

Merge K sorted arrays

Problem

Submissions

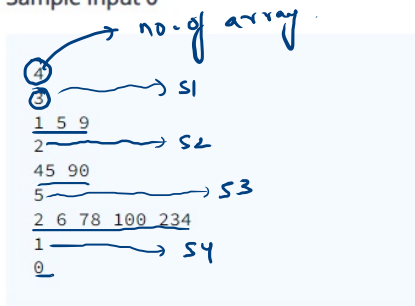
Leaderboard

Discussions

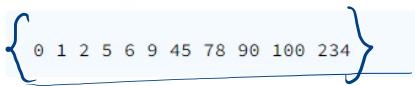
Given k different arrays, which are sorted individually (in ascending order). You need to merge all the given arrays such that output array should be sorted (in ascending order).

Hint : Use Heaps.

Sample Input 0

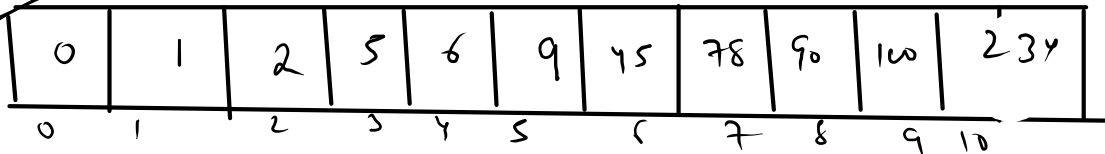
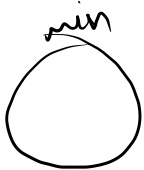


Sample Output 0



1. Que? ✓

2. Approach



Sample Input 0

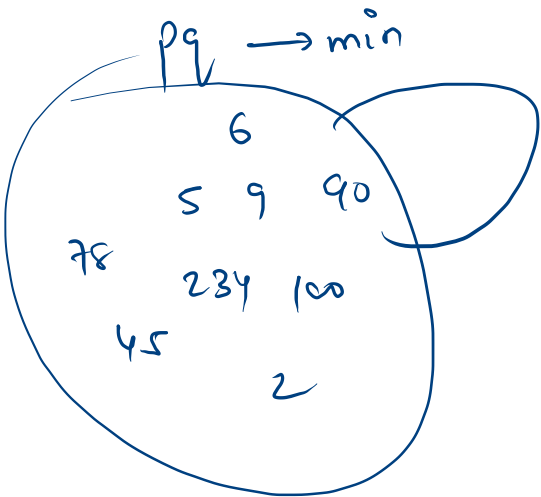
manage.

```
4
3
1 5 9
2
45 90
5
2 6 78 100 234
1
0
```

Sample Output 0

0 1 2 5 6 9 45 78 90 100 234

add.
pg



0 1

```
1 import java.io.*;
2 import java.util.*;
3
4 public class Solution {
5
6     public static void main(String[] args) {
7         Scanner scn = new Scanner(System.in);
8         int t = scn.nextInt(); //number of array
9         PriorityQueue<Integer> pq = new PriorityQueue<>();
10        while(t-- > 0){
11            int n = scn.nextInt(); //3
12            while(n-- > 0){
13                pq.add(scn.nextInt());
14            }
15        }
16        while(pq.size() != 0){
17            System.out.print(pq.remove() + " ");
18        }
19
20    }
21 }
```

```

1 import java.io.*;
2 import java.util.*;
3
4 public class Solution {
5
6     public static void main(String[] args) {
7         Scanner scn = new Scanner(System.in);
8         int t = scn.nextInt(); //number of array
9         PriorityQueue<Integer> pq = new PriorityQueue<>();
10        for(int i = 0; i < t; i++){
11            int n = scn.nextInt();
12            for(int j = 0; j < n; j++){
13                pq.add(scn.nextInt());
14            }
15        }
16        // while(t-- > 0){
17        //     int n = scn.nextInt(); //3
18        //     while(n-- > 0){
19        //         pq.add(scn.nextInt());
20        //     }
21        // }
22        while(pq.size() != 0){
23            System.out.print(pq.remove() + " ");
24        }
25    }
26 }
27

```

t=4

min

	5	9	45
90	2	6	78
100	234		

size=11 !=0

10 !=0

0 1 - - - - -

④

③

1 5 9

2

45 90

5

2 6 78 100 234

1

0

weakest rows

Problem	Submissions	Leaderboard	Discussions
---------	-------------	-------------	-------------

You are given an $m \times n$ binary matrix **mat** of 1's (representing soldiers) and 0's (representing civilians). The soldiers are positioned in **front** of the civilians. That is, all the 1's will appear to the **left** of all the 0's in each row.

A row i is **weaker** than a row j if one of the following is true:

- The number of soldiers in row i is less than the number of soldiers in row j .
- Both rows have the same number of soldiers and $i < j$.

Return the indices of the k **weakest** rows in the matrix ordered from weakest to strongest.

1 \rightarrow soldier
0 \rightarrow civilian

Sample Input 0

m	n					
5	5	3				
1	1	0	0	0		
1	1	1	1	0		
1	0	0	0	0		
1	1	0	0	0		
1	1	1	1	1		

$m \times n$

Sample Output 0

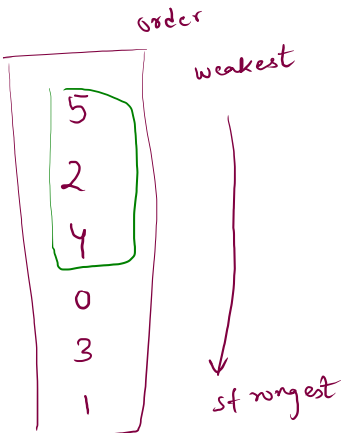
2	0	3
---	---	---

Case 1:
 $i \dots 1 \ 1 \ 0 \ 0 \ 0 \rightarrow 2$
 $j \dots 1 \ 1 \ 1 \ 1 \ 0 \rightarrow 4$
 $2 < 4$

Case 2:
 $i \dots 1 \ 1 \ 0 \ 0 \ 0 \rightarrow 2$
 $j \dots 1 \ 1 \ 0 \ 0 \ 0 \rightarrow 2$
 $i < j$
 $k=3$ $k=5$

Case 3:

0	1	1	1	1	0	0	0	...	4
1	1	1	1	1	1	0	0	...	5
2	1	1	0	0	0	0	0	...	2
3	1	1	1	1	0	0	0	...	4
4	1	1	0	0	0	0	0	...	2
5	1	0	0	0	0	0	0	...	1



5 2 4
5 2 4 0 3

k=3.

2 0 3.

	m	n	k	
	5	5	3	
0	1	1	0	0
1	1	1	1	1
2	1	0	0	0
3	1	1	0	0
4	1	1	1	1

↓
2
4
1
2
5

24
125

1. no. of soldiers.

2
4
1
2
5

2. row number.

encoding.
n + 2 → x

	5	5	3	
0	1	1	0	0
1	1	1	1	1
2	1	0	0	0
3	1	1	0	0
4	1	1	1	1

$$x = \underline{\underline{8 \times 10^5 + i}}$$

s	x
2	$2 \times 10^5 + 0$
3	$4 \times 10^5 + 1$
1	$1 \times 10^5 + 2$
2	$2 \times 10^5 + 3$
5	$5 \times 10^5 + 4$

min

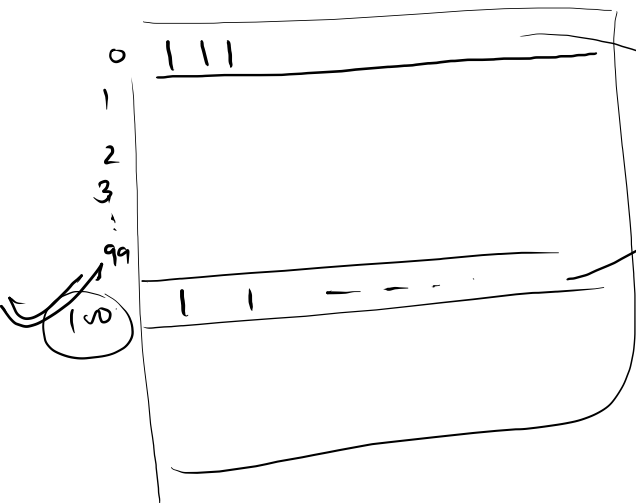
200000
400001
100002
200003
500004

1st → 100002 % 10⁵ = ? (2) ✓
 2nd → 200000 % 10⁵ = (0) ✓
 3rd → 200003 % 10⁵ = (3) ✓

$$x = s * y + i$$

$$x \% y =$$

$$x = 5 \times (100) + i$$



$$2 \times 100 + 100 = 300$$

$$3 \times 100 + 0 = 300$$


```

1 import java.io.*;
2 import java.util.*;
3 public class Solution {
4     public static void main(String[] args) {
5         Scanner scn = new Scanner(System.in);
6         int m = scn.nextInt();
7         int n = scn.nextInt();
8         int k = scn.nextInt();
9         int [][] A = new int[m][n];
10        for(int i = 0; i < m; i++){
11            for(int j = 0; j < n; j++){
12                A[i][j] = scn.nextInt();
13            }
14        }
15        //logic
16        PriorityQueue<Integer> pq = new PriorityQueue();
17        for(int i = 0; i < m; i++){
18            int sum = 0;
19            for(int j = 0; j < n; j++){
20                sum += A[i][j];
21            }
22            int x = sum * 100000 + i;
23            pq.add(x);
24        }
25        for(int i = 0; i < k; i++){
26            System.out.print(pq.remove() % 100000 + " ");
27        }
28    }
29 }

```

m n k
 2 4 2

	0	1	2	3
0	1	1	1	0
1	1	0	0	0

30000
~~10000~~

$100001 \% 10^5$

$= 1$
 $30000 \times 10^5 = 0$

Reduce Array Size to the half 1

Problem	Submissions	Leaderboard	Discussions
---------	-------------	-------------	-------------

You are given an integer array `arr`. You can choose a set of integers and remove all the occurrences of these integers in the array.

Return the minimum size of the set so that at least half of the integers of the array are removed.

Sample Input 0

10

3

3

3

3

3

5

5

5

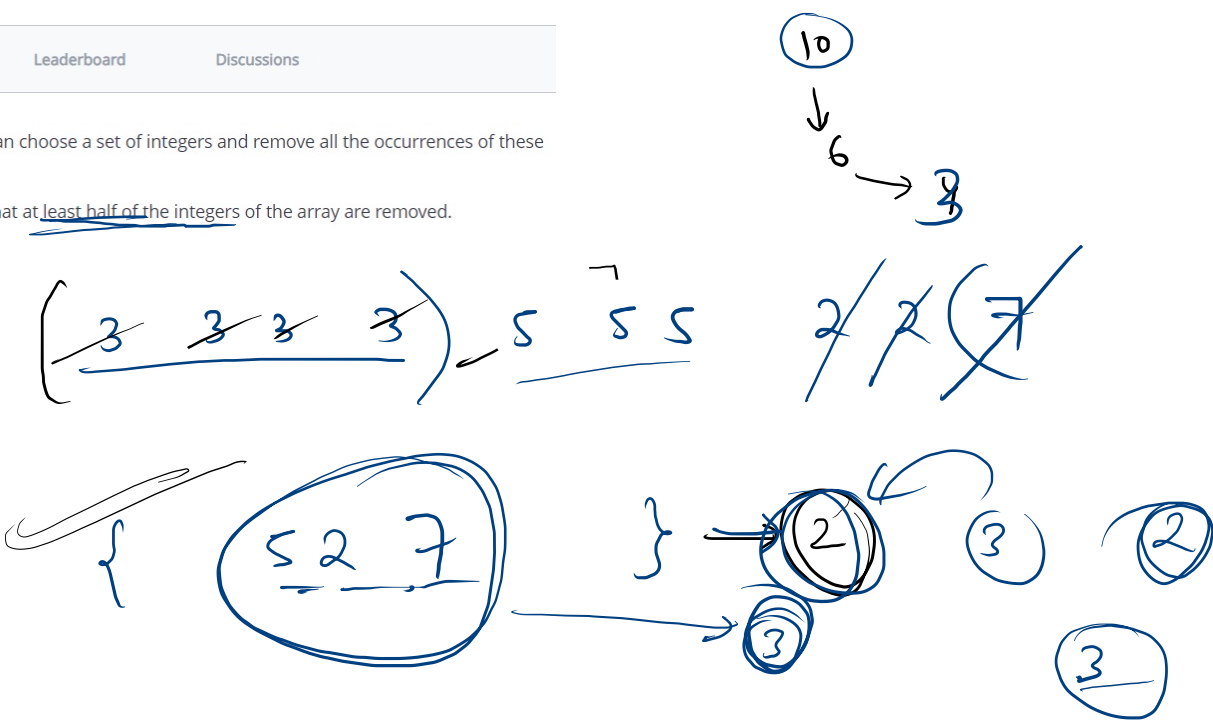
2

2

7

Sample Output 0

2



* 10
 atleast
 x (5)

3 3 3 3 5 5 5 7 7 2

pg \rightarrow hm.

~~3~~ \rightarrow 4
~~5~~ \rightarrow 3
~~7~~ \rightarrow 2
 2 \rightarrow 1

ans = 2.

10 - 4 - 3

= 3 $\leq n/2$