

Name: Jingshi Liu

Section: Image Processing

Project: **Project 4 - Eight and four Connected Component Algo**

Due Date: **Oct 12th**

Algorithm Steps

step 0: inFile \leftarrow open the input file from args [0]

Connectness \leftarrow args [1]

RFprettyPrintFile, labelFile, propertyFile, deBugFile \leftarrow open from args []

numRows, numCols, minVal, maxVal \leftarrow read from inFile

zeroFramedAry \leftarrow dynamically allocate.

newLabel \leftarrow 0

step 1: zero2D (zeroFramedAry)

step 2: loadImage (inFile, zeroFramedAry)

step 3: if connectness == 4

connected4 (zeroFramedAry, newLabel, EQAry, RFprettyPrintFile, deBugFile)

step 4: if connectness == 8

connected4 (zeroFramedAry, newLabel, EQAry, RFprettyPrintFile, deBugFile)

step 5: labelFile \leftarrow output numRows, numCols, newMin, newMax to labelFile

step 6: printImg (zeroFramedAry, labelFile) // Output the result of pass3 inside of zeroFramedAry

step 7: printCCproperty (propertyFile) // print cc properties to propertyFile

step 8: drawBoxes (zeroFramedAry, CCproperty, trueNumCC, debugFile) // draw on zeroFramed image.

step 9: imgReformat (zeroFramedAry, RFprettyPrintFile)

step 10: RFprettyPrintFile ← print trueNumCC to RFprettyPrintFile with proper caption

step 12: close all files

Source Code:

```
import java.io.File;
import java.io.FileReader;
import java.io.FileWriter;
import java.io.IOException;
import java.util.Arrays;
import java.util.Scanner;

class Property{
    int label,
        numPixels,
        minR,
        minC,
        maxR,
        maxC;

    Property(int label, int numPixels, int minR, int minC, int maxR, int maxC){
        this.label = label;
        this.numPixels = numPixels;
        this.minR = minR;
        this.minC = minC;
        this.maxR = maxR;
        this.maxC = maxC;
    }
}

class ConnectedComponentLabel{
    int numRows,
        numCols,
        minVal,
        maxVal,
        newLabel,
        trueNumConnectedComponent,
        newMin,
```

```

        newMax;
int[][] zeroFramedArray;
int[] nonZeroNeighborArray;
int[] equalArray;
char option;
Property[] connectedComponentProperty;

ConnectedComponentLabel(Scanner inFile){
    numRows = inFile.nextInt();
    numCols = inFile.nextInt();
    minVal = inFile.nextInt();
    maxVal = inFile.nextInt();

    equalArray = new int[(numRows * numCols)/4];
    for (int i = 0; i < equalArray.length; i++) {
        equalArray[i] = i;
    }
    zeroFramedArray = new int[numRows + 2][numCols + 2];
    newLabel = 0;

    loadImage(inFile);
}

void zero2D(int[][] array){
    for (int[] ints : array) {
        Arrays.fill(ints, 0);
    }
}

void negative1D(int[] array){
    Arrays.fill(array, -1);
}

void loadImage(Scanner inFile){

```

```

        for (int i = 1; i < numRows + 1; i++) {
            for (int j = 1; j < numCols + 1; j++) {
                zeroFramedArray[i][j] = inFile.nextInt();
            }
        }
    }
}

```

```

int findMax(int[][] array){
    int max = array[0][0];
    for (int[] ints : array) {
        for (int anInt : ints) {
            if (anInt > max)
                max = anInt;
        }
    }
    return max;
}

```

```

void imageReformat(FileWriter outFile) throws IOException {
    int curWidth, pixelWidth = Integer.toString(findMax(zeroFramedArray)).length();

    for(int r = 1; r < numRows + 1; r++){
        for(int c = 1; c < numCols + 1; c++){
            if (zeroFramedArray[r][c] == 0)
                outFile.write(".");
            else
                outFile.write(Integer.toString(zeroFramedArray[r][c]));

            curWidth = Integer.toString(zeroFramedArray[r][c]).length();
            outFile.write(" ");
            while(curWidth < pixelWidth){
                outFile.write(" ");
                curWidth++;
            }
        }
    }
}

```

```

        }
        outFile.write('\n');
    }
}

void connected4(FileWriter reformatPrettyPrintFile, FileWriter debugFile) throws IOException
{
    debugFile.write("Entering connect4 method\n");

    // 1
    connect4Pass1();
    debugFile.write("After connected4 pass1, newLabel = " + newLabel + "\n");
    reformatPrettyPrintFile.write("After connected4 pass1, zeroFramedArray\n");
    imageReformat(reformatPrettyPrintFile);
    reformatPrettyPrintFile.write("\n\nAfter connected4 pass1, equalArray\n");
    printEqualArray(reformatPrettyPrintFile);

    // 2
    connect4Pass2();
    debugFile.write("After connected4 pass2, newLabel = " + newLabel);
    reformatPrettyPrintFile.write("\n\nAfter connected4 pass2\n");
    reformatPrettyPrintFile.write("\n\nAfter connected4 pass2, zeroFramedArray\n");
    imageReformat(reformatPrettyPrintFile);
    reformatPrettyPrintFile.write("\n\nAfter connected4 pass2, equalArray\n");
    printEqualArray(reformatPrettyPrintFile);

    // 3
    trueNumConnectedComponent = manageEqualArray();
    reformatPrettyPrintFile.write("\n\nAfter connected4 manageEqualArray(), equalArray\n");
    printEqualArray(reformatPrettyPrintFile);
    newMin = 0;
    newMax = trueNumConnectedComponent;
    connectedComponentProperty = new Property[newMax + 1];
    debugFile.write("In connected4, after manageEqualArray, trueNumConnectedComponent = "
        + trueNumConnectedComponent + "\n");
}

```

```

// 4
connectPass3(debugFile);

reformatPrettyPrintFile.write("\n\nAfter connected4 pass3, zeroFramedArray\n");
imageReformat(reformatPrettyPrintFile);
reformatPrettyPrintFile.write("\n\nAfter connected4 pass3, equalArray\n");
printEqualArray(reformatPrettyPrintFile);

debugFile.write("Leaving connected4 method\n");
}

void connected8(FileWriter reformatPrettyPrintFile, FileWriter debugFile) throws IOException
{
    debugFile.write("Entering connect8 method\n");

    // 1
    connect8Pass1();
    debugFile.write("After connected8 pass1, newLabel = " + newLabel + "\n");
    reformatPrettyPrintFile.write("After connected8 pass1, zeroFramedArray\n");
    imageReformat(reformatPrettyPrintFile);
    reformatPrettyPrintFile.write("\n\nAfter connected8 pass1, equalArray\n");
    printEqualArray(reformatPrettyPrintFile);

    // 2
    connect8Pass2();
    debugFile.write("After connected8 pass2, newLabel = " + newLabel);
    reformatPrettyPrintFile.write("\n\nAfter connected8 pass2\n");
    reformatPrettyPrintFile.write("\n\nAfter connected8 pass2, zeroFramedArray\n");
    imageReformat(reformatPrettyPrintFile);
    reformatPrettyPrintFile.write("\n\nAfter connected8 pass2, equalArray\n");
    printEqualArray(reformatPrettyPrintFile);

    // 3
    trueNumConnectedComponent = manageEqualArray();

```

```

        reformatPrettyPrintFile.write("\n\nAfter connected8 manageEqualArray(), equalArray\n");
        printEqualArray(reformatPrettyPrintFile);
        newMin = 0;
        newMax = trueNumConnectedComponent;
        connectedComponentProperty = new Property[newMax + 1];
        debugFile.write("\nIn connected8, after manageEqualArray, trueNumConnectedComponent = "
            + trueNumConnectedComponent + "\n");

        // 4
        connectPass3(debugFile);

        reformatPrettyPrintFile.write("\n\nAfter connected8 pass3, zeroFramedArray\n");
        imageReformat(reformatPrettyPrintFile);
        reformatPrettyPrintFile.write("\n\nAfter connected8 pass3, equalArray\n");
        printEqualArray(reformatPrettyPrintFile);

        debugFile.write("Leaving connected8 method\n");
    }

    void connect8Pass1(){
        // zeroFramedArray, newLabel, equalArray
        int[] neighbors;
        int minLabel = 0;
        for (int i = 1; i < numRows + 1; i++) {
            for (int j = 1; j < numCols + 1; j++) {
                if (zeroFramedArray[i][j] == 0)
                    continue;
                neighbors = new int[]{
                    zeroFramedArray[i-1][j-1],
                    zeroFramedArray[i-1][j],
                    zeroFramedArray[i-1][j+1],
                    zeroFramedArray[i][j-1]
                };
                if(neighbors[0] == 0 && neighbors[1] == 0 && neighbors[2] == 0 && neighbors[3] ==
0){

```



```

        zeroFramedArray[i][j] = ++newLabel;
    }
    else if(neighbors[0] == neighbors[1] && neighbors[0] == neighbors[2] &&
neighbors[0] == neighbors[3]){
        zeroFramedArray[i][j] = neighbors[0];
    }
    else{
        // find minLabel
        Arrays.sort(neighbors);
        for(int label: neighbors){
            if(label != 0){
                minLabel = label;
                break;
            }
        }
        zeroFramedArray[i][j] = minLabel;

        // update equalArray
        for(int label: neighbors){
            if(label != 0){
                equalArray[label] = minLabel;
            }
        }
    }
}
}
}

```

```

void connect8Pass2(){
    // zeroFramedArray, equalArray
    int[] neighbors;
    int minLabel = 0;
    for (int i = numRows; i > 0 ; i--) {
        for (int j = numCols; j > 0 ; j--) {

```

```

        if(zeroFramedArray[i][j] == 0)
            continue;
        neighbors = new int[]{
            zeroFramedArray[i][j+1],
            zeroFramedArray[i+1][j-1],
            zeroFramedArray[i+1][j],
            zeroFramedArray[i+1][j+1]
        };

        // do nothing if neighbors are 0 or have same label,
        // but we only have to check if they have same label bc the latter includes the
former
        if(!(neighbors[0] == neighbors[1] && neighbors[0] == neighbors[2] && neighbors[0]
== neighbors[3])){
            Arrays.sort(neighbors);
            for(int label: neighbors){
                if(label != 0){
                    minLabel = label;
                    break;
                }
            }

            equalArray[zeroFramedArray[i][j]] = minLabel;
            zeroFramedArray[i][j] = minLabel;
        }
    }
}

}

void connect4Pass1(){
    // zeroFramedArray, newLabel, equalArray
    int[] neighbors;
    int minLabel = 0;
    for (int i = 1; i < numRows + 1; i++) {
        for (int j = 1; j < numCols + 1; j++) {

```

```

        if (zeroFramedArray[i][j] == 0)
            continue;
        neighbors = new int[]{
            zeroFramedArray[i-1][j],
            zeroFramedArray[i][j-1]
        };
        if(neighbors[0] == 0 && neighbors[1] == 0){
            zeroFramedArray[i][j] = ++newLabel;
        }
        else if(neighbors[0] == neighbors[1]){
            zeroFramedArray[i][j] = neighbors[0];
        }
        else{
            // find minLabel
            Arrays.sort(neighbors);
            minLabel = neighbors[0];
            if(neighbors[0] == 0){
                minLabel = neighbors[1];
            }
            zeroFramedArray[i][j] = minLabel;

            // update equalArray
            for(int label: neighbors){
                if(label != 0){
                    equalArray[label] = minLabel;
                }
            }
        }
    }
}

void connect4Pass2(){

```

```

// zeroFramedArray, equalArray
int[] neighbors;
int minLabel = 0;
for (int i = numRows; i > 0 ; i--) {
    for (int j = numCols; j > 0 ; j--) {
        if(zeroFramedArray[i][j] == 0)
            continue;
        neighbors = new int[]{
            zeroFramedArray[i][j+1],
            zeroFramedArray[i+1][j]
        };

        // do nothing if neighbors are 0 or have same label,
        // but we only have to check if they have same label bc the latter includes the
former
        if(!(neighbors[0] == neighbors[1])){
            Arrays.sort(neighbors);
            minLabel = neighbors[0];
            if(neighbors[0] == 0){
                minLabel = neighbors[1];
            }

            equalArray[zeroFramedArray[i][j]] = minLabel;
            zeroFramedArray[i][j] = minLabel;
        }
    }
}
}
}

```

```

void connectPass3(FileWriter debugFile) throws IOException {
    debugFile.write("Entering connectPass3()\n");
    for (int i = 1; i < connectedComponentProperty.length; i++) {
        connectedComponentProperty[i] = new Property(
            i,
            0,

```

```

        numRows,
        numCols,
        0,
        0
    );
}

Property property;
for (int r = 1; r <= numRows; r++) {
    for (int c = 1; c <= numCols; c++) {
        if (zeroFramedArray[r][c] == 0)
            continue;
        // relabelling
        zeroFramedArray[r][c] = equalArray[zeroFramedArray[r][c]];
        // property at index i associate with group i
        property = connectedComponentProperty[zeroFramedArray[r][c]];
        property.numPixels++;
        if(r < property.minR)
            property.minR = r;
        if(r > property.maxR)
            property.maxR = r;
        if(c < property.minC)
            property.minC = c;
        if(c > property.maxC)
            property.maxC = c;
    }
}

debugFile.write("Leaving connectedPass3()\n");
}

void updateEqualArray(){

}

int manageEqualArray(){

```

```

    int readLabel = 0;
    for (int i = 1; i <= newLabel; i++) {
        if(i == equalArray[i])
            equalArray[i] = ++readLabel;
        else
            equalArray[i] = equalArray[equalArray[i]];
    }
    return readLabel;
}

void printConnectedComponentProperty(FileWriter outFile) throws IOException{
    outFile.write(numRows + " " + numCols + " " + newMin + " " + newMax + "\n");
    outFile.write(trueNumConnectedComponent + "\n");
    for (int i = 1; i <= trueNumConnectedComponent; i++) {
        outFile.write(connectedComponentProperty[i].label + "\n"
            + connectedComponentProperty[i].numPixels + "\n"
            + (connectedComponentProperty[i].minR - 1) + " "
            + (connectedComponentProperty[i].minC - 1) + "\n"
            + (connectedComponentProperty[i].maxR - 1) + " "
            + (connectedComponentProperty[i].maxC - 1) + "\n");
    }
}

void printEqualArray(FileWriter outFile) throws IOException{
    for (int i = 1; i <= newLabel; i++) {
        outFile.write(equalArray[i] + " ");
    }
    outFile.write("\n");
}

void drawBoxes(FileWriter debugFile) throws IOException{
    debugFile.write("Entering drawBoxes()\n");
}

```

```

        int label;

        for (int i = 1; i <= trueNumConnectedComponent; i++) {
            label = connectedComponentProperty[i].label;

            for (int r = connectedComponentProperty[i].minR; r <=
connectedComponentProperty[i].maxR; r++) {
                zeroFramedArray[r][connectedComponentProperty[i].minC] = label;
                zeroFramedArray[r][connectedComponentProperty[i].maxC] = label;
            }

            for (int c = connectedComponentProperty[i].minC; c <=
connectedComponentProperty[i].maxC; c++) {
                zeroFramedArray[connectedComponentProperty[i].minR][c] = label;
                zeroFramedArray[connectedComponentProperty[i].maxR][c] = label;
            }
        }

        debugFile.write("Leaving drawBoxes()\n");
    }

    void printImage(FileWriter outFile) throws IOException{
        outFile.write(numRows + " " + numCols + " " + newMin + " " + newMax + "\n");

        for (int i = 1; i < numRows + 1; i++) {
            for (int j = 1; j < numCols + 1; j++) {
                outFile.write(zeroFramedArray[i][j] + " ");
            }

            outFile.write("\n");
        }

    }

}

public class LiuJ_Project4_Main {
    public static void main(String[] args) throws IOException {
        Scanner inFile;

        String connectness = args[1];

```

```

FileWriter reformatPrettyPrintFile, labelFile, propertyFile, debugFile;

try{
    inFile = new Scanner(new FileReader(args[0]));
    reformatPrettyPrintFile = new FileWriter(args[2]);
    labelFile = new FileWriter(args[3]);
    propertyFile = new FileWriter(args[4]);
    debugFile = new FileWriter(args[5]);

}catch (IOException exception){
    System.out.println("Invalid input");
    return;
}

ConnectedComponentLabel connectedComponentLabel = new ConnectedComponentLabel(inFile);

switch (connectness){
    case "4":
        connectedComponentLabel.connected4(reformatPrettyPrintFile, debugFile);
        break;

    case "8":
        connectedComponentLabel.connected8(reformatPrettyPrintFile, debugFile);
        break;

    default:
        System.out.println("Invalid connectness input. Should be either '4' or '8'");
        return;
}

connectedComponentLabel.printImage(labelFile);
connectedComponentLabel.printConnectedComponentProperty(propertyFile);
connectedComponentLabel.drawBoxes(debugFile);

```



```

reformatPrettyPrintFile.write("\n\nAfter connectedComponentLabel.drawBoxes()\n");
connectedComponentLabel.imageReformat(reformatPrettyPrintFile);
reformatPrettyPrintFile.write("\nTrue Number of Connected Component: "
                                + connectedComponentLabel.trueNumConnectedComponent +
'\n');

inFile.close();
reformatPrettyPrintFile.close();
labelFile.close();
propertyFile.close();
debugFile.close();
}
}

```

Program Output

RFprettyPrintFile for 8-connectness on data 1

After connected8 pass1, zeroFramedArray

```

1 1 . 2 . . 3 . 4 .
. 1 . 2 2 . 3 . 4 .
. 1 . . 2 . 3 . 4 .

```

```

1 1 . . 2 . 3 . 4 4
1 . 1 1 . . 3 . 4 .
. . . . 1 1 1 1 1 .
. . 5 . . . . 1 . 1
6 5 5 5 . . 1 . 1 .
5 . 5 . 5 1 1 1 . .
. . . . . 1 . 1 . .

```

After connected8 pass1, equalArray

```

1 1 1 1 1 5

```

After connected8 pass2

After connected8 pass2, zeroFramedArray

```

1 1 . 1 . . 1 . 1 .
. 1 . 1 1 . 1 . 1 .
. 1 . . 1 . 1 . 1 .
1 1 . . 1 . 1 . 1 1
1 . 1 1 . . 1 . 1 .
. . . . 1 1 1 1 1 .
. . 1 . . . . 1 . 1
1 1 1 1 . . 1 . 1 .
5 . 5 . 1 1 1 1 . .

```

. 1 . 1 . .

After connected8 pass2, equalArray

1 1 1 1 1 1

After connected8 manageEqualArray(), equalArray

1 1 1 1 1 1

After connected8 pass3, zeroFramedArray

1 1 . 1 . . 1 . 1 .
. 1 . 1 1 . 1 . 1 .
. 1 . . 1 . 1 . 1 .
1 1 . . 1 . 1 . 1 1
1 . 1 1 . . 1 . 1 .
. . . . 1 1 1 1 1 .
. . 1 1 . 1
1 1 1 1 . . 1 . 1 .
1 . 1 . 1 1 1 1 . .
. 1 . 1 . .

After connected8 pass3, equalArray

1 1 1 1 1 1

After connectedComponentLabel.drawBoxes()

```
1 1 1 1 1 1 1 1 1 1
1 1 . 1 1 . 1 . 1 1
1 1 . . 1 . 1 . 1 1
1 1 . . 1 . 1 . 1 1
1 . 1 1 . . 1 . 1 1
1 . . . 1 1 1 1 1 1
1 . 1 . . . . 1 . 1
1 1 1 1 . . 1 . 1 1
1 . 1 . 1 1 1 1 . 1
1 1 1 1 1 1 1 1 1 1
```

True Number of Connected Component: 1

Label File for 8-connectness on data 1

```
10 10 0 1
1 1 0 1 0 0 1 0 1 0
0 1 0 1 1 0 1 0 1 0
0 1 0 0 1 0 1 0 1 0
1 1 0 0 1 0 1 0 1 1
1 0 1 1 0 0 1 0 1 0
```

```
0 0 0 0 1 1 1 1 1 0
0 0 1 0 0 0 0 1 0 1
1 1 1 1 0 0 1 0 1 0
1 0 1 0 1 1 1 1 0 0
0 0 0 0 0 1 0 1 0 0
```

Property File for 8-connectness on data1

```
10 10 0 1
1
1
47
0 0
9 9
```

Debug File for 8-connectness on data1

```
Entering connect8 method
After connected8 pass1, newLabel = 6
After connected8 pass2, newLabel = 6
In connected8, after manageEqualArray, trueNumConnectedComponent
= 1
Entering connectPass3()
Leaving connectedPass3()
Leaving connected8 method
Entering drawBoxes()
Leaving drawBoxes()
```

RFprettyPrintFile for 4-connectness run on data1

After connected4 pass1, zeroFramedArray

```
1 1 . 2 . . 3 . 4 .  
. 1 . 2 2 . 3 . 4 .  
. 1 . . 2 . 3 . 4 .  
5 1 . . 2 . 3 . 4 4  
5 . 6 6 . . 3 . 4 .  
. . . . 7 7 3 3 3 .  
. . 8 . . . . 3 . 9  
10 10 8 8 . . 11 . 12 .  
10 . 8 . 13 13 11 11 . .  
. . . . . 13 . 11 . .
```

After connected4 pass1, equalArray

```
1 2 3 3 5 6 3 8 9 10 11 12 13
```

After connected4 pass2

After connected4 pass2, zeroFramedArray

```

1 1 . 2 . . 3 . 3 .
. 1 . 2 2 . 3 . 3 .
. 1 . . 2 . 3 . 3 .
1 1 . . 2 . 3 . 3 4
5 . 6 6 . . 3 . 3 .
. . . . 3 3 3 3 3 .
. . 8 . . . . 3 . 9
8 8 8 8 . . 11 . 12 .
10 . 8 . 11 11 11 11 . .
. . . . . 13 . 11 . .

```

After connected4 pass2, equalArray

```
1 2 3 3 1 6 3 8 9 8 11 12 11
```

After connected4 manageEqualArray(), equalArray

```
1 2 3 3 1 4 3 5 6 5 7 8 7
```

After connected4 pass3, zeroFramedArray

```

1 1 . 2 . . 3 . 3 .
. 1 . 2 2 . 3 . 3 .
. 1 . . 2 . 3 . 3 .
1 1 . . 2 . 3 . 3 3
1 . 4 4 . . 3 . 3 .

```

....3333.

..5....3.6

5555..7.8.

5.5.7777..

.....7.7..

After connected4 pass3, equalArray

1 2 3 3 1 4 3 5 6 5 7 8 7

After connectedComponentLabel.drawBoxes()

1 1 . 2 3 3 3 3 3

1 1 . 2 3 . 3 . 3 3

1 1 . 2 3 . 3 . 3 3

1 1 . 2 3 . 3 . 3 3

1 1 4 4 3 . 3 . 3 3

....333333

5555333336

555577778.

55557777..

....7777..

True Number of Connected Component: 8

labelFile for 4-connectness run on data1

10 10 0 8
1 1 0 2 0 0 3 0 3 0
0 1 0 2 2 0 3 0 3 0
0 1 0 0 2 0 3 0 3 0
1 1 0 0 2 0 3 0 3 3
1 0 4 4 0 0 3 0 3 0
0 0 0 0 3 3 3 3 3 0
0 0 5 0 0 0 0 3 0 6
5 5 5 5 0 0 7 0 8 0
5 0 5 0 7 7 7 7 0 0
0 0 0 0 0 7 0 7 0 0

- propertyFile for 4-connectness run on data1

10 10 0 8
8
1
7
0 0
4 1
2
5

0 3

3 4

3

17

0 4

6 9

4

2

4 2

4 3

5

7

6 0

8 3

6

1

6 9

6 9

7

7

7 4

9 7

8

1

7 8

7 8

- deBugFile for 4-connectness on data 1

Entering connect4 method

After connected4 pass1, newLabel = 13

After connected4 pass2, newLabel = 13
In connected4, after manageEqualArray,
trueNumConnectedComponent = 8

Entering connectPass3()

Leaving connectedPass3()

Leaving connected4 method

Entering drawBoxes()

Leaving drawBoxes()

- RFprettyPrintFile for 8-connectness run on data2

After connected8 pass1, zeroFramedArray

```
. . . . .
. . 1 . . . . . 2 . . . . 3 3 . . . 4 . . .
. . 1 . . . . 5 . 6 . . . . 2 2 2 . . . . 3 3 . . 4 4 4 . . .
. 1 1 . . . . 5 5 5 5 . . 2 2 2 2 2 . . . 3 3 . . 4 4 4 . . .
. 1 1 . . . . 5 . 5 5 . . . 2 2 2 2 2 . . . . 3 . . . . .
. . . 1 . . . 5 5 5 . . . 2 2 2 2 2 2 2 . . . 3 3 3 . . . .
. . . . 1 1 1 1 1 . 5 5 2 . . . . . 2 2 . . . 3 3 3 3 3 . .
. . . . 1 . . 1 . 1 1 1 1 1 1 1 1 1 1 1 1 . . . . 3 3 . . .
. . . . 1 1 1 1 1 . . . . . 1 1 1 1 1 1 1 1 . . . . 3 . . . .
. . . . . . 1 . 1 1 1 1 1 1 1 1 1 1 1 1 1 . . . 3 . . . . .
. . 7 7 7 . 1 1 . . . . . 1 . . . . . . 3 . . . 8 .
. . . 7 7 . 1 . . 9 . . . . . 1 1 1 1 1 1 1 1 . . 3 3 . . 8 .
. . 7 . 7 . 1 . 9 9 9 9 9 9 1 1 1 1 1 1 1 1 . . . 3 3 . . 8 .
```

```

. . 7 . 7 . . . . 9 9 9 9 1 1 1 1 1 1 1 1 . . . 3 . . . 8 .
. . 7 . . . . . . 9 9 1 1 1 1 1 1 1 1 . 1 . . . 3 . 8 . .
. . . 7 7 . . . . 9 . . 1 . . . . . 1 . . . . . 3 3 3 .
. 10 . . 7 . . 11 9 . . . . 1 1 1 1 1 . . . . . 3 3 3 .
. 10 10 . 7 . 11 . . . . . 1 1 1 . . 12 . . . . 13 3 3 3 3 .
. . 10 7 7 7 7 7 . . . . . 1 . 1 1 . 12 . . . . 3 3 3 3 3 .
. . 7 7 7 7 7 7 . . . . . 14 . . . . . 12 . . . 3 3 3 3 . .
. . . . 7 7 7 7 . . . . . 14 . 15 . . . 16 . . . 3 3 3 3 . . .
. . . 7 7 . . . . . . 14 . 15 . . . 16 . . . 3 3 3 . . . 17 .
. . 7 7 . . . . . . . 14 14 14 14 . . . 16 . . . 3 3 3 3 3 3 .
. . . . . . . . . . 14 14 14 . 18 16 16 16 16 3 3 . . . . .

```

After connected8 pass1, equalArray

```
1 1 3 4 1 5 7 3 9 7 7 12 3 14 14 3 3 16
```

After connected8 pass2

After connected8 pass2, zeroFramedArray

```

. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
. . 1 . . . . . . . . . 1 . . . . 3 3 . . . 4 . . . .
. . 1 . . . . 1 . 1 . . . . 1 2 1 . . . . 3 3 . . 4 4 4 . . .
. 1 1 . . . . 1 1 1 1 . . 1 1 1 1 1 . . . 3 3 . . 4 4 4 . . .
. 1 1 . . . . 1 . 1 1 . . . 1 2 1 2 1 . . . . 3 . . . . .

```

```

... 1 ... 5 1 1 ... 1 1 1 1 1 1 1 ... 3 3 3 ...
... 1 1 1 1 1 . 1 5 1 ... 2 1 ... 3 3 3 3 3 ...
... 1 . . 1 . 1 1 1 1 1 1 1 1 1 1 1 1 ... 3 3 ...
... 1 1 1 1 1 ... 1 1 1 1 1 1 1 ... 3 ...
... . . 1 . 1 1 1 1 1 1 1 1 1 1 1 1 ... 3 ...
.. 7 7 7 . 1 1 ... 1 ... 3 ... 3 .
... 7 7 . 1 . . 1 ... 1 1 1 1 1 1 1 1 . 3 3 . 3 .
.. 7 . 7 . 1 . 1 1 9 1 1 1 1 1 1 1 1 1 1 ... 3 3 . 3 .
.. 7 . 7 ... 1 1 1 9 1 1 1 1 1 1 1 1 1 ... 3 ... 3 .
.. 7 ... 1 1 1 1 1 1 1 1 1 1 1 . 1 ... 3 . 3 ...
... 7 7 ... 9 . . 1 ... 1 ... 3 3 3 .
. 7 . . 7 . . 7 9 ... 1 1 1 1 1 ... 3 3 3 .
. 7 7 . 7 . 7 ... 1 1 1 . . 1 ... 3 3 3 3 3 .
.. 7 7 7 7 7 7 ... 1 . 1 1 . 12 ... 3 3 3 3 3 .
.. 7 7 7 7 7 7 ... 14 ... 12 ... 3 3 3 3 ...
... 7 7 7 7 ... 14 . 14 ... 3 ... 3 3 3 3 ...
... 7 7 ... 14 . 14 ... 3 ... 3 3 3 ... 3 .
.. 7 7 ... 14 14 14 14 ... 3 ... 3 3 3 3 3 3 .
... 14 14 14 . 3 3 3 3 3 3 3 ...

```

After connected8 pass2, equalArray

```
1 1 3 4 1 1 7 3 1 7 7 1 3 14 14 3 3 3
```

After connected8 manageEqualArray(), equalArray

1 1 2 3 1 1 4 2 1 4 4 1 2 5 5 2 2 2

After connected8 pass3, zeroFramedArray

```
.....
..1.....1.....2 2...3....
..1....1.1....1 1 1...2 2...3 3 3...
.1 1....1 1 1 1..1 1 1 1 1...2 2...3 3 3...
.1 1....1.1 1...1 1 1 1 1...2.....
...1...1 1 1...1 1 1 1 1 1 1...2 2 2.....
....1 1 1 1 1.1 1 1.....1 1...2 2 2 2 2..
....1..1.1 1 1 1 1 1 1 1 1 1 1 1...2 2...
....1 1 1 1 1.....1 1 1 1 1 1 1...2....
.....1.1 1 1 1 1 1 1 1 1 1 1 1...2.....
..4 4 4.1 1.....1.....2...2.
...4 4.1..1....1 1 1 1 1 1 1 1...2 2...2.
..4.4.1.1 1 1 1 1 1 1 1 1 1 1 1 1...2 2...2.
..4.4....1 1 1 1 1 1 1 1 1 1 1 1...2...2.
..4.....1 1 1 1 1 1 1 1 1 1 1.1...2.2..
...4 4....1..1....1.....2 2 2.
.4..4..4 1....1 1 1 1 1.....2 2 2.
.4 4.4.4.....1 1 1..1....2 2 2 2 2.
..4 4 4 4 4 4.....1.1 1.1...2 2 2 2 2.
..4 4 4 4 4 4.....5.....1...2 2 2 2..
....4 4 4 4.....5.5...2....2 2 2 2...
...4 4.....5.5...2...2 2 2...2.
```

..44.....5555...2...222222.
.....555.2222222.....

After connected8 pass3, equalArray

1 1 2 3 1 1 4 2 1 4 4 1 2 5 5 2 2 2

After connectedComponentLabel.drawBoxes()

.....
.1111111111111111111222222233322.
.11...1.1...111.2..21..333.2.
.11...1111..111112..21..333.2.
.11...1.11...11112...12....2.
.1.1...111...1111121..1222...2.
.1..11111.111.....211.1.222222.
.1..1..1.11111111121111...22.2.
.1..11111.....11121111...2..2.
.1.....1.11111111121111..2...2.
.4444444.....1..2...1..2...2.
.4.44.14.1.....11121111..22..2.
.44.4.14111111111121111..22..2.
.44.4..4.11111111121111..2...2.
.44....4..11111111211.1...2.22.
.4.44..4.1..1.....2...1....222.
.4..4..41....111112...1....222.

```
.44.4.44.....111.21..1..22222.
.4444444.....1.12.1.1..22222.
.4111114111115555121111..22222.
.4..4444.....5.55.22....2222.2.
.4.44..4.....5.55.22...222...2.
.4444444.....5555.2.2...222222.
.....5555.22222222222222.
```

True Number of Connected Component: 5

- labelFile for 8-connectness run on data2

```
24 31 0 5
000000000000000000000000000000000000
00100000000000001000002200030000
0010000101000011100002200333000
0110000111100111110002200333000
0110000101100011111000020000000
0001000111000111111100022200000
0000111110111000000110002222200
0000100101111111111111000022000
0000111110000001111111000020000
0000000101111111111111000200000
0044401100000001000000000200020
0004401001000001111111100220020
0040401011111111111111000220020
0040400001111111111111000200020
```


0040000000111111111110100020200
0004400001001000001000000002220
0400400410000111110000000002220
0440404000000011100100000222220
0044444400000001011010000222220
0044444400000500000001000222200
0000444400000505000200002222000
0004400000000505000200022200020
004400000000055500020002222220
0000000000000055502222222000000

- propertyFile for 8-connectness run on r data2
24 31 0 5

5
1
162
1 1
19 22
2
75
1 18
23 29
3
7
1 25
3 27
4

39

10 1

22 7

5

12

19 13

23 16

debugFile

Entering connect8 method

After connected8 pass1, newLabel = 18

After connected8 pass2, newLabel = 18

In connected8, after manageEqualArray, trueNumConnectedComponent
= 5

Entering connectPass3()

Leaving connectedPass3()

Leaving connected8 method

Entering drawBoxes()

Leaving drawBoxes()

- RFprettyPrintFile for 4-connectness run on data2

After connected4 pass1, zeroFramedArray

.
 . . 1 2 3 3 . . . 4
 . . 1 5 . 6 7 2 2 3 3 . . 8 4 4 . . .
 . 9 1 5 5 5 5 . . 10 7 2 2 2 . . . 3 3 . . 8 4 4 . . .
 . 9 1 5 . 5 5 . . . 7 2 2 2 2 11
 . . . 12 . . . 5 5 5 . . . 13 7 2 2 2 2 2 . . . 11 11 11
 14 14 14 5 5 . 15 15 15 2 2 . . . 11 11 11 11 11 . .
 14 . . 5 . 16 15 15 15 15 15 15 15 15 2 2 2 11 11 . . .
 14 14 14 5 5 15 15 15 15 2 2 2 11
 5 . 17 17 17 17 17 17 15 15 15 15 2 2 2 . . . 18
 . . 19 19 19 . 20 5 15 18 . . . 21 .
 . . . 19 19 . 20 . . 22 15 15 15 15 15 15 15 15 . . 18 18 . . 21 .
 . . 23 . 19 . 20 . 24 22 22 22 22 22 22 15 15 15 15 15 15 15 . . . 18 18 . . 21 .
 . . 23 . 19 22 22 22 22 22 22 15 15 15 15 15 15 15 . . . 18 . . . 21 .
 . . 23 22 22 22 22 22 15 15 15 15 15 15 . 25 . . . 26 . 27 . .
 . . . 28 28 29 . . 22 15 30 27 27 .
 . 31 . . 28 . . 32 32 33 33 33 33 33 30 27 27 .
 . 31 31 . 28 . 34 33 33 33 . . 35 36 36 30 27 27 .
 . . 31 31 28 28 28 28 33 . 37 37 . 38 36 36 30 27 27 .
 . . 31 31 28 28 28 28 39 40 . . . 36 36 30 27 . .
 28 28 28 28 39 . 41 . . . 42 43 36 36 30 . . .
 . . . 44 28 39 . 41 . . . 42 . . . 45 43 36 . . . 46 .
 . . 47 44 39 39 39 39 . . . 48 . . . 43 36 36 36 36 36 .
 39 39 39 . 49 49 48 48 48 48 43

After connected4 pass1, equalArray

1 2 3 4 5 5 2 4 1 7 11 12 7 5 15 15 15 18 19 20 21 22 23 22 25 26 27 28 29 30 28 32 33 28 35 36
37 38 39 40 39 42 43 44 43 36 44 43 48

After connected4 pass2

After connected4 pass2, zeroFramedArray

.
. . 1 2 3 3 . . . 4
. . 1 5 . 5 7 2 2 3 3 . . 8 4 4 . . .
. 9 1 5 5 5 5 . . 2 2 2 2 2 . . . 3 3 . . 4 4 4 . . .
. 1 1 5 . 5 5 . . . 7 2 2 2 2 11
. . . 12 . . . 5 5 5 . . . 2 2 2 2 2 2 2 . . . 11 11 11
. . . . 14 5 5 5 5 . 2 15 2 2 2 . . . 11 11 11 11 11 . .
. . . . 5 . . 5 . 2 2 2 2 2 2 2 15 15 2 2 2 2 11 11 . . .
. . . . 5 5 5 5 5 2 15 2 15 2 2 2 11
. 5 . 2 2 2 2 2 2 2 2 2 2 2 2 2 . . . 18
. . 19 19 19 . 5 5 15 18 . . . 21 .
. . . 19 19 . 20 . . 15 15 15 15 15 15 15 15 . . 18 18 . . 21 .
. . 23 . 19 . 20 . 15 15 15 15 22 22 15 15 15 15 15 15 15 . . . 18 18 . . 21 .
. . 23 . 19 22 22 15 22 15 22 15 15 15 15 15 15 . . . 18 . . . 21 .
. . 23 15 15 15 15 15 15 15 15 15 15 . 25 . . . 26 . 27 . .
. . . 28 28 29 . . 22 15 27 27 27 .
. 28 . . 28 . . 32 32 33 33 33 33 33 30 27 27 .

```

. 28 28 . 28 . 28 . . . . . 33 33 33 . . 35 . . . . . 36 36 27 27 27 .
. . 28 31 28 28 28 28 . . . . . 33 . 37 37 . 38 . . . . 36 27 30 27 27 .
. . 28 28 28 28 28 28 . . . . . 39 . . . . . 40 . . . 27 27 27 27 . .
. . . . 28 28 28 28 . . . . . 39 . 39 . . . 42 . . . . 30 30 30 30 . . .
. . . 28 28 . . . . . 39 . 39 . . . 42 . . . 43 43 36 . . . 36 .
. . 44 44 . . . . . 39 39 39 39 . . . 43 . . . 36 36 36 36 36 36 .
. . . . . 39 39 39 . 43 43 43 43 43 43 43 . . . . .

```

After connected4 pass2, equalArray

```

1 2 3 4 5 5 2 4 1 2 11 12 2 5 2 2 2 18 19 5 21 15 23 15 25 26 27 28 29 27 28 32 33 28 35 27 37
38 39 40 39 42 30 28 43 36 44 43 43

```

After connected4 manageEqualArray(), equalArray

```

1 2 3 4 5 5 2 4 1 2 6 7 2 5 2 2 2 8 9 5 10 2 11 2 12 13 14 15 16 14 15 17 18 15 19 14 20 21 22 23
22 24 14 15 14 14 15 14 14

```

After connected4 pass3, zeroFramedArray

```

. . . . .
. . 1 . . . . . 2 . . . . 3 3 . . . 4 . . . .
. . 1 . . . . 5 . 5 . . . . 2 2 2 . . . . 3 3 . . 4 4 4 . . .
. 1 1 . . . . 5 5 5 5 . . 2 2 2 2 2 . . . 3 3 . . 4 4 4 . . .
. 1 1 . . . . 5 . 5 5 . . . 2 2 2 2 2 . . . . 6 . . . . .
. . . 7 . . . 5 5 5 . . . 2 2 2 2 2 2 2 . . . 6 6 6 . . . .
. . . . 5 5 5 5 5 . 2 2 2 . . . . . 2 2 . . . 6 6 6 6 6 . .

```

. . . . 5 . . 5 . 2 2 2 2 2 2 2 2 2 2 2 2 6 6 . . .
 5 5 5 5 5 2 2 2 2 2 2 2 6
 5 . 2 2 2 2 2 2 2 2 2 2 2 2 2 . . . 8
 . . 9 9 9 . 5 5 2 8 . . . 10 .
 . . . 9 9 . 5 . . 2 2 2 2 2 2 2 2 2 . . 8 8 . . 10 .
 . . 11 . 9 . 5 . 2 2 2 2 2 2 2 2 2 2 2 2 2 2 . . . 8 8 . . 10 .
 . . 11 . 9 2 2 2 2 2 2 2 2 2 2 2 2 2 . . . 8 . . . 10 .
 . . 11 2 2 2 2 2 2 2 2 2 2 2 2 . 12 . . . 13 . 14 . .
 . . . 15 15 16 . . 2 2 14 14 14 .
 . 15 . . 15 . . 17 17 18 18 18 18 18 14 14 14 .
 . 15 15 . 15 . 15 18 18 18 . . 19 14 14 14 14 14 .
 . . 15 15 15 15 15 15 18 . 20 20 . 21 14 14 14 14 14 .
 . . 15 15 15 15 15 15 22 23 . . . 14 14 14 14 . .
 15 15 15 15 22 . 22 . . . 24 14 14 14 14 . . .
 . . . 15 15 22 . 22 . . . 24 . . . 14 14 14 . . . 14 .
 . . 15 15 22 22 22 22 . . . 14 . . . 14 14 14 14 14 14 .
 22 22 22 . 14 14 14 14 14 14 14

After connected4 pass3, equalArray

1 2 3 4 5 5 2 4 1 2 6 7 2 5 2 2 2 8 9 5 10 2 11 2 12 13 14 15 16 14 15 17 18 15 19 14 20 21 22 23
 22 24 14 15 14 14 15 14 14

After connectedComponentLabel.drawBoxes()

.

. 1 1 2 2 2 2 2 2 2 2 2 2 2 2 3 3 . . 4 4 4 . . .
 . 1 1 . 5 5 5 5 5 5 5 . . . 2 2 2 3 3 . . 4 4 4 . . .
 . 1 1 . 5 . . 5 2 5 5 . . 2 2 2 2 2 . . . 3 3 . . 4 4 4 . . .
 . 1 1 . 5 . . 5 2 5 5 . . . 2 2 2 2 2 . . . 2 6 6 6 6 6 6 6 . .
 . . . 7 5 . . 5 2 5 5 . . 2 2 2 2 2 2 2 . . 2 6 6 6 . . 6 . .
 5 5 5 5 2 . 5 2 2 2 2 . 2 6 6 6 6 6 6 6 . .
 5 . . 5 2 2 5 2 2 2 2 2 2 2 2 2 2 2 2 6 . . 6 6 6 . .
 5 5 5 5 2 . 5 2 2 2 2 2 2 2 2 6 6 6 6 6 6 . .
 5 . . 5 2 2 5 2 2 2 2 2 2 2 2 2 2 2 2 . . 8 8
 . . 9 9 9 . 5 5 2 . 5 2 2 . . 8 8 . . 10 .
 . . 9 9 9 . 5 . 2 2 5 2 2 2 2 2 2 2 2 . . 8 8 . . 10 .
 . . 11 . 9 5 5 5 5 5 5 2 2 2 2 2 2 2 2 2 2 2 . . 8 8 . . 10 .
 . . 11 9 9 . . . 2 2 2 2 2 2 2 2 2 2 2 2 2 2 . . 8 8 . . 10 .
 . . 11 2 . 2 2 2 2 2 2 2 2 14 14 14 14 14 14 14 14 14 14 .
 . 15 15 15 15 15 15 15 2 16 2 2 2 2 2 2 2 2 14 2 2 2 2 14 14 14 .
 . 15 . . 15 . . 17 17 18 18 18 18 18 14 14 14 14 .
 . 15 15 . 15 . 15 15 18 18 18 18 18 14 19 14 14 14 14 14 .
 . 15 15 15 15 15 15 15 18 18 18 18 20 20 . 21 14 14 14 14 14 .
 . 15 15 15 15 15 15 15 22 22 22 22 . 14 . . 23 . . . 14 14 14 14 14 .
 . 15 . . 15 15 15 15 22 . 22 22 . 14 24 14 14 14 14 . 14 .
 . 15 . 15 15 . . 15 22 . 22 22 . 14 24 . . . 14 14 14 . . . 14 .
 . 15 15 15 15 15 15 15 22 22 22 22 . 14 . 14 . . . 14 14 14 14 14 14 .
 22 22 22 22 . 14 14 14 14 14 14 14 14 14 14 14 14 .

True Number of Connected Component: 24

- labelFile for 4-connectness run on data2

24 31 0 24

```
000000000000000000000000000000000000
001000000000000002000003300040000
0010000505000022200003300444000
0110000555500222220003300444000
01100005055000222220000600000000
0007000555000222222200066600000
0000555550222000000220006666600
000050050222222222222000066000
0000555550000002222222000060000
0000000502222222222222000800000
00999055000000020000000008000100
00099050020000022222222008800100
001109050222222222222220008800100
0011090000222222222222220008000100
001100000002222222222220120001301400
0001515000016002000002000000001414140
0150015001717000018181818180000000001414140
01515015015000000018181800190000014141414140
00151515151500000001802020021000014141414140
00151515151500000220000000230001414141400
000015151515000002202200024000014141414000
0001515000000002202200024000141414000140
```


0 0 15 15 0 0 0 0 0 0 0 0 22 22 22 22 0 0 0 14 0 0 0 14 14 14 14 14 14 0

0 0 0 0 0 0 0 0 0 0 0 0 0 22 22 22 0 14 14 14 14 14 14 14 0 0 0 0 0 0

- propertyFile for 4-connectness run on data2

24 31 0 24

24

1

6

1 1

4 2

2

109

1 8

15 22

3

6

1 21

3 22

4

7

1 25

3 27

5

29

2 4

12 10

6

12

4 23

8 28

7

1

5 3

5 3

8

7

9 25

13 26

9

7

10 2

13 4

10

4

10 29

13 29

11

3

12 2

14 2

12

1

14 22

14 22

13

1

14 26

14 26

14

43

14 18

23 29

15

28

15 1

22 7

16

1

15 9

15 9

17

2

16 7

16 8

18

9

16 13

18 17

19

1

17 19

17 19

20

2

18 17

18 18

21

1

18 20

18 20

22

12

19 13

23 16

23

1

19 21

19 21

24

2

20 19

21 19

deBugFile

Entering connect4 method

After connected4 pass1, newLabel = 49

After connected4 pass2, newLabel = 49In connected4, after manageEqualArray,
trueNumConnectedComponent = 24

Entering connectPass3()

Leaving connectedPass3()

Leaving connected4 method

Entering drawBoxes()

Leaving drawBoxes()