1. Given an array of intergers, find two numbers such that they add up to a specific target number. The function twoSum should return indices of the two numbers such that they add up to the target, where index1 must be less than index2 Please note that your returned answers (both index1 and index2) are not zero-based.

You may assume that each input would have exactly one solution.

Input: numbers={2, 7, 11, 15}, target=9 Output: index1=1, index2=2

题目翻译： 这道题目的意思是给定一个数组和一个值，让求出这个数组中两个值的和等于这个给定值的坐标。输出是有要求的，1， 坐标较小的放在前面，较大的放在后面。2， 这俩坐标不能为零。

如果存在的话，返回如果不存在的话，继续查找: myMap[target-numbers[i]]

```
class Solution(object):
    def twoSum(self, nums, target):
        if len(nums)<=1:
            return False
        myMap={}
        for i in range(len(nums)):
            if    nums[i] in myMap:
                return [myMap[nums[i]],i]
            else:
                myMap[target- nums[i]]=i
```

167. Two Sum II - Input array is sorted

Given an array of integers that is already sorted in ascending order, find two numbers such that they add up to a specific target number.

The function twoSum should return indices of the two numbers such that they add up to the target, where index1 must be less than index2. Please note that your returned answers (both index1 and index2) are not zero-based.

You may assume that each input would have exactly one solution and you may not use the same element twice.

Input: numbers={2, 7, 11, 15}, target=9

Output: index1=1, index2=2

题目翻译：一个整型的数组，是递增序列。求出这个数组中两个值的和等于这个给定值的坐标。

```
class Solution(object):
    def twoSum(self, numbers, target):
        if len(numbers)<=1:
            return False
        myMap={}
        for i in range(len(numbers)):
            if    numbers[i] in myMap:
                return [myMap[numbers[i]],i+1]
            else:
                myMap[target- numbers[i]]=i+1
```