# 和为S的两个数字

## 题目

[牛客网](https://www.nowcoder.com/practice/390da4f7a00f44bea7c2f3d19491311b?tpId=13&tqId=11195&tPage=3&rp=2&ru=%2Fta%2Fcoding-interviews&qru=%2Fta%2Fcoding-interviews%2Fquestion-ranking)

输入一个递增排序的数组和一个数字S，在数组中查找两个数，使得他们的和正好是S，如果有多对数字的和等于S，输出两个数的乘积最小的。

对应每个测试案例，输出两个数，小的先输出。

## 解题思路

1. 利用二分查找的思想，由于是排序数组，通过两个指针来进行遍历

public ArrayList<Integer> FindNumbersWithSum(int[] array, int sum) {
 ArrayList<Integer> res = new ArrayList<>();

 if (array == null || array.length == 1) {
 return res;
 }

 int start = 0, end = array.length - 1;
 int minMulti = Integer.MAX\_VALUE;
 int a = -1, b = -1;

 while (start < end) {
 int t = array[start] + array[end];
 if (t == sum) {
 int multi = array[start] \* array[end];
 if (multi < minMulti) {
 a = array[start];
 b = array[end];
 minMulti = multi;
 }
 start++;
 end--;
 } else if (t > sum) end--;
 else start++;
 }

 if (a == -1 || b == -1) {
 return res;
 }

 res.add(a);
 res.add(b);

 return res;
}