10910EECS204001  
Data Structures Homework 5

Due date: 2021/1/1 23:59

Submit to OJ: #13037

Upload code to iLMS

Submission

* Please **1)** submit your code to OJ (OJ: #13037),   
  and **2)** zip your code as studentID.zip and submit to iLMS HW5.(eg, 108062568.zip)
* **Both should be done before the due date.**

Description

In this homework, you are asked to implement 2 functions.

1. QQ Inversion pairs count

2. K’s largest number

QQ Inversion pair

Let **A** be a sequence of numbers. **(i, j)** is an QQ inversion pair if **i < j** but **A[i] > 2 \* A[j]**.

For example: **A = [1, 2, 3, 4]**, then **A** has no QQ inversion pair.

For example: **B = [5, 4, 3, 2,1],** **B** has 4 QQ inversion pairs:

**(5, 2), (5, 1), (4, 1), (3, 1).**

QQ Inversion pairs count

Output the number of inversion pairs

For example: **A = [9, 4, 5, 3]**, the output is 2.

Because **A** has inversion pairs: **(9, 4)**, **(9, 3)**.

K’s Largest Number

Output the k’s largest number of the input sequence

For example: **A = [179, 208, 306, 93, 859, 984, 55, 9, 271, 33]**, **K = 2,** the output is: 859

You are allowed to use STL.

But don’t use any STL related to sort.

Input

There are multiple test cases, and each test case begins with a line containing two integers **n** and **op**, the number of elements in sequence and the option.

1. If **op** = 0, calculate the QQ inversion pairs count.
2. Else, find the k’th largest number (k = **op**)

The next line includes the **n integers** in the sequence.

Please note:

(each value is decimal integer)

**There won’t be a new line at the end of the file.**

Output

The k’s largest element / QQ inversion pair count according to option

**Each output should followed by a new line**

Sample input

10 0

17 7 1 5 18 14 12 15 11 10

6 3

15 8 13 7 4 18

9 1

8 12 17 18 5 14 10 9 11

Sample output

4

13

18