**Growth in 2 Dimensions**

Start with an infinite two-dimensional grid filled with zeros, indexed from (1,1) at the bottom left corner with coordinates increasing toward the top and right. Given a series of coordinates (r,c) where r is the ending row and c is the ending column, add 1 to each element in the range from (1,1) to (r,c) inclusive. Once all coordinates are processed, determine how many cells contain the maximal value in the grid.

Example: upRight = ["1 4", "2 3", "4 1"]

The two space-separated integers within each string represent r and c respectively. The diagrams the attached images show each iteration starting at zero. The maximal value in the grid is 3, and there is 1 occurence at cell (1,1).

**Initial Grid**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **4** | **0** | **0** | **0** | **0** |
| **3** | **0** | **0** | **0** | **0** |
| **2** | **0** | **0** | **0** | **0** |
| **1** | **0** | **0** | **0** | **0** |
|  | **1** | **2** | **3** | **4** |

**Step 0: r = 1, c = 4**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **4** | **0** | **0** | **0** | **0** |
| **3** | **0** | **0** | **0** | **0** |
| **2** | **0** | **0** | **0** | **0** |
| **1** | **1** | **1** | **1** | **1** |
|  | **1** | **2** | **3** | **4** |

**Step 1: r = 2, c = 3**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **4** | **0** | **0** | **0** | **0** |
| **3** | **0** | **0** | **0** | **0** |
| **2** | **1** | **1** | **1** | **0** |
| **1** | **2** | **2** | **2** | **1** |
|  | **1** | **2** | **3** | **4** |

**Step 1: r = 4, c = 1**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **4** | **1** | **0** | **0** | **0** |
| **3** | **1** | **0** | **0** | **0** |
| **2** | **2** | **1** | **1** | **0** |
| **1** | **3** | **2** | **2** | **1** |
|  | **1** | **2** | **3** | **4** |

Complete a function countMax to return the number of occurrences of the final grid's maximal element. countMax has the following parameter(s): string upRight[n]: an array of strings made of two space-seperated integers, r and c.

**Return**

Long: the number of occurrences of the final grid’s maximal element.