You are given an m x 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.

Example 1:

Input: mat =

[[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]],

k = 3

Output: [2,0,3]

Explanation:

The number of soldiers in each row is:

- Row 0: 2

- Row 1: 4

- Row 2: 1

- Row 3: 2

- Row 4: 5

The rows ordered from weakest to strongest are [2,0,3,1,4].

Example 2:

Input: mat =

[[1, 0, 0, 0],

[1, 1, 1,1],

[1, 0, 0, 0],

[1, 0, 0,0]],

k = 2

Output: [0,2]

Explanation:

The number of soldiers in each row is:

- Row 0: 1

- Row 1: 4

- Row 2: 1

- Row 3: 1

The rows ordered from weakest to strongest are [0, 2, 3, 1].

CODE:

import java.util.\*;

public class WeakestRows {

public static int[] kWeakestRows(int[][] mat, int k) {

int m = mat.length;

int n = mat[0].length;

PriorityQueue<int[]> pq = new PriorityQueue<>((a, b) -> {

if (a[1] == b[1])

return a[0] - b[0];

return a[1] - b[1];

});

for (int i = 0; i < m; i++) {

int count = 0;

for (int j = 0; j < n; j++) {

if (mat[i][j] == 1)

count++;

else

break;

}

pq.offer(new int[]{i, count});

}

int[] result = new int[k];

for (int i = 0; i < k; i++) {

result[i] = pq.poll()[0];

}

return result;

}

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter the number of rows: ");

int m = scanner.nextInt();

System.out.print("Enter the number of columns: ");

int n = scanner.nextInt();

int[][] mat = new int[m][n];

System.out.println("Enter the elements of the matrix:");

for (int i = 0; i < m; i++) {

for (int j = 0; j < n; j++) {

mat[i][j] = scanner.nextInt();

}

}

System.out.print("Enter the value of k: ");

int k = scanner.nextInt();

int[] result = kWeakestRows(mat, k);

System.out.println("Output: " + Arrays.toString(result));

}

}

OUTPUT:

C:\javap>javac WeakestRows.java

C:\javap>java WeakestRows

Enter the number of rows: 5

Enter the number of columns: 5

Enter the elements of the matrix:

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

Enter the value of k: 3

Output: [2, 0, 3]

