10810EECS204001  
Data Structures Homework 5

Due date: 2019/12/23 23:59

Submit to OJ: #12554

Upload code to iLMS

Submission

* Please **1)** submit your code to OJ (OJ: #12554),   
  and **2)** upload the zipped file (source codes) to iLMs.   
  **Both should be done before the due date.**
* Scores will be given based on your OJ results, and the uploaded file (the source codes) should be identical to those submitted to OJ. TAs will examine your uploaded codes.

Description

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

1. Inversion pairs count

2. Radix sort(LSD) process

Inversion pair

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

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

For example: **B = [4, 3, 2, 1]**, **B** has 6 inversion pairs:

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

Inversion pairs count

Output the number of inversion pairs

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

Because **A** has inversion pairs: **(1, 2)**, **(1, 3)**, **(1, 4)**, **(2, 4)**, **(3, 4)**.

Radix sort(LSD) process

Output first element and last element of results in every pass of the radix

sort(LSD).

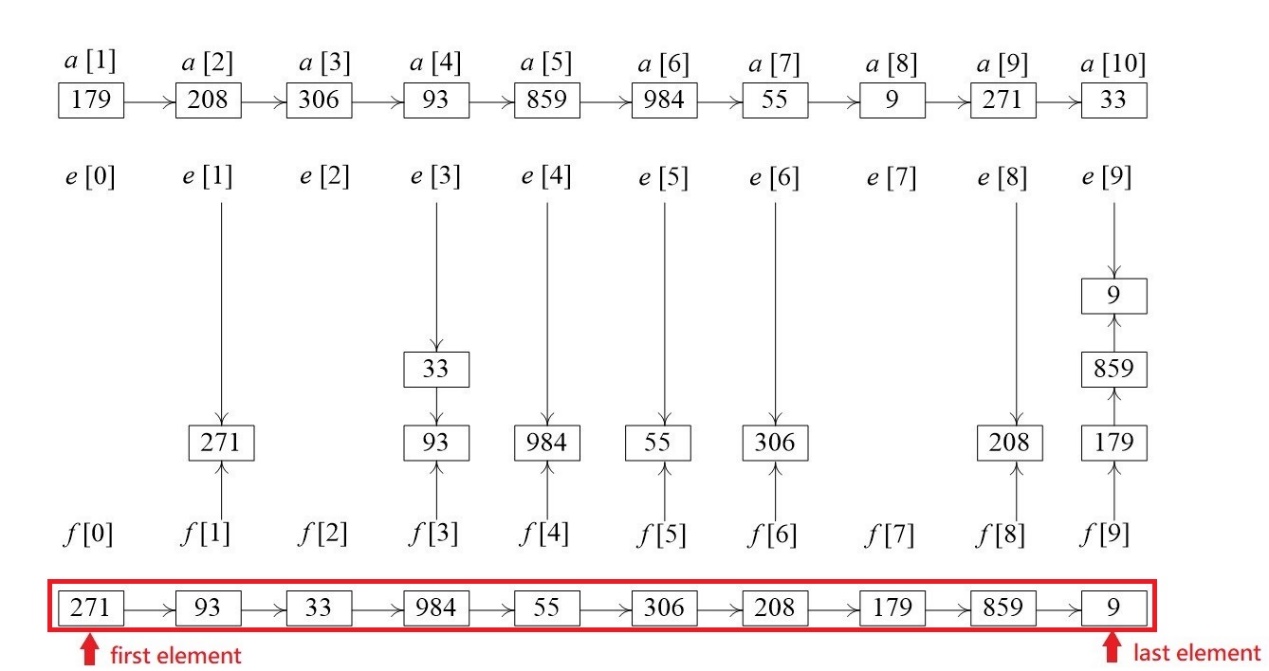
For example: **A = [179, 208, 306, 93, 859, 984, 55, 9, 271, 33]**

**r = 10**

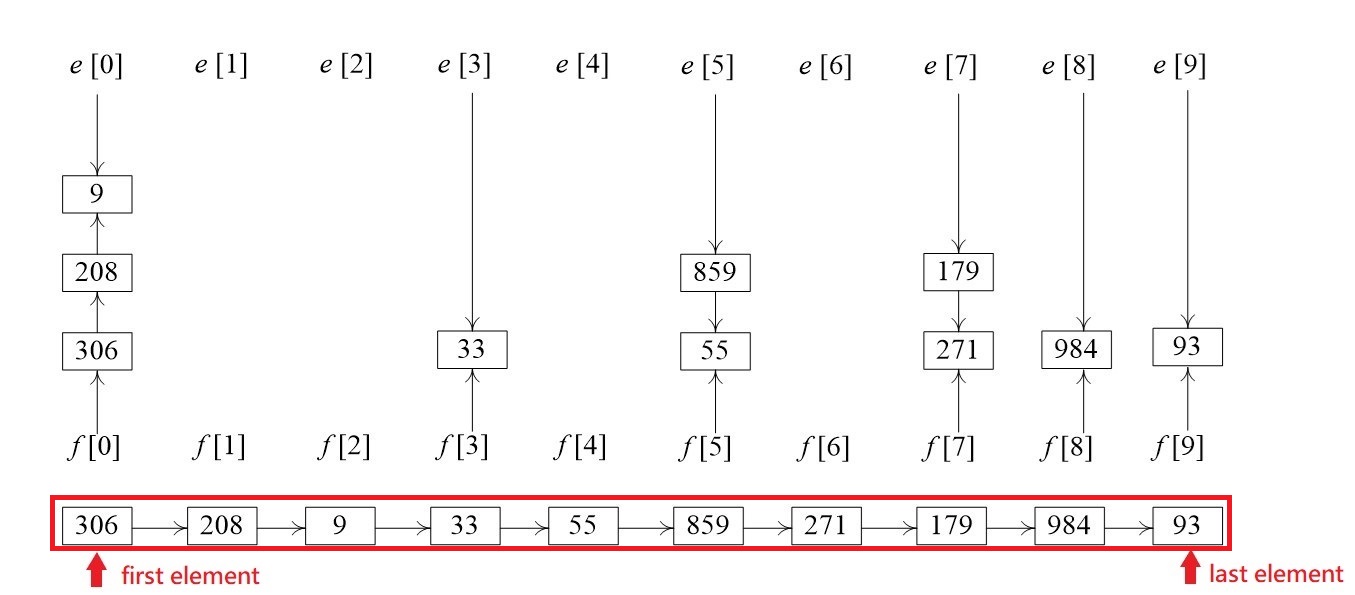
the output will be: 271 9

306 93

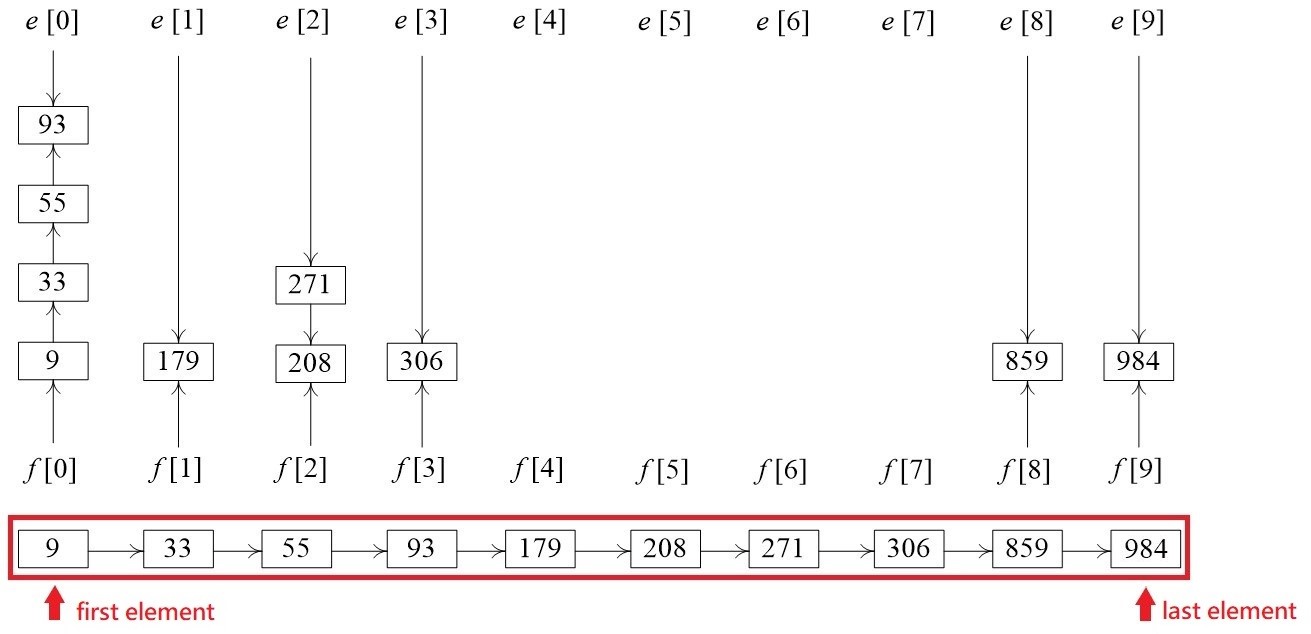
9 984

Pass 1 

Pass 2



Pass 3



Another Example: **A = [57, 67, 59, 7, 82]**

**r = 4**

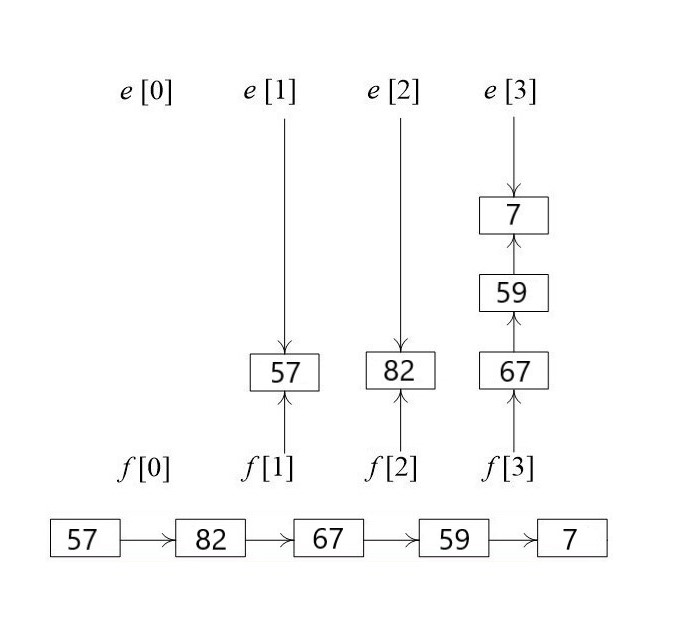
the output will be: 57 7

82 59

67 59

7 82

Pass 1



**57 = 0321**

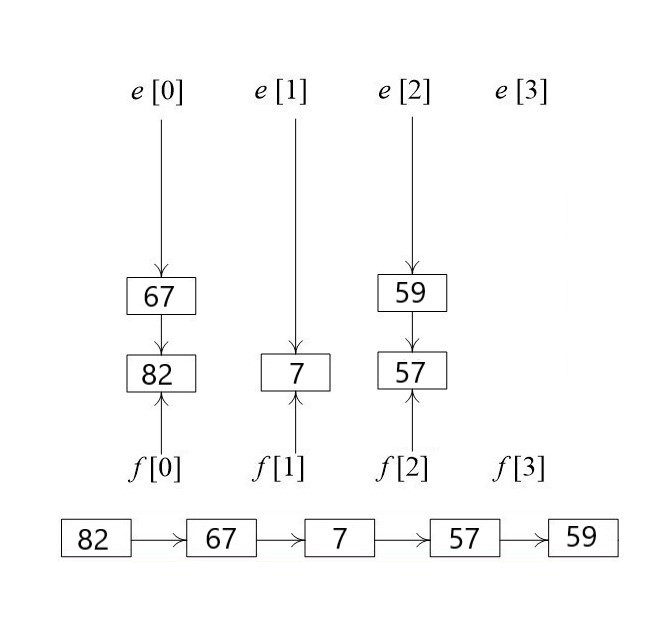
**67 = 1003**

**59 = 0323**

**7 = 0013**

**82 = 1102**

Pass 2



**57 = 0321**

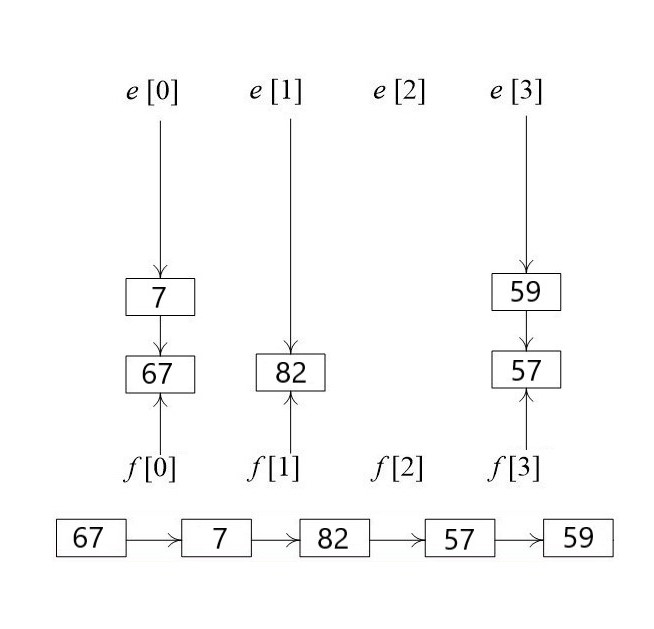
**82 = 1102**

**67 = 1003**

**59 = 0323**

**7 = 0013**

Pass 3



**82 = 1102**

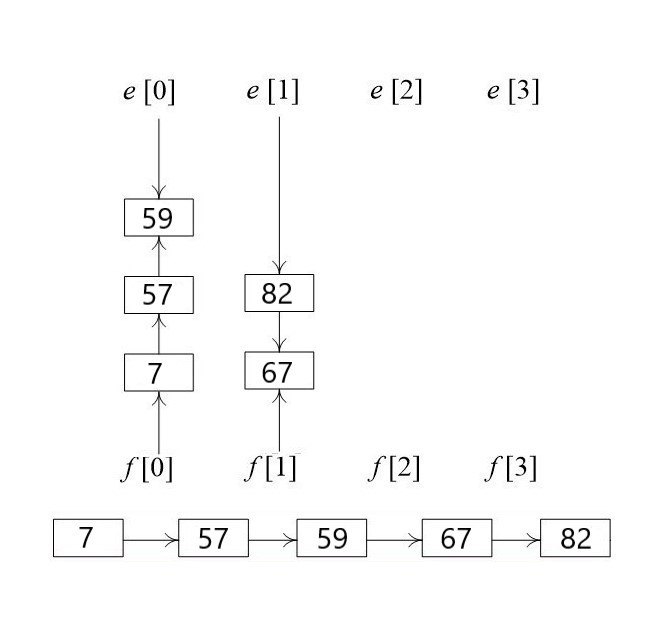
**67 = 1003**

**7 = 0013**

**57 = 0321**

**59 = 0323**

Pass 4



**67 = 1003**

**7 = 0013**

**82 = 1102**

**57 = 0321**

**59 = 0323**

You are allowed to use STL.

But don’t use any STL related to sort.

Input

There are multiple testcases, and each testcase begins with a line containing two integers **n** and **r**, the number of elements in sequence and the radix.

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

Please note:

1)

2)

3)

(each value is decimal integer)

Output

For each input, output the following results separated by a newline symbol.

* Inversion pairs count
* First element and last element of results in every pass of the radix sort(LSD)

(output of every pass separated by a newline symbol too)

Sample input

5 4

57 67 59 7 82

Sample output

4

57 7

82 59

67 59

7 82