Experiment 2.1

Stacks

Student Name: Rajdeep Jaiswal
UID: 20BCS2761
Branch: BE-CSE
Section/Group: WM-902/B
Semester: 5th Sem
Date of Performance: 22nd Aug,2022
Subject Name: Competitive Programming - I
Subject Code: 20CSP-314

1. Aim/Overview of the practical:

You have three stacks of cylinders where each cylinder has the same diameter, but they may vary in height. You can change the height of a stack by removing and discarding its topmost cylinder any number of times. Find the maximum possible height of the stacks such that all of the stacks are exactly the same height. This means you must remove zero or more cylinders from the top of zero or more of the three stacks until they are all the same height, then return the height.

Example: H1 = [1,2,1,1]
H2 = [1,1,2]
H3 = [1,1]

There are 4, 3 and 2 cylinders in the three stacks, with their heights in the three arrays. Remove the top 2 cylinders from H1(heights = [1, 2]) and from H2 (heights = [1, 1]) so that the three stacks all are 2 units tall. Return 2 as the answer.

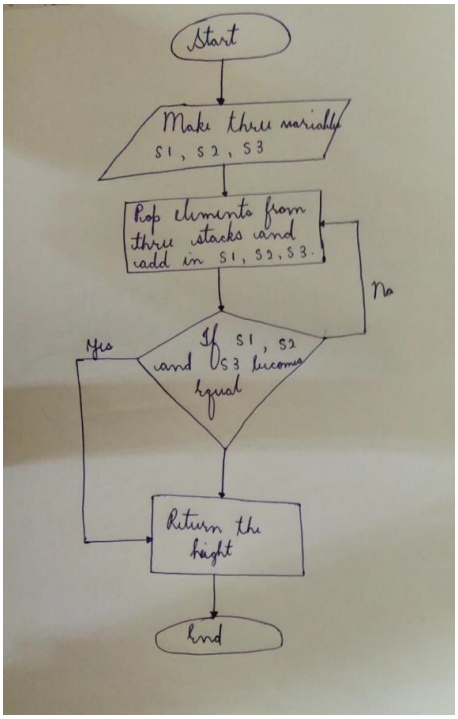
Note: An empty stack is still a stack.

2. Task to be done/ Which logistics used:

- Using stacks push and pop operation.
- By creating the sum variable.

3. Algorithm/Flowchart (For programming-based labs):

a.) Given various stacks s1, s2 and s3.
b.) Popping the elements and then adding it in variables.
c.) When the sum becomes equal then return the sum.



4. Steps In the Experiment/Code:

```
def read_stack():
    stack = [int(x) for x in input().split(' ')]
stack = list(reversed(stack))     sum_stack =
set()     psum = 0     for i in
range(len(stack)):     psum += stack[i]
sum_stack.add(psum)
    return sum_stack

input() ans =
read_stack() ans &=
read_stack() ans &=
read_stack() if
len(ans) > 0:
print(max(ans))
else:
```

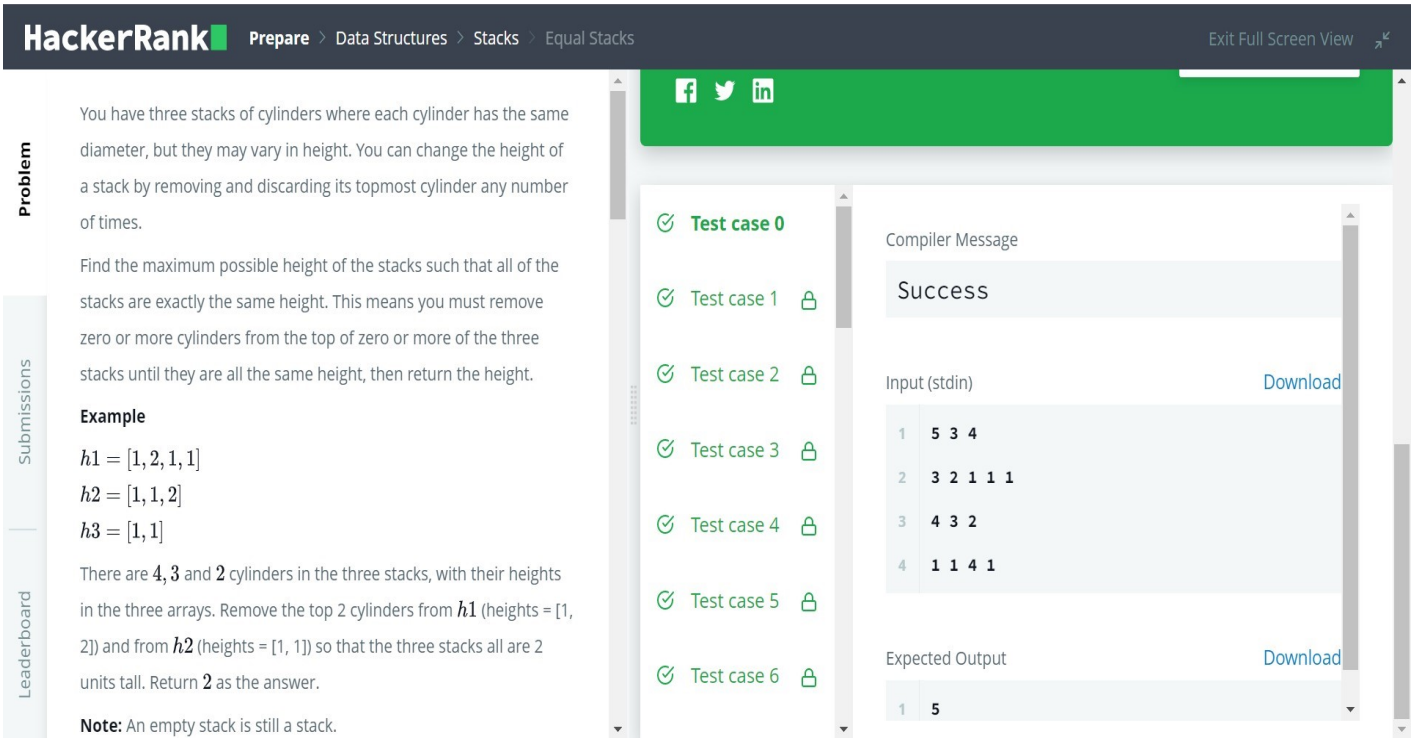
```
print(0)
```

5. Observations/Discussions/ Complexity Analysis:

Time Complexity = O(1)

Because the operations in stacks like push, pop is having linear time complexity (constant time complexity).

6. Result/Output/Writing Summary:



7. Learning outcomes (What I have learnt):

a. Learnt about stacks.

b. Got an overview of the type of questions on hacker-rank.

c. Get to know about crucial test cases.

d. Got an understanding about stack's operations

e. Learnt about using various pages and insights of hacker-rank.

Experiment 2.2

Queues

Student Name: Rajdeep Jaiswal
UID: 20BCS2761
Branch: BE-CSE
Section/Group: WM-902/B
Semester: 5th Sem
Date of Performance: 22nd Aug,2022
Subject Name: Competitive Programming - I
Subject Code: 20CSP-314

1. Aim/Overview of the practical:

Suppose there is a circle. There are N petrol pumps on that circle. Petrol pumps are numbered 0 to (N-1) (both inclusive). You have two pieces of information corresponding to each of the petrol pump: (1) the amount of petrol that particular petrol pump will give, and (2) the distance from that petrol pump to the next petrol pump.

Initially, you have a tank of infinite capacity carrying no petrol. You can start the tour at any of the petrol pumps. Calculate the first point from where the truck will be able to complete the circle. Consider that the truck will stop at each of the petrol pumps. The truck will move one kilometer for each litre of the petrol.

Example:

Input: 3
1 5
10 3
3 4

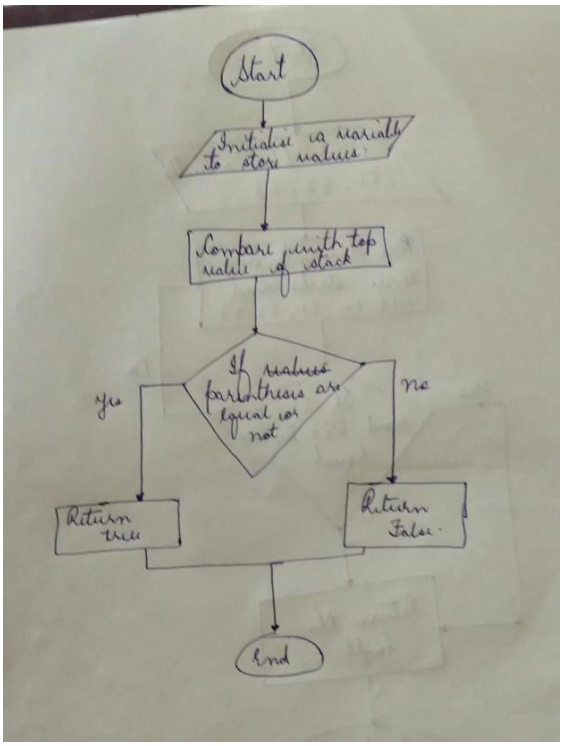
Output: 1

Note: We can start the tour from the second petrol pump.

2. Task to be done/ Which logistics used:

Approach: Usage of queues as storage of carrying petrol in a variable.

3. Algorithm/Flowchart (For programming-based labs):



4. Steps for experiment/practical/Code:

```
num=int(input()) pet=[]
dist=[] for line in
range(num):
i=input().split(" ")
pet.append(int(i[0]))
dist.append(int(i[1])) bal=[]
for i in range(num):
bal.append(pet[i]-dist[i])

small=0 for strt in
range(num):     s=bal[strt]
i=(strt+1)%num
while(s>=0 and i!=strt):
        s+=bal[i]
```

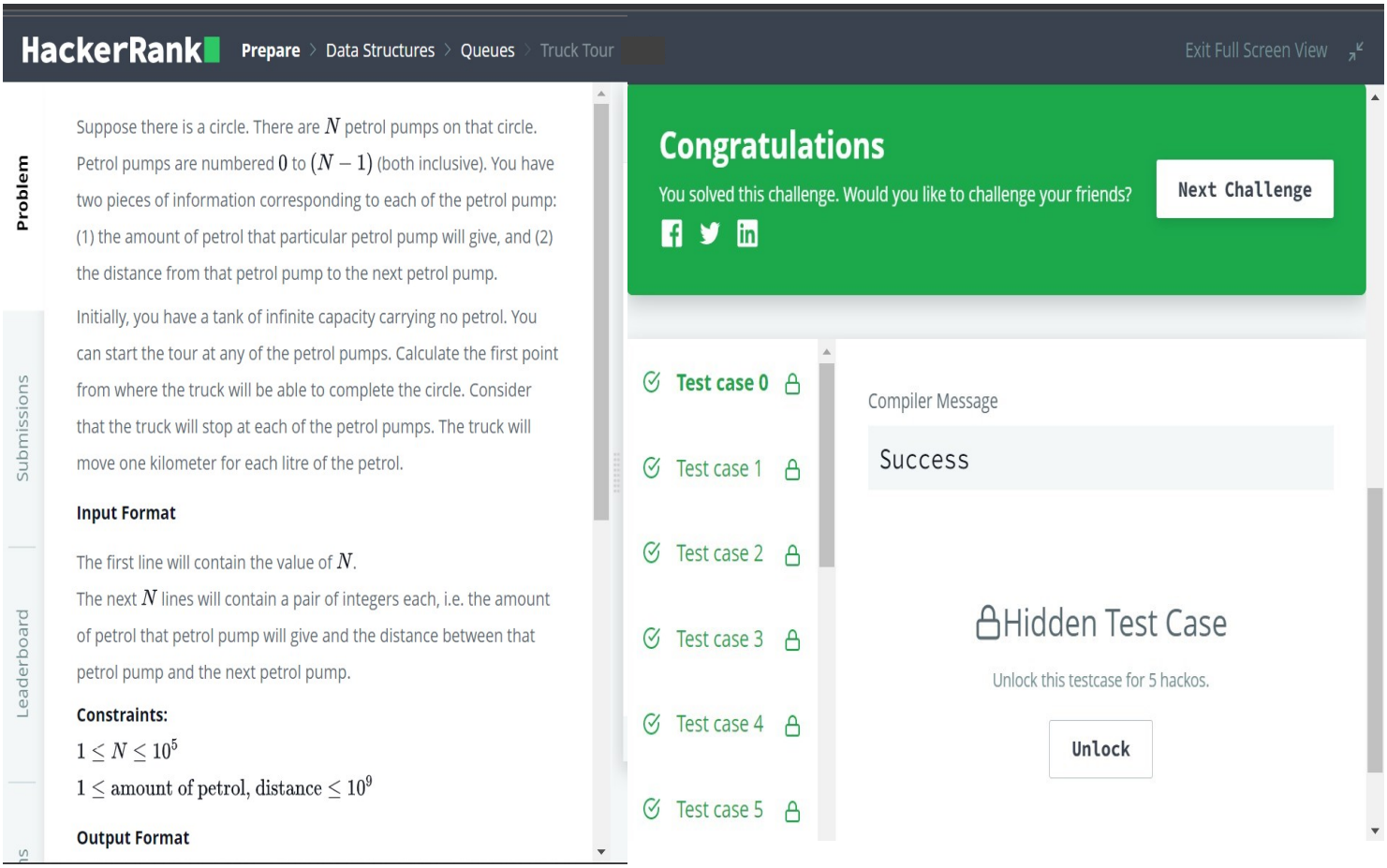
```
            i=(i+1)%num
        if(i==strt):
small=strt      break
print(small)
```

5. Observations/Discussions/ Complexity Analysis:

Time complexity is **O(N).**

Space complexity is **O(N).**

6. Result/Output/Writing Summary:



7. Learning outcomes (What I have learnt):

a. Learnt about stacks.

b. Got an overview of the type of questions on hacker-rank.

c. Get to know about crucial test cases.

d. Got an understanding about stack's operations

e. Learnt about using various pages and insights of hacker-rank.