Indexes of Subarray Sum

Difficulty: **Medium** Accuracy: **16.5%** Submissions: **1.7M** Points: **4**

Given an array **arr[]** containing only non-negative integers, your task is to find a continuous subarray (a contiguous sequence of elements) whose sum equals a specified value **target**. You need to return the **1-based indices** of the leftmost and rightmost elements of this subarray. You need to find the first subarray whose sum is equal to the target.

Note: If no such array is possible then, return [-1].

Examples:

Input: arr[] = [1, 2, 3, 7, 5], target = 12
Output: [2, 4]
Explanation: The sum of elements from 2nd to 4th position is 12.

Input: arr[] = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10], target = 15
Output: [1, 5]
Explanation: The sum of elements from 1st to 5th position is 15.

Input: arr[] = [5, 3, 4], target = 2

```java
// ) Driver Code Ends
class Solution {
    static ArrayList<Integer> subarraySum(int[] arr, int target) {
        ArrayList<Integer> list = new ArrayList<>();
        int j=0; int sum=0;
        for(int i=0;i<arr.length;i++){
            sum+=arr[i];
            while(sum>target){
                sum-=arr[j];
                j++;
            }
            if(sum==target){
                list.add(j+1);
                list.add(i+1);
                return list;
            }
        }
        list.add(-1);
        return list;
    }
}
```