Problem 885: Spiral Matrix III

Problem Information

Difficulty: Medium
Acceptance Rate: 84.54%
Paid Only: No
Tags: Array, Matrix, Simulation

Problem Description

You start at the cell `(rStart, cStart)` of an `rows x cols` grid facing east. The northwest corner is at the first row and column in the grid, and the southeast corner is at the last row and column.

You will walk in a clockwise spiral shape to visit every position in this grid. Whenever you move outside the grid's boundary, we continue our walk outside the grid (but may return to the grid boundary later.). Eventually, we reach all `rows * cols` spaces of the grid.

Return _an array of coordinates representing the positions of the grid in the order you visited them_.

Example 1:

![](https://s3-lc-upload.s3.amazonaws.com/uploads/2018/08/24/example_1.png)

Input: rows = 1, cols = 4, rStart = 0, cStart = 0 **Output:** [[0,0],[0,1],[0,2],[0,3]]

Example 2:

![](https://s3-lc-upload.s3.amazonaws.com/uploads/2018/08/24/example_2.png)

Input: rows = 5, cols = 6, rStart = 1, cStart = 4 **Output:** [[1,4],[1,5],[2,5],[2,4],[2,3],[1,3],[0,3],[0,4],[0,5],[3,5],[3,4],[3,3],[3,2],[2,2],[1,2],[0,2],[4,5],[4,4],[4,3],[4,2],[4,1],[3,1],[2,1],[1,1],[0,1],[4,0],[3,0],[2,0],[1,0],[0,0]]

Constraints:

* `1 <= rows, cols <= 100` * `0 <= rStart < rows` * `0 <= cStart < cols`

Code Snippets

C++:

```
class Solution {
public:
vector<vector<int>> spiralMatrixIII(int rows, int cols, int rStart, int
cStart) {

}
};
```

Java:

```
class Solution {
public int[][] spiralMatrixIII(int rows, int cols, int rStart, int cStart) {

}
}
```

Python3:

```
class Solution:
def spiralMatrixIII(self, rows: int, cols: int, rStart: int, cStart: int) ->
List[List[int]]:
```