Toeplitz Matrix

A matrix is *Toeplitz* if every diagonal from top-left to bottom-right has the same element.

Now given an `M x N` matrix, return `True` if and only if the matrix is *Toeplitz*.

Example 1:

```
Input: matrix = [[1,2,3,4],[5,1,2,3],[9,5,1,2]]
Output: True
Explanation:
1234
5123
9512

In the above grid, the diagonals are "[9]", "[5, 5]", "[1, 1, 1]", "[2, 2, 2]", "[3, 3]", "[4]", and in each diagonal all elements are the same, so the answer is True.
```

Example 2:

```
Input: matrix = [[1,2],[2,2]]
Output: False
Explanation:
The diagonal "[1, 2]" has different elements.
```

Note:

1. `matrix` will be a 2D array of integers.
2. `matrix` will have a number of rows and columns in range `[1, 20]`.
3. `matrix[i][j]` will be integers in range `[0, 99]`.

Solution 1

```
class Solution {
    public boolean isToeplitzMatrix(int[][] matrix) {
        for (int i = 0; i < matrix.length - 1; i++) {
            for (int j = 0; j < matrix[i].length - 1; j++) {
                if (matrix[i][j] != matrix[i + 1][j + 1]) return false;
            }
        }
        return true;
    }
}
```

written by shawngao original link here

Solution 2

```
def isToeplitzMatrix(self, m):
        for i in range(len(m) - 1):
            for j in range(len(m[0]) - 1):
                if m[i][j] != m[i + 1][j + 1]:
                    return False
        return True
```

Make it 1 line:

```
    def isToeplitzMatrix(self, m):
        return all(m[i][j] == m[i+1][j+1] for i in range(len(m)-1) for j in range(le
n(m[0])-1))
```

written by lee215 original link here

Solution 3

C++

```
class Solution {
public:
    bool isToeplitzMatrix(vector<vector<int>>& matrix) {
        int m = matrix.size(), n = matrix[0].size();
        for (int i = 1; i < m; i++)
            for (int j = 1; j < n; j++)
                if (matrix[i][j] != matrix[i - 1][j - 1])
                    return false;
        return true;
    }
};
```

Java

```
class Solution {
    public boolean isToeplitzMatrix(int[][] matrix) {
        int m = matrix.length, n = matrix[0].length;
        for (int i = 1; i < m; i++)
            for (int j = 1; j < n; j++)
                if (matrix[i][j] != matrix[i - 1][j - 1])
                    return false;
        return true;
    }
}
```

written by alexander original link here