**Subarray with given sum**

[array](http://www.practice.geeksforgeeks.org/tag-page.php?tag=array&isCmp=0)[Google](http://www.practice.geeksforgeeks.org/tag-page.php?tag=Google&isCmp=1)

Given an unsorted array of non-negative integers, find a continous subarray which adds to a given number.

**Input:**

The first line of input contains an integer T denoting the number of test cases.  
The first line of each test case is N and S, N is the size of array and S in sum.  
The second line of each test case contains N input C[i].  
  
**Output:**

Print the first such occuring subarray a[i] and a[j] if sum equal to subarray else print -1.  
  
**Constraints:**

1 ≤ T ≤ 50  
1 ≤ N ≤ 100  
1 ≤ C[i] ≤ 200  
  
**Example:**

**Input:**  
2  
5 12  
1 2 3 7 5  
10 15  
1 2 3 4 5 6 7 8 9 10

**Output:**  
2 4  
1 5

**Explanation :**  
In first test case, sum of elements from 2nd position to 4th position is 12

In second test case, sum of elements from 1st position to 5th position is 15

\*\*For More Examples Use Expected Output\*\*

<http://www.practice.geeksforgeeks.org/problem-page.php?pid=590>

#include <iostream>

#include <stdio.h>

using namespace std;

int main() {

    int t;

    scanf("%d", &t);

    while(t--) {

        int n,s;

        scanf("%d %d", &n,&s);

        int c[n];

        for(int i =0; i<n; i++) {

            scanf("%d", &c[i]);

        }

        bool res = false;

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

            int sum =0;

            for(int j = i; j<n && !res; j++) {

                sum +=c[j];

                if(sum == s) {

                    printf("%d %d", i + 1,j + 1);

                    res = true;

                }

            }

        }

        if(!res) {

           printf("-1");

        }

        printf("**\n**");

    }

 return 0;

}