

Day-9 Time: 1 hour 30 min. (12:00-01:30) Date:- 29-07-2024

Q 1.)

You have a car with a gas tank of infinite capacity. There are 'N' gas stations along a circular route. Gas stations are numbered from 0 to N - 1. You begin the journey with an empty tank at one of the gas stations. You want to travel around the circular route once in the clockwise direction. I.e if you start to travel from station 'i', then you will go to i + 1, i + 2, ..., n - 1, 0, 1, 2, i - 1 and then return back to 'i'.

You are given two integer arrays 'gas' and 'cost'. 'gas[i]' gives the amount of gas available at station 'i' and cost[i] gives the amount of gas required to travel from station 'i' to next station i.e station 'i'+1 (or 0 if the station is N - 1).

Return the starting gas station's index if it is possible to complete cycle of given circular route once in the clockwise direction. If there are multiple possible starting gas station's indexes, then return the minimum of those gas station indexes. If there is no such gas station then return -1.

Input Format :

The first line of input contains a single integer T, representing the number of test cases or queries to be run, then the T test cases follow.

The first line of each test case contains a positive integer 'N' which represents the number of gas stations.

The second line of each test case contains 'N' space-separated integers representing the integer array 'gas'.

The third line of each test case contains 'N' space-separated integers representing the integer array 'cost'.

Output Format :

For each test case, print a single integer denoting the minimum index of the starting gas station if you are able to travel around the cycle once, otherwise print -1.

Constraint :

1 <= T <=50

1 <= N <= 10 ^ 4

0 <= GAS[i] <= 10 ^ 5

0 <= COST[i] <= 10 ^ 5

Where GAS[i] represents the ith element of 'GAS' array,
COST[i] represents the ith element of 'COST' array.

Sample Input 1 :

```
1
2
1 2
2 1
```

Sample Output 1:

```
1
```

Explanation of Sample Input 1 :

Test Case 1:

If you start from index 0, you can fill in 1 unit of gas. Now your tank has 1 unit of gas. But you need 2 units of gas to travel to station 1. Thus you can not start at station 0.

If you start from index 1, you can fill in 2 units of gas. Now your tank has 2 units of gas. You need 1 unit of gas to get to station 0. So, you travel to station 0 and still have 1 unit of gas left. You fill in 1 unit of additional gas, making your current gas = 2 unit. It costs you 2 amounts of gas to get to station 1, which you do and complete the cycle. Thus you can start at index 1.

Sample Input 2 :

```
1
5
1 2 3 4 5
3 4 5 1 2
```

Sample Output 2:

```
3
```

Explanation of Sample Input 2 :

Test Case 1:

If you start from index 3 and fill all available gas at each station, then you can reach station 4 with 3 units of gas, station 0 with 6 units of gas, station 1 with 4 units of gas, station 2 with 2 units of gas, and back to station 3 after consuming all the gas.

We can show that index 3 is the minimum index of the gas station from where the complete circular trip is possible

Question no 2.

ou are given a set of 'N' rectangular envelopes. The height and width of each envelope are given by arrays, 'height' and 'width' respectively, each consisting of 'N' positive integers. The height, width of the ith envelope is given by 'height[i]' and 'width[i]' respectively.

You can put one envelope inside another envelope if and only if both the height and width of one envelope is strictly greater than the height and width of the other envelope.

What is the maximum number of envelopes you can Russian doll? (put one inside other)

Note

Rotation of the envelope is not allowed, that is, height and width can't be exchanged

Input Format:

The first line of input contains an integer 'T' denoting the number of test cases.

The next 3*T lines represent the 'T' test cases.

The first line of each test case contains an integer 'N', representing the number of envelopes.

The second line of the test case contains 'N' space-separated integers representing elements of the array 'height'. The third line of the test case contains 'N' space-separated integers representing elements of the array 'width'.

Output Format :

For each test case, print, in a separate line, the maximum number of envelopes you can Russian doll.

Constraints:

$1 \leq T \leq 50$

$1 \leq n \leq 10^4$

$1 \leq \text{height}[i] \leq 10^9$

$1 \leq \text{width}[i] \leq 10^9$

Sample Input 1:

```
2
4
5 6 6 2
4 4 7 3
2
2 1
2 1
```

Sample Output 1:

```
3
2
```

Explanation For Sample Output 1:

Test Case 1:

The number envelopes, 'N' = 4

'height' = {5, 6, 6, 2}

'width' = {4, 4, 7, 3}

Let denote dimensions of the envelope in (Height, Width) manner then, one way of Russian Doll envelopes in outermost to the innermost manner is as follow:

Select the third envelope, i.e., envelope with dimensions (6, 7) as the outermost envelope.

Place the first envelope i.e envelope with dimensions (5, 4) inside the outermost envelope. You can do this because both the height and width of this envelope is strictly less than the outermost envelope.

Place the fourth envelope i.e envelope with dimensions (2, 3) inside the previous envelope.

In this way, we can Russian Doll 3 envelopes.

No other way can Russian Doll more than 3 envelopes.

Test Case 2:

You can put the second envelope inside the first envelope because both the height and width of the second envelope are strictly less than the first envelope.

Sample Input 2:

```
2
1
2
3
3
1 1 1
1 1 1
```

Sample Output 2:

```
1
1
```

Question no 3.)

Implement heap sort.

Also, write the time and space complexity of all sorting algorithms in the multi-comment line.