

Prompt	Scratchpad	Our Solution(s)	Video Explanation	Run Code
--------	------------	-----------------	-------------------	----------

Solution 1	Solution 2	Solution 3	Solution 4
<pre>1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved. 2 3 ▼ import java.util.*; 4 5 ▼ class Program { 6 // O(n^4) time O(1) space - where n is the height and width of the matrix 7 ▼ public static boolean squareOfZeroes(List<List<Integer>> matrix) { 8 int n = matrix.size(); 9 ▼ for (int topRow = 0; topRow < n; topRow++) { 10 ▼ for (int leftCol = 0; leftCol < n; leftCol++) { 11 int squareLength = 2; 12 ▼ while (squareLength <= n - leftCol && squareLength <= n - topRow) { 13 int bottomRow = topRow + squareLength - 1; 14 int rightCol = leftCol + squareLength - 1; 15 if (isSquareOfZeroes(matrix, topRow, leftCol, bottomRow, rightCol)) return true; 16 squareLength++; 17 } 18 } 19 } 20 return false; 21 } 22 23 // r1 is the top row, c1 is the left column 24 // r2 is the bottom row, c2 is the right column 25 public static boolean isSquareOfZeroes(List<List<Integer>> matrix, 26 int r1, 27 int c1, 28 int r2, 29 int c2 30 ▼) { 31 ▼ for (int row = r1; row < r2 + 1; row++) { 32 if (matrix.get(row).get(c1) != 0 matrix.get(row).get(c2) != 0) return false; 33 } 34 ▼ for (int col = c1; col < c2 + 1; col++) { 35 if (matrix.get(r1).get(col) != 0 matrix.get(r2).get(col) != 0) return false; 36 } 37 return true; 38 } 39 } 40</pre>			

