MNC COMPANIES - SET 001

Solved Challenges **5/10**

Back To Challenges List

function findSequence

ID:12058 **Solved By 46 Users**

Accenture

You are required to complete the given code. You can click on Run anytime to check the compilation/execution status of the program. You can use printf to debug your code. The submitted code should be logically/syntactically correct and pass all test cases. Do not write the **main()** function as it is not required.

Code Approach: For this question, you will need to complete the code as in the given implementation. We do not expect you to modify the approach.

The function/method **findSequence** accepts two arguments - **SIZE** and **arr**, an integer representing the size of the integer array arr and the integer array arr.

The function/method **findSequence** must return **1** if it is possible to split the array arr into two sets in such a way that the first set is strictly decreasing while the second set is strictly increasing. Else the function must return **0**.

Your task is to complete the code in the function **findSequence** so that it passes all the test cases.

Boundary Condition(s):
4 <= N <= 100
-10^5 <= Each integer value <= 10^5

Example Input/Output 1:
Input:
7
4 2 -1 0 1 2 5

Output:
1

Explanation:
There are two possible ways to split the array which are given below.
(**4 2 -1**) and (**0 1 2 5**)
(**4 2**) and (**-1 0 1 2 5**)

Example Input/Output 2:
Input:
6
4 1 9 10 12 11

Output:
0

Max Execution Time Limit: 50 millisecs

Ambiance

C (gcc 8.x)

Reset

```
int findSequence(int SIZE, int *arr)
{
    int stop=0;
    int index=0;
    for(index=1;index<SIZE;index++)
    {
        if(arr[index]<arr[index-1])
            continue;
        else
        {
            stop = index;
            break;
        }
    }
    if(stop==1)
        return(0);
    for(index=stop;index<SIZE;index++)
    {
        if(arr[index]>arr[index-1])
            continue;
        else
            break;
    }
    if(index==SIZE)
        return(1);
    else
        return(0);
}
```

Please wait while we run the program ....

Run with a custom test case (Input/Output)