Monk and New Array

Attempted by: **495** / Accuracy: **50%** / Maximum Score: **20** / 4 Votes

Tag(s): Binary Search, Easy, Sorting, approved

Monk has a matrix of size $N \times M$, and he wants to pick one element from each row to make a new array A of size N. He wants to find the minimum possible value of absolute difference between any two adjacent elements in the array A. Please note that the element picked from row 1, will become $A[1]$, element picked from row 2 will become $A[2]$, and so on.

Input:
First line consists of two space separated integers denoting N and M.
Each of the following N lines consists of M space separated integers denoting the matrix mat.

Output:
Print the required answer in a new line.

Constraints:
$2 \leq N, M \leq 1000$
$1 \leq mat[i][j] \leq 10^9$

SAMPLE INPUT

```
2 2
8 4
6 8
```

SAMPLE OUTPUT

```
0
```

Explanation

Minimum difference Monk can achieve is 0 by choosing 8 from both rows.

Time Limit:	1.0 sec(s) for each input file.
Memory Limit:	256 MB
Source Limit:	1024 KB
Marking Scheme:	Marks are awarded when all the testcases pass.
Allowed Languages:	C, C++, Clojure, C#, D, Erlang, F#, Go, Groovy, Haskell, Java, Java 8, JavaScript(Rhino), JavaScript(Node.js), Lisp, Lisp (SBCL), Lua, Objective-C, OCaml, Octave, Pascal, Perl, PHP, Python, Python 3, R(RScript), Racket, Ruby, Rust, Scala, Scala 2.11.8, Swift, Visual Basic