1. Let A[0 : n-1] be an array of n distinct numbers. If i<j (i and j are ordered indices) and A[i]>A[j], then the pair (a[i], a[j]) is called an out of ordered pair of an array A. Write a program to count the out of ordered pair(s) of a given array. Implement an O(nlogn) algorithm.
Input : A = {5, 9, 14, 25, 31, 45}
Output : 0
Input : A = {54044, 14108, 79294, 29649, 25260, 60660, 2995, 53777, 49689, 9083}
Output : 28
Input : A = {8, 7, 6, 5, 4, 3, 2, 1}
Output : 28
Input : A = {2, 4, 1, 3, 5}
Output : 3
Input: A = {1, 20, 6, 4, 5}
Output: 5
Input: A = {2, 3, 6, 9, 1}
Output: 4
Input : A={1,3,5,2,4,6}
Output : 3
Input : A = 4,3,5,2,1)
Output : 8

2. Suppose you are given an array A[0 . . . n-1] of integers, some of the integers may be negative, zero, containing at least one positive integer. A sub-array A[i, j] of A, where 0≤i ≤ j<n, is defined by the sequence A[i], A[i + 1], . . . , A[j]. Your task is to find the contiguous subarray (containing at least one integer) which has the largest sum. Write a program to implement an O(nlogn) time algorithm.
Input :
The given array is : {8, 3, 8, -5, 4, 3, -4, 3, 5}
Output :
The largest sum of contiguous subarray is : 25 (8+3+8+-5+4+3-4+3+5)
Input :
The given array is : {-2, 1, -3, 4, -1, 2, 1, -5, 4}
Output :
The largest sum of contiguous subarray is : 6 (4, -1, 2, 1)
Input :
The given array is : {3, -5, 3, 8, 2, -4, 9, -6,3,-2,-8,3,-5,1,7,-9}
Output :
The largest sum of contiguous subarray is : 18 (3, 8, 2, -4, 9)
Input :
The given array is : {-2,1,-3,4,-1,2,1,-5,4}
Output:
The largest sum of contiguous subarray is : 6 (4,-1,2,1)
Input :
The given array is : { 1, 2, 3, 4, -10}
Output:

The largest sum of contiguous subarray is : 10 (1,2,3,4)

Input :

The given array is : { -1, 2, 3, -4, 5, 10}

Output:

The largest sum of contiguous subarray is : 16 (2,3,-4,5,10)

3. You are given an unsorted array of n integers and an arbitrary integer k. Determine whether there is pair(s) of elements in the array that sums to exactly k. Array elements can be positive, negative, or zero and the pair should consist of two different array elements. Also count how many such pairs exist. For example, given the array [1, 3, 7] and k = 8, the answer is "yes," but given k = 6 the answer is "no." For example, if the given array is [3, 5, 2, -4, 8, 11] and k = 7, your program should return [[11, -4], [2, 5]] because 11 + -4 = 7 and 2 + 5 = 7. Write a program to implement an O(n log n) time algorithm to find all pair(s) with given sum in the array.

Input

The given array : {6, 8, 4, -5, 7, 9}

The given sum : 15

Output

(6, 9)

(8, 7)

Input

The given array : {-5, 1, -40, 20, 6, 8, 7}

The given sum : 15

Output

(7, 8)

(-5, 20)

Input

The given array : {-5, 4, -2, 16, 8, 9}

The given sum : 15

Output

There is no pair of elements whose sum is equal to 15

Input

The given array : {1, 5, 7, -1}

The given sum : 6

Output

(1, 5)

(7, -1)

Input

The given array : {1, 5, 7, -1, 5}

The given sum : 6

Output

(1, 5)

(7, -1)

(1, 5)

4. Given an unsorted array of integers containing duplicate elements. Your task is to find if there exist any integer x that appears more than n/2 times (if it exists) in an array A[0...n-1] of size n, otherwise prints “No such integer”(there is at most one such element). **You are not allowed to sort the array.** No array can have more than one such integer. Write a program that implement an O(nlogn) time algorithm.

Input :
The given array is : {1, 3, 3, 7, 4, 3, 2, 3, 3}
Output : 3
Input :
The given array is : {4, 8, 4, 6, 7, 4, 4, 8}
Output :
No such integer in the given array.
Input :
The given array is : {2, 3, 2, 3, 2, 3, 2, 3}
Output :
No such integer in the given array.
[count is = n/2 not > n/2, so neither 2 nor 3 is present more than n/2 times]