import java.util.PriorityQueue;

class Solution{

class pair{

int val,row,col;

public pair(int v, int r, int c){

this.val = v;

this.row = r;

this.col = c;

}

}

public void addHeap(PriorityQueue<pair> heap, int[][] matrix, int row, int col){

int length = matrix[0].length;

if(col >= length)

return;

heap.add(new pair(matrix[row][col],row,col));

}

public int solve(int[][] matrix, int k){

int n = matrix.length;

int length = matrix[0].length;

PriorityQueue<pair> heap = new PriorityQueue<>((a,b) -> a.val - b.val);

for(int i = 0; i < n; i++)

addHeap(heap,matrix,i,0);

pair temp;

int value = -1;

while(k > 0){

temp = heap.poll();

value = temp.val;

k -= 1;

addHeap(heap,matrix,temp.row,temp.col+1);

}

return value;

}

}