

Part A

Public static position findPosition(int arr[], int l, int r, int x)

Length = int arr.length;

for (int i = 0; i < Length; i++) {

for (int f = 0; f < int arr[0]; length++) {

return Position(i, x);

else if (

return null;

} // End else statement

} // End main method

Part B

Public static positionC (2D 2 Success, All of Col 0 is 0's)

PositionC (PositionArr = New Position[int arr.length][arr.length])

for (int r = 0; r < int arr.length; r++) {

for (int r = 0; r < int arr[0].length; r++) {

PositionArr [(2C r)] = findPositionC (arr [r] [0], r, int arr);

} // End for loop

return PositionArr;

} // End main

Part A

public digits Cint num25

digitList = new ArrayList<Integer>();

while (num > 0):

digitList.add(new Integer(0));

7

if (num > 0):

digitList.add(new Integer());

7 || end if

7 || end digit method

Part B

public boolean isStrictlyIncreasing()

for (int i=0; i < digitList.size(); i+1):

if (digitList.get(i) >= digitList.get(i+1))

return false;

7 || end if statement

7 here for loop

return true;

7 || end main method