### Subarray with given sum(6 times asked)

Given an unsorted array **A**of size **N** that contains only positive integers, find a continuous sub-array that adds to a given number **S**and return the left and right index(**1-based indexing**) of that subarray.

In case of multiple subarrays, return the subarray indexes which come first on moving from left to right.

**Note**:- You have to return an ArrayList consisting of two elements left and right. In case no such subarray exists return an array consisting of element **-1**.

**Example 1:**

**Input:**

N = 5, S = 12

A[] = {1,2,3,7,5}

**Output:** 2 4

**Explanation:** The sum of elements

from 2nd position to 4th position

is 12.

**Example 2:**

**Input:**

N = 10, S = 15

A[] = {1,2,3,4,5,6,7,8,9,10}

**Output:** 1 5

**Explanation:** The sum of elements

from 1st position to 5th position

is 15.

### Java code

//{ Driver Code Starts

import java.util.\*;

import java.lang.\*;

import java.io.\*;

class Main{

static BufferedReader br;

static PrintWriter ot;

public static void main(String[] args) throws IOException{

br = new BufferedReader(new InputStreamReader(System.in));

ot = new PrintWriter(System.out);

int t = Integer.parseInt(br.readLine());

while(t-->0){

String s[] = br.readLine().trim().split(" ");

int n = Integer.parseInt(s[0]);

int k = Integer.parseInt(s[1]);

int a[] = new int[n];

s = br.readLine().trim().split(" ");

for(int i = 0; i < n; i++)

a[i] = Integer.parseInt(s[i]);

Solution obj = new Solution();

ArrayList<Integer> res = obj.subarraySum(a, n, k);

for(int ii = 0;ii<res.size();ii++)

ot.print(res.get(ii) + " ");

ot.println();

}

ot.close();

}

}

// } Driver Code Ends

class Solution

{

//Function to find a continuous sub-array which adds up to a given number.

static ArrayList< Integer> subarraySum(int[] arr, int n, int s)

{

int c = 0;

ArrayList<Integer> list = new ArrayList<Integer>();

if(s==0){

list.add(-1);

return list;

}

int j=0,sum=0;

for(int i=0;i<n;i++){

sum +=arr[i];

if(sum>=s){

while(sum>s &&j<i){

sum-=arr[j];

j++ ;

}

if(sum==s){

c=1;

list.add(j+1);

list.add(i+1);

break;

}

}

}

if(c==0)

{

list.add(-1);

return list;

}

return list;

}

}