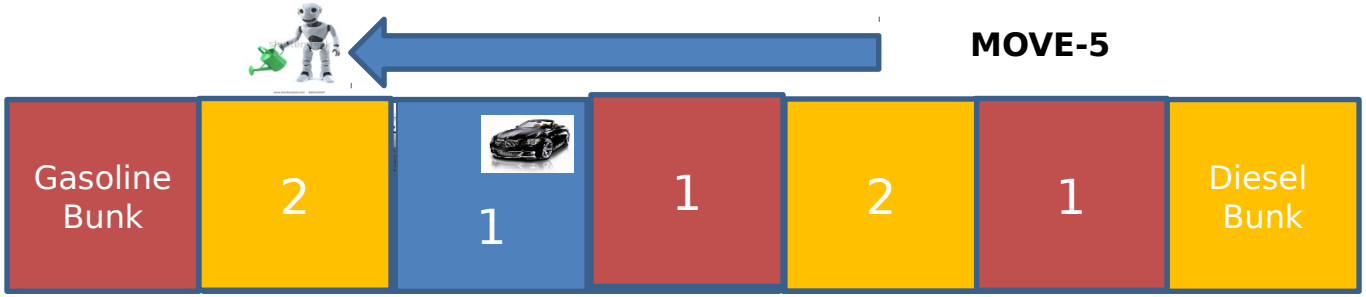
Distance covered **=5+1=6** (5th car to Diesel bunk). So **14** is the shortest distance among all the available options possible

MOVE-4



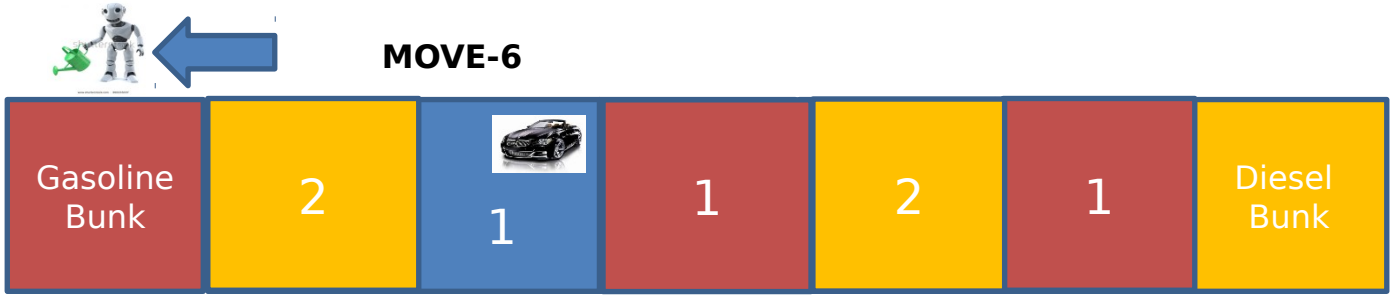
Distance covered **=6+2=8** (Diesel bunk to 4th car)

MOVE-5



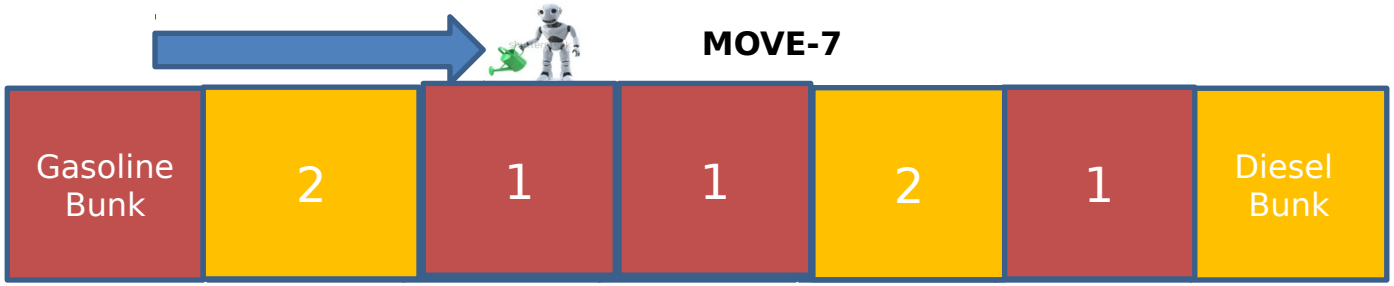
Distance covered **=8+3=11** (4th car to 1st car)

MOVE-6



Distance covered **=11+1=12** (1st car to gasoline bunk)

MOVE-7



Distance covered **=12+2=14** (gasoline bunk to 2nd car)

Approach-1:

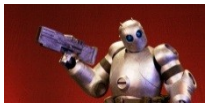
- We should always start from the Gasoline station.
- Once we fuel any car, we have 3 options to perform.
  1) Fuel next car(gasoline or Diesel car, with all the combinations)
  2) Go to Gasoline station and start refueling
  3) Go to Diesel station and start fueling.

Keep updating the distances as we move, once all cars are over, store result in global variable , if we find optimal distance with the current combination than the previous combinations.

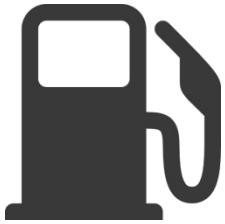
Solution is attached:

robo_fueling.cpp

Approach-2:



Robot has Two Functions in the problem statement



Robot @ Pump
Fill 2 Units of Fuel
Move in next direction
Increment count

Robot @ Car
If fuel carried by robot and car not same
    increment count
If fuel carried by robot and car is same same
3 decisions
    Don’t fill Fuel, continue to next car
    Fill the fuel and continue to next car
    Fill the fuel and continue backward
Increment count

Pseudo Algorithm

If it is a Pump
    Fill 2 Units of Fuel
    Move in next direction(Gasoline right/Diesel left)
    Increment count
If it is a Car
    If fuel carried by robot and car not same
        increment count
        move next
    If fuel carried by robot and car is same and empty
        if this is last car
            note the count
            return
        Don’t fill Fuel, continue to next spot
        Fill the fuel and continue to next spot
        Fill the fuel and continue backward
Solution is attached:

robo_fueling_rohit.cpp