Given a **square** array of integers A, we want the **minimum** sum of a *falling path* through A.

A falling path starts at any element in the first row, and chooses one element from each row. The next row's choice must be in a column that is different from the previous row's column by at most one.

Example 1:

Input: [[1,2,3],[4,5,6],[7,8,9]]**Output:** 12**Explanation:** The possible falling paths are:

- [1,4,7], [1,4,8], [1,5,7], [1,5,8], [1,5,9]
- [2,4,7], [2,4,8], [2,5,7], [2,5,8], [2,5,9], [2,6,8], [2,6,9]
- [3,5,7], [3,5,8], [3,5,9], [3,6,8], [3,6,9]

The falling path with the smallest sum is [1,4,7], so the answer is 12.

Constraints:

- 1 <= A.length == A[0].length <= 100
- -100 <= A[i][j] <= 100