We are given an array A of N lowercase letter strings, all of the same length.

Now, we may choose any set of deletion indices, and for each string, we delete all the characters in those indices.

For example, if we have an array A = ["abcdef","uvwxyz"] and deletion indices {0, 2, 3}, then the final array after deletions is ["bef", "vyz"], and the remaining columns of A are ["b","v"], ["e","y"], and ["f","z"].  (Formally, the c-th column is [A[0][c], A[1][c], ..., A[A.length-1][c]]).

Suppose we chose a set of deletion indices D such that after deletions, each remaining column in A is in **non-decreasing** sorted order.

Return the minimum possible value of D.length.

**Example 1:**

**Input:** A = ["cba","daf","ghi"]

**Output:** 1

**Explanation:**

After choosing D = {1}, each column ["c","d","g"] and ["a","f","i"] are in non-decreasing sorted order.

If we chose D = {}, then a column ["b","a","h"] would not be in non-decreasing sorted order.

**Example 2:**

**Input:** A = ["a","b"]

**Output:** 0

**Explanation:** D = {}

**Example 3:**

**Input:** A = ["zyx","wvu","tsr"]

**Output:** 3

**Explanation:** D = {0, 1, 2}

**Constraints:**

* 1 <= A.length <= 100
* 1 <= A[i].length <= 1000