Problem 1030: Matrix Cells in Distance Order

Problem Information

Difficulty: Easy
Acceptance Rate: 73.57%
Paid Only: No
Tags: Array, Math, Geometry, Sorting, Matrix

Problem Description

You are given four integers `row`, `cols`, `rCenter`, and `cCenter`. There is a `rows x cols` matrix and you are on the cell with the coordinates `(rCenter, cCenter)`.

Return _the coordinates of all cells in the matrix, sorted by their**distance** from _`(rCenter, cCenter)`_from the smallest distance to the largest distance_. You may return the answer in **any order** that satisfies this condition.

The **distance** between two cells `(r1, c1)` and `(r2, c2)` is `|r1 - r2| + |c1 - c2|`.

Example 1:

Input: rows = 1, cols = 2, rCenter = 0, cCenter = 0 **Output:** [[0,0],[0,1]] **Explanation:** The distances from (0, 0) to other cells are: [0,1]

Example 2:

Input: rows = 2, cols = 2, rCenter = 0, cCenter = 1 **Output:** [[0,1],[0,0],[1,1],[1,0]] **Explanation:** The distances from (0, 1) to other cells are: [0,1,1,2] The answer [[0,1],[1,1],[0,0],[1,0]] would also be accepted as correct.

Example 3:

Input: rows = 2, cols = 3, rCenter = 1, cCenter = 2 **Output:** [[1,2],[0,2],[1,1],[0,1],[1,0],[0,0]] **Explanation:** The distances from (1, 2) to other cells are: [0,1,1,2,2,3] There are other answers that would also be accepted as correct, such as [[1,2],[1,1],[0,2],[1,0],[0,1],[0,0]].

Constraints:

* `1 <= rows, cols <= 100` * `0 <= rCenter < rows` * `0 <= cCenter < cols`

Code Snippets

C++:

```
class Solution {
public:
vector<vector<int>> allCellsDistOrder(int rows, int cols, int rCenter, int
cCenter) {

}
};
```

Java:

```
class Solution {
public int[][] allCellsDistOrder(int rows, int cols, int rCenter, int
cCenter) {

}
}
```

Python3:

```
class Solution:
def allCellsDistOrder(self, rows: int, cols: int, rCenter: int, cCenter: int)
-> List[List[int]]:
```