**Lab Exercise #2**

**Implementation of Agents**

**RA1811030010045**

**DIVYANSH SINHA**

**Title: CHECK IF DOOR IS OPEN OR CLOSED (AUTOMATIC DOOR SYSTEM)**

**Problem Description:**

Given n doors and n persons. The doors are numbered 1 to n and persons are given id’s numbered 1 to n. Each door can have only 2 status open and closed. Initially all the doors have status closed. Find the final status of all the doors if a person changes the current status of all the doors, i.e. if status open then change to status closed and vice versa, for which he is authorized. A person with id ‘i’ is authorized to change the status of door numbered ‘j’ if ‘j’ is a multiple of ‘i’.

Note:

– A person has to change the current status of all the doors for which he is authorized exactly once.

– There can be a situation that before a person changes the status of the door, another person who is also authorized for the same door changes the status of the door.

**Example:**

Input : 3

Output : open closed closed

**Solution:** As n = 3, therefore there are

3 doors {1, 2, 3} and

3 persons with ids {1, 2, 3}

person with id = 1 can change the status of door 1, 2, 3

person with id = 2 can change the status of door 2

person with id = 3 can change the status of door 3

Current status of all doors: closed closed closed

Consider a sequence of events,

1. Person with id = 1 changes status of door 2  
   Current status of all doors: closed open closed
2. Person with id = 3 changes status of door 3  
   Current status of all doors: closed open open
3. Person with id = 1 changes status of door 1, 3  
   Current status of all doors: open open closed
4. Person with id = 2 changes status of door 2  
   Current status of all doors: open closed closed

In this sequence, all persons have changed the current status of the all doors exactly once for which they are authorized.

**Python Code:**

# Python 3 implementation of

# doors open or closed

import math

# Function to check whether

# 'n' has even number of

# factors or not

def hasEvenNumberOfFactors(n):

root\_n = math.sqrt(n)

# if 'n' is a perfect square

# it has odd number of factors

if ((root\_n \* root\_n) == n):

return False

# else 'n' has even

# number of factors

return True

# Function to find and print

# status of each door

def printStatusOfDoors(n):

for i in range(1, n + 1):

# If even number of factors

# final status is closed

if (hasEvenNumberOfFactors(i) == True):

print("closed", end =" ")

# else odd number of factors

# final status is open

else:

print("open", end =" ")

# Driver program

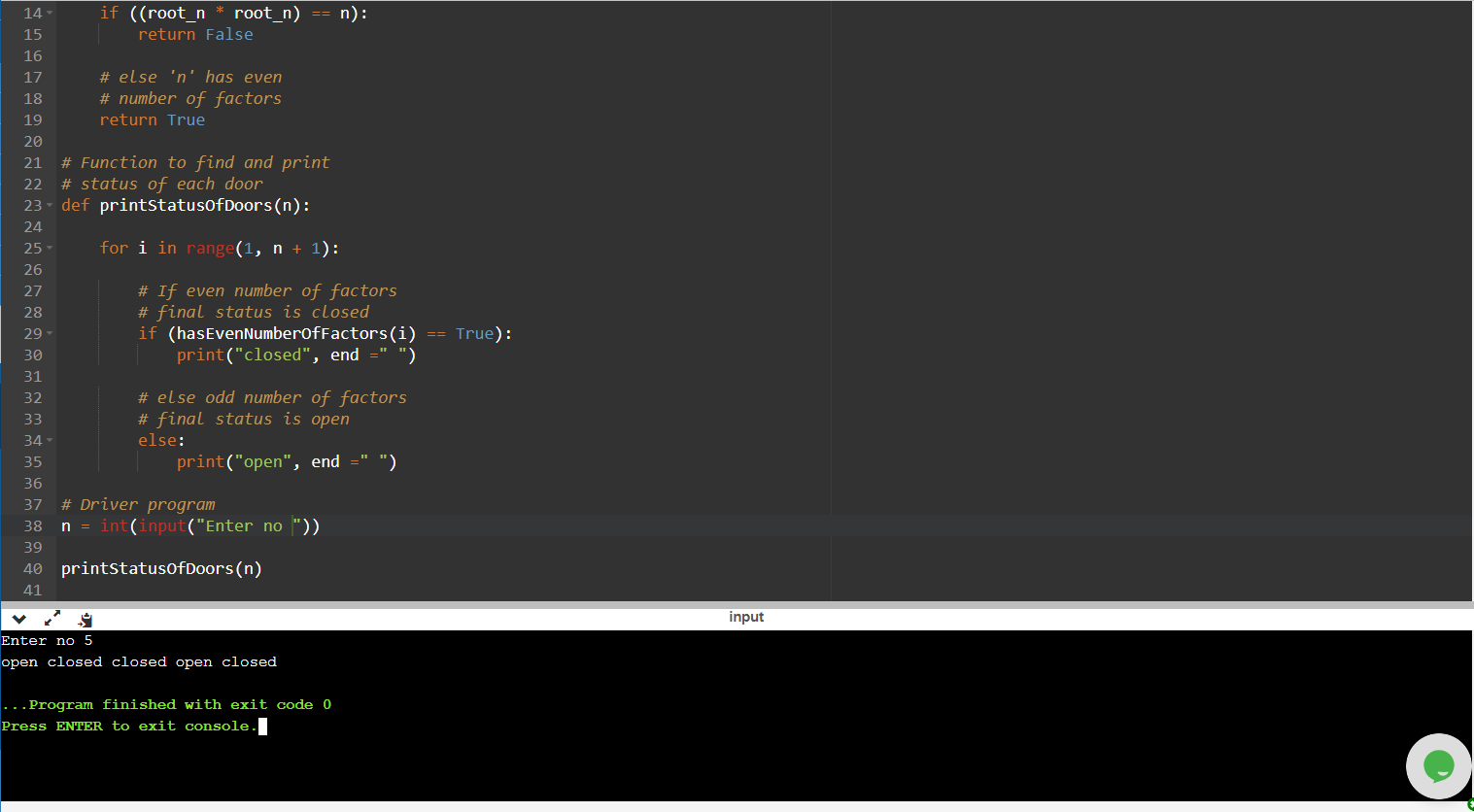
n = 5

printStatusOfDoors(n)

**Input and output:**

Input : 3

Output : open closed closed



**Result: Door opening and closing is implemented using a simple agent(if else).**