REPORT

AUTHORS:

Kelvin Anim Offei

Gabriel Mawunyo Owusu

Description

To aid in coming up with an algorithm to write our code, we created a populateArray() function which had a two dimensional array and variable of type int as parameters. It also had two integers, i and j as attributes. A for loop was created to randomly generate the values of i and j. We represented our undefined values with the integer values 0 since the array was initialized as an integer array. For a random index called, given that zero represents undefined values, indexes with value zero will be assigned 1, and indexes with values greater than one would have their count increased by a value of 1. E.g if the values of i and j at any given time are 0, their values will be changed to 1. Else, the value is increased by 1.

To find the values that meet the criteria, thus positive values, a searchvalidentries function was created. It consisted of 4 integer parameters which were the indexes of the boundary box, an array parameter to refer to the object two-dimensional array. A nested for loop was created to loop through the two-dimensional array using the indexes as limits, and only prints out the values of that meet the criteria of being positive (greater than 0).

Pseudocode

Create an array with default values 0.

populateArray(int array[101][101], int N)

* For a number of loops N;
  + Generate a random row value i less than 101
  + Generate a random colum value j less than 101
  + If the value in the location [i][j] is 0, change count value to 1
  + Else, increases count value by 1

}

searchValidEntries(int XL, int YL, int XH, int YH, int array[101][101]){

* For a range of rows XL to XH
  + For a range of columns YL to YH
    - If the value at index [i][j] is not zero
      * Print out index [i][j] and count value

}