

CV Project 4 Connected Components
Java
Jonathan Mathew
Project Due 03/19/23

Algorithm Steps

```
step 0: inFile-->open the input file from argv [1]
Connectness-->argv [2] option-->argv [3] RFprettyPrintFile, labelFile, propertyFile, debugFile-->open
from argv []
numRows, numCols, minVal, maxVal-->read from inFile zeroFramedAry-->dynamically allocate.
newLabel --> 0
step 1: zero2D (zeroFramedAry) step 2: loadImage (inFile, zeroFramedAry)
step 3: if option == 'y' or 'Y'
conversion (zeroFramedAry)
step 4: if connectness == 4
connected4 (zeroFramedAry, newLabel, EQAry, RFprettyPrintFile, debugFile)
step 5: if connectness == 8
connected4 (zeroFramedAry, newLabel, EQAry, RFprettyPrintFile, debugFile)
step 6: labelFile-->output numRows, numCols, newMin, newMax to labelFile
step 7: printImg (zeroFramedAry, labelFile) // Output the result of pass3 inside of zeroFramedAry
step 8: printCCproperty (propertyFile) // print cc properties to propertyFile step 9: drawBoxes
(zeroFramedAry, CCproperty, trueNumCC) // draw on zeroFramed image. step 10: imgReformat
(zeroFramedAry, RFprettyPrintFile)
step 11: print trueNumCC to RFprettyPrintFile with proper caption
step 12: close all files
```

CONN 8 Data 1.txt

PrettyPrint.txt

```
1  ----Printing Original Table-----
2
3  After Connected4Pass1, newLabel = 6
4  numRows: 10 numCols: 10 minVal: 0 maxVal: 1
5  1  1  .  2  .  .  3  .  4  .
6  .  1  .  2  2  .  3  .  4  .
7  .  1  .  .  2  .  3  .  4  .
8  1  1  .  .  2  .  3  .  4  4
9  1  .  1  1  .  .  3  .  4  .
10 .  .  .  .  1  1  1  1  1  .
11 .  .  5  .  .  .  .  1  .  1
12 6  5  5  5  .  .  1  .  1  .
13 5  .  5  .  5  1  1  1  .  .
14 .  .  .  .  .  1  .  1  .  .
15
16 -----Printing EQ Table-----
17  1  2  3  4  5  6
18  1  1  1  1  1  5
19
20 After Connected4Pass2, newLabel = 6
21 numRows: 10 numCols: 10 minVal: 0 maxVal: 1
22  1  1  .  1  .  .  1  .  1  .
23  .  1  .  1  1  .  1  .  1  .
24  .  1  .  .  1  .  1  .  1  .
25  1  1  .  .  1  .  1  .  1  1
26  1  .  1  1  .  .  1  .  1  .
27  .  .  .  .  1  1  1  1  1  .
28  .  .  1  .  .  .  .  1  .  1
29  1  1  1  1  .  .  1  .  1  .
30  1  .  1  .  1  1  1  1  .  .
31  .  .  .  .  .  1  .  1  .  .
32
33 -----Printing EQ Table-----
34  1  2  3  4  5  6
35  1  1  1  1  1  1
36
```

```

37  ----Running Managing EQ ARRAY----
38  -----Printing EQ Table-----
39  1    2    3    4    5    6
40  1    1    1    1    1    1
41
42
43  In Connected4, after manage EQarr, trueNumCC = 1
44
45
46  -----After Pass3-----
47  numRows: 10 numCols: 10 minVal: 0 maxVal: 1
48  1    1    .    1    .    .    1    .    1    .
49  .    1    .    1    1    .    1    .    1    .
50  .    1    .    .    1    .    1    .    1    .
51  1    1    .    .    1    .    1    .    1    1
52  1    .    1    1    .    .    1    .    1    .
53  .    .    .    .    1    1    1    1    1    .
54  .    .    1    .    .    .    .    1    .    1
55  1    1    1    1    .    .    1    .    1    .
56  1    .    1    .    1    1    1    1    .    .
57  .    .    .    .    .    1    .    1    .    .
58
59  -----Printing EQ Table-----
60  1    2    3    4    5    6
61  1    1    1    1    1    1
62
63
64  Number of Conected Components: 1

```

```

65
66
67  -----Result of Drawing Boxes-----
68  numRows: 10 numCols: 10 minVal: 0 maxVal: 1
69  1  1  1  1  1  1  1  1  1  1
70  1  1  .  1  1  .  1  .  1  1
71  1  1  .  .  1  .  1  .  1  1
72  1  1  .  .  1  .  1  .  1  1
73  1  .  1  1  .  .  1  .  1  1
74  1  .  .  .  1  1  1  1  1  1
75  1  .  1  .  .  .  .  1  .  1
76  1  1  1  1  .  .  1  .  1  1
77  1  .  1  .  1  1  1  1  .  1
78  1  1  1  1  1  1  1  1  1  1
79

```

Label.txt

```

1  numRows: 10 numCols: 10 minVal: 0 maxVal: 1
2  -----Printing image-----
3  10 10 0 1
4  1  1  .  1  .  .  1  .  1  .
5  .  1  .  1  1  .  1  .  1  .
6  .  1  .  .  1  .  1  .  1  .
7  1  1  .  .  1  .  1  .  1  1
8  1  .  1  1  .  .  1  .  1  .
9  .  .  .  .  1  1  1  1  1  .
10 .  .  1  .  .  .  .  1  .  1
11 1  1  1  1  .  .  1  .  1  .
12 1  .  1  .  1  1  1  1  .  .
13 .  .  .  .  .  1  .  1  .  .
14

```

Property.txt

```
1  CCProperties
2  10 10 0 1
3  1
4  1
5  47
6  0 0
7  9 9
8  |
```

Debug.txt

```
1  |-----entering connected 8 method-----
2  After Connected8Pass1, newLabel = 6
3  numRows: 10 numCols: 10 minVal: 0 maxVal: 1
4  1 1 . 2 . . 3 . 4 .
5  . 1 . 2 2 . 3 . 4 .
6  . 1 . . 2 . 3 . 4 .
7  1 1 . . 2 . 3 . 4 4
8  1 . 1 1 . . 3 . 4 .
9  . . . . 1 1 1 1 1 .
10 . . 5 . . . . 1 . 1
11 6 5 5 5 . . 1 . 1 .
12 5 . 5 . 5 1 1 1 . .
13 . . . . . 1 . 1 . .
14
15 -----Printing EQ Table-----
16 1 2 3 4 5 6
17 1 1 1 1 1 5
18
19 After Connected8Pass2, newLabel = 6
20 numRows: 10 numCols: 10 minVal: 0 maxVal: 1
21 1 1 . 1 . . 1 . 1 .
22 . 1 . 1 1 . 1 . 1 .
23 . 1 . . 1 . 1 . 1 .
24 1 1 . . 1 . 1 . 1 1
25 1 . 1 1 . . 1 . 1 .
26 . . . . 1 1 1 1 1 .
27 . . 1 . . . . 1 . 1
28 1 1 1 1 . . 1 . 1 .
29 1 . 1 . 1 1 1 1 . .
30 . . . . . 1 . 1 . .
31
32 -----Printing EQ Table-----
33 1 2 3 4 5 6
34 1 1 1 1 1 1
```

```

36  ----Running Managing EQ ARRAY----
37  -----Printing EQ Table-----
38  1   2   3   4   5   6
39  1   1   1   1   1   1
40
41  In Connected8, after manageEQarr, trueNumCC = 1
42  Entering connectPass3 Method
43  ---leaving connectPass3 Method---
44
45
46  -----After Pass3-----
47  numRows: 10 numCols: 10 minVal: 0 maxVal: 1
48  1   1   .   1   .   .   1   .   1   .
49  .   1   .   1   1   .   1   .   1   .
50  .   1   .   .   1   .   1   .   1   .
51  1   1   .   .   1   .   1   .   1   1
52  1   .   1   1   .   .   1   .   1   .
53  .   .   .   .   1   1   1   1   1   .
54  .   .   1   .   .   .   .   1   .   1
55  1   1   1   1   .   .   1   .   1   .
56  1   .   1   .   1   1   1   1   .   .
57  .   .   .   .   .   1   .   1   .   .
58
59  -----Printing EQ Table-----
60  1   2   3   4   5   6
61  1   1   1   1   1   1
62
63  Leaving connected8 method
64
65  -----Result of Drawing Boxes-----
66  numRows: 10 numCols: 10 minVal: 0 maxVal: 1
67  1   1   1   1   1   1   1   1   1   1
68  1   1   .   1   1   .   1   .   1   1
69  1   1   .   .   1   .   1   .   1   1
70  1   1   .   .   1   .   1   .   1   1
71  1   .   1   1   .   .   1   .   1   1

```

CONN 4 Data 1.txt

PrettyPrint.txt

```
1  ----Printing Original Table----
2  numRows: 10 numCols: 10 minVal: 0 maxVal: 1
3  1 1 . 1 . . 1 . 1 .
4  . 1 . 1 1 . 1 . 1 .
5  . 1 . . 1 . 1 . 1 .
6  1 1 . . 1 . 1 . 1 1
7  1 . 1 1 . . 1 . 1 .
8  . . . . 1 1 1 1 1 .
9  . . 1 . . . . 1 . 1
10 1 1 1 1 . . 1 . 1 .
11 1 . 1 . 1 1 1 1 . .
12 . . . . . 1 . 1 . .
13
14 After Connected4Pass1, newLabel = 13
15 numRows: 10 numCols: 10 minVal: 0 maxVal: 1
16 1 1 . 2 . . 3 . 4 .
17 . 1 . 2 2 . 3 . 4 .
18 . 1 . . 2 . 3 . 4 .
19 5 1 . . 2 . 3 . 4 4
20 5 . 6 6 . . 3 . 4 .
21 . . . . 7 7 3 3 3 .
22 . . 8 . . . . 3 . 9
23 10 10 8 8 . . 11 . 12 .
24 10 . 8 . 13 13 11 11 . .
25 . . . . . 13 . 11 . .
26
27 ----Printing EQ Table----
28 1 2 3 4 5 6 7 8 9 10 11 12 13
29 1 2 3 3 1 6 3 8 9 8 11 12 11
30
```

```

31 After Connected4Pass2, newLabel = 13
32 numRows: 10 numCols: 10 minVal: 0 maxVal: 1
33 1 1 . 2 . . 3 . 3 .
34 . 1 . 2 2 . 3 . 3 .
35 . 1 . . 2 . 3 . 3 .
36 1 1 . . 2 . 3 . 3 3
37 1 . 6 6 . . 3 . 3 .
38 . . . . 3 3 3 3 3 .
39 . . 8 . . . . 3 . 9
40 8 8 8 8 . . 11 . 12 .
41 8 . 8 . 11 11 11 11 . .
42 . . . . . 11 . 11 . .
43
44 -----Printing EQ Table-----
45 1 2 3 4 5 6 7 8 9 10 11 12 13
46 1 2 3 3 1 6 3 8 9 8 11 12 11
47
48 ----Running Managing EQ ARRAY----
49 -----Printing EQ Table-----
50 1 2 3 4 5 6 7 8 9 10 11 12 13
51 1 2 3 3 1 4 3 5 6 5 7 8 7
52
53
54 In Connected4, after manage EQarr, trueNumCC = 8

```



```

57  -----After Pass3-----
58  numRows: 10 numCols: 10 minVal: 0 maxVal: 1
59  1  1  .  2  .  .  3  .  3  .
60  .  1  .  2  2  .  3  .  3  .
61  .  1  .  .  2  .  3  .  3  .
62  1  1  .  .  2  .  3  .  3  3
63  1  .  4  4  .  .  3  .  3  .
64  .  .  .  .  3  3  3  3  3  .
65  .  .  5  .  .  .  .  3  .  6
66  5  5  5  5  .  .  7  .  8  .
67  5  .  5  .  7  7  7  7  .  .
68  .  .  .  .  .  7  .  7  .  .
69
70  -----Printing EQ Table-----
71  1  2  3  4  5  6  7  8  9  10  11  12  13
72  1  2  3  3  1  4  3  5  6  5  7  8  7
73
74
75  Number of Conected Components: 8
76
77
78  -----Result of Drawing Boxes-----
79  numRows: 10 numCols: 10 minVal: 0 maxVal: 1
80  1  1  .  2  3  3  3  3  3  3
81  1  1  .  2  3  .  3  .  3  3
82  1  1  .  2  3  .  3  .  3  3
83  1  1  .  2  3  .  3  .  3  3
84  1  1  4  4  3  .  3  .  3  3
85  .  .  .  .  3  3  3  3  3  3
86  5  5  5  5  3  3  3  3  3  6
87  5  5  5  5  7  7  7  7  8  .
88  5  5  5  5  7  7  7  7  .  .
89  .  .  .  .  7  7  7  7  .  .

```

Label.txt

```
1  numRows: 10 numCols: 10 minVal: 0 maxVal: 1
2  -----Printing image-----
3  10 10 0 1
4  1 1 . 2 . . 3 . 3 .
5  . 1 . 2 2 . 3 . 3 .
6  . 1 . . 2 . 3 . 3 .
7  1 1 . . 2 . 3 . 3 3
8  1 . 4 4 . . 3 . 3 .
9  . . . . 3 3 3 3 3 .
10 . . 5 . . . . 3 . 6
11 5 5 5 5 . . 7 . 8 .
12 5 . 5 . 7 7 7 7 . .
13 . . . . . 7 . 7 . .
14
```

Properties.txt

```
1  CCProperties
2  10 10 0 1
3  8
4  1
5  7
6  0 0
7  4 1
8  2
9  5
10 0 3
11 3 4
12 3
13 17
14 0 4
15 6 9
16 4
17 2
18 4 2
19 4 3
20 5
21 7
22 6 0
23 8 3
24 6
25 1
26 6 9
27 6 9
28 7
29 7
30 7 4
31 9 7
32 8
33 1
34 7 8
35 7 8
```

Debug.txt

```

1  -----entering connected 4 method-----
2
3  After Connected4Pass1, newLabel = 13
4  numRows: 10 numCols: 10 minVal: 0 maxVal: 1
5  1  1  .  2  .  .  3  .  4  .
6  .  1  .  2  2  .  3  .  4  .
7  .  1  .  .  2  .  3  .  4  .
8  5  1  .  .  2  .  3  .  4  4
9  5  .  6  6  .  .  3  .  4  .
10 .  .  .  .  7  7  3  3  3  .
11 .  .  8  .  .  .  .  3  .  9
12 10 10 8  8  .  .  11 .  12 .
13 10 .  8  .  13 13 11 11 .  .
14 .  .  .  .  .  13 .  11 .  .
15
16 -----Printing EQ Table-----
17  1  2  3  4  5  6  7  8  9  10 11 12 13
18  1  2  3  3  1  6  3  8  9  8  11 12 11
19
20 After Connected4Pass2, newLabel = 13
21 numRows: 10 numCols: 10 minVal: 0 maxVal: 1
22  1  1  .  2  .  .  3  .  3  .
23  .  1  .  2  2  .  3  .  3  .
24  .  1  .  .  2  .  3  .  3  .
25  1  1  .  .  2  .  3  .  3  3
26  1  .  6  6  .  .  3  .  3  .
27  .  .  .  .  3  3  3  3  3  .
28  .  .  8  .  .  .  .  3  .  9
29  8  8  8  8  .  .  11 .  12 .
30  8  .  8  .  11 11 11 11 .  .
31  .  .  .  .  .  11 .  11 .  .
32
33 -----Printing EQ Table-----
34  1  2  3  4  5  6  7  8  9  10 11 12 13
35  1  2  3  3  1  6  3  8  9  8  11 12 11
36

```

```

37  ----Running Managing EQ ARRAY----
38  -----Printing EQ Table-----
39  1   2   3   4   5   6   7   8   9   10  11  12  13
40  1   2   3   3   1   4   3   5   6   5   7   8   7
41
42
43  In Connected4, after manage EQarr, trueNumCC = 8
44  Entering connectPass3 Method
45  ---leaving connectPass3 Method---
46
47
48  -----After Pass3-----
49  numRows: 10 numCols: 10 minVal: 0 maxVal: 1
50  1   1   .   2   .   .   3   .   3   .
51  .   1   .   2   2   .   3   .   3   .
52  .   1   .   .   2   .   3   .   3   .
53  1   1   .   .   2   .   3   .   3   3
54  1   .   4   4   .   .   3   .   3   .
55  .   .   .   .   3   3   3   3   3   .
56  .   .   5   .   .   .   .   3   .   6
57  5   5   5   5   .   .   7   .   8   .
58  5   .   5   .   7   7   7   7   .   .
59  .   .   .   .   .   7   .   7   .   .
60
61  -----Printing EQ Table-----
62  1   2   3   4   5   6   7   8   9   10  11  12  13
63  1   2   3   3   1   4   3   5   6   5   7   8   7
64
65  Leaving connected4 method
66

```

CONN 8 Data 2.txt

PrettyPrint.txt

```
1  ----Printing Original Table-----
2  After Connected4Pass1, newLabel = 9
3  numRows: 30 numCols: 35 minVal: 0 maxVal: 1
4  . . . . . 1 1 1 1 1 1 1 1 1 1 . . . . .
5  . . . . . 1 1 1 1 1 1 1 1 1 1 . . . . .
6  . . . . . 1 1 1 1 1 1 1 1 1 1 . . . . . 2 2 2 2 . . 1 . . .
7  . . . . . . . . . 1 1 . . . . . . . . 2 2 2 2 2 2 1 . . . .
8  . . . . . . . . . 1 1 . . . . . . . . 2 2 . . . . 1 . . . .
9  . . . . . . . . . 1 1 . . . . . . . . 2 2 . . 3 3 . 1 1 . . .
10 . . . . . . . . . 1 1 . . . . . . . . 2 2 . . 3 3 . 1 1 . . .
11 . . . . . 4 . . . 1 1 . . . . . . . . 2 2 . . 3 3 . 1 1 . . .
12 . . . . . 4 . . . 1 1 . . . . . . . . 2 2 . . . . 1 . . . .
13 . . . . . . 4 . . 1 1 . . . . . . . . 2 2 2 2 1 1 . . . .
14 . . . . . . . 1 1 1 1 . . . . . . . . 2 2 1 1 . . . . .
15 . . . . . . . 1 1 1 1 . . . . . . . . . . . . . . . . . .
16 . . . . . . . . . 1 . . . . . . . . . . . . . . . . . .
17 . . . . . . . . . 1 . . . . . . . 5 5 5 5 5 5 5 5 . . . . .
18 . . . . . . . . . 1 . . . . . . . 5 5 5 5 5 5 5 5 . . . . .
19 . . . . . . . . 1 . . . . . . . 5 5 5 . . . . 5 5 5 5 1 1 1 . .
20 . . . . . . . 1 . . . . . . . 5 5 . . . . . . . 5 5 1 1 1 1 . .
21 . . . . . 1 1 1 . . . . . . . 5 5 . . . . 6 6 . . 5 1 1 1 1 1 . .
22 . . . . 7 1 1 1 1 1 1 . . . . . 5 5 . . . . 6 6 . . . . 1 1 1 1 . .
23 . . . 7 1 1 1 1 1 1 . . . . . 5 5 . . . . 8 6 6 6 . . . . 1 1 1 1 . .
24 . . . 1 1 1 . . . . 1 1 1 1 . . . 5 5 . . . 6 6 6 6 6 . . . . 1 1 . .
25 . . . 1 1 . . . . . 1 1 1 1 . . . 5 5 . . . 6 6 6 6 6 . . . . 1 1 . .
26 . . . 1 1 . . . . . 1 1 1 1 . . . 5 5 . . . 6 6 6 6 6 . . . . 1 1 . .
27 . . . 1 1 1 . . . . . 1 1 1 1 . . . 5 5 . . . 6 6 6 6 6 . . . . 1 1 . .
28 . . . 1 1 1 1 1 1 . . 1 1 1 . . . 5 5 . . . . 6 6 . . . . 1 1 . .
29 . . . 1 1 1 1 1 1 1 1 1 1 1 . . . 5 5 . . . . 6 6 . . . . 9 1 1 1 . .
30 . . . 1 1 1 1 1 1 1 1 1 1 1 . . . 5 5 . . . . . . . 9 1 1 1 1 . .
31 . . . 1 1 . . . . . 1 1 1 1 . . . 5 5 5 . . . . . . . 1 1 1 1 1 . .
32 . . . 1 1 . . . . . 1 1 1 1 . . . 5 5 5 5 5 5 5 5 5 5 1 1 1 1 1 . .
33 . . . 1 1 . . . . . 1 1 1 1 . . . . 5 5 5 5 5 5 5 5 5 5 1 1 1 1 1 . .
```

```

35  -----Printing EQ Table-----
36  1  2  3  4  5  6  7  8  9
37  1  1  3  1  1  6  1  6  1
38
39  After Connected4Pass2, newLabel = 9
40  numRows: 30 numCols: 35 minVal: 0 maxVal: 1
41  . . . . . 1 1 1 1 1 1 1 1 1 . . . . .
42  . . . . . 1 1 1 1 1 1 1 1 1 . . . . .
43  . . . . . 1 1 1 1 1 1 1 1 1 . . . . .
44  . . . . . . . . . 1 1 . . . . . 1 1 1 1 1 1 . . . . .
45  . . . . . . . . . 1 1 . . . . . 1 1 . . . . . 1 . . . . .
46  . . . . . . . . . 1 1 . . . . . 1 1 . . 3 3 . 1 1 . . . . .
47  . . . . . . . . . 1 1 . . . . . 1 1 . . 3 3 . 1 1 . . . . .
48  . . . . . 1 . . . 1 1 . . . . . 1 1 . . 3 3 . 1 1 . . . . .
49  . . . . . . 1 . . 1 1 . . . . . 1 1 . . . . . 1 . . . . .
50  . . . . . . . 1 . 1 1 . . . . . . . 1 1 1 1 1 1 . . . . .
51  . . . . . . . . 1 1 1 1 . . . . . . . 1 1 1 1 . . . . .
52  . . . . . . . . 1 1 1 1 . . . . . . . . . . . . . . . . .
53  . . . . . . . . . . 1 . . . . . . . . . . . . . . . . .
54  . . . . . . . . . . 1 . . . . . . 1 1 1 1 1 1 1 1 . . . . .
55  . . . . . . . . . . 1 . . . . . . 1 1 1 1 1 1 1 1 . . . . .
56  . . . . . . . . . 1 . . . . . . 1 1 1 . . . . . 1 1 1 1 1 1 . .
57  . . . . . . . . . 1 . . . . . . 1 1 . . . . . . . 1 1 1 1 1 1 . .
58  . . . . . . . 1 1 1 . . . . . . 1 1 . . . . . 6 6 . . 1 1 1 1 1 1 . .
59  . . . . . 1 1 1 1 1 1 1 . . . . . 1 1 . . . . . 6 6 . . . 1 1 1 1 . .
60  . . . . 1 1 1 1 1 1 1 . . . . . 1 1 . . . . . 6 6 6 6 . . . 1 1 1 1 . .
61  . . . . 1 1 1 . . . . 1 1 1 1 . . . . . 1 1 . . . 6 6 6 6 6 . . . 1 1 . .
62  . . . . 1 1 . . . . . 1 1 1 1 . . . . . 1 1 . . . 6 6 6 6 6 . . . 1 1 . .
63  . . . . 1 1 . . . . . 1 1 1 1 . . . . . 1 1 . . . 6 6 6 6 6 . . . 1 1 . .
64  . . . . 1 1 1 . . . . . 1 1 1 1 . . . . . 1 1 . . . 6 6 6 6 6 . . . 1 1 . .
65  . . . . 1 1 1 1 1 1 . . 1 1 1 . . . . . 1 1 . . . 6 6 . . . . . 1 1 . .
66  . . . . 1 1 1 1 1 1 1 1 1 1 . . . . . 1 1 . . . 6 6 . . . . . 1 1 1 1 . .
67  . . . . 1 1 1 1 1 1 1 1 1 1 . . . . . 1 1 . . . . . . . 1 1 1 1 1 1 . .
68  . . . . 1 1 . . . . . 1 1 1 1 . . . . . 1 1 1 . . . . . . . 1 1 1 1 1 1 . .
69  . . . . 1 1 . . . . . 1 1 1 1 . . . . . 1 1 1 1 1 1 1 1 1 1 1 1 1 1 . . .
70  . . . . 1 1 . . . . . 1 1 1 1 . . . . . 1 1 1 1 1 1 1 1 1 1 1 1 1 1 . . .

```

```

72  -----Printing EQ Table-----
73  1  2  3  4  5  6  7  8  9
74  1  1  3  1  1  6  1  6  1
75
76  ----Running Managing EQ ARRAY----
77  -----Printing EQ Table-----
78  1  2  3  4  5  6  7  8  9
79  1  1  2  1  1  3  1  3  1
80
81
82  In Connected4, after manage EQarr, trueNumCC = 3
83

```

```

85  -----After Pass3-----
86  numRows: 30 numCols: 35 minVal: 0 maxVal: 1
87  . . . . . 1 1 1 1 1 1 1 1 1 1 . . . . .
88  . . . . . 1 1 1 1 1 1 1 1 1 1 . . . . .
89  . . . . . 1 1 1 1 1 1 1 1 1 1 . . . . .
90  . . . . . . . . 1 1 . . . . . 1 1 1 1 1 1 . . . . .
91  . . . . . . . . 1 1 . . . . . 1 1 . . . . .
92  . . . . . . . . 1 1 . . . . . 1 1 . . 2 2 . 1 1 . . . . .
93  . . . . . . . . 1 1 . . . . . 1 1 . . 2 2 . 1 1 . . . . .
94  . . . . . 1 . . . 1 1 . . . . . 1 1 . . 2 2 . 1 1 . . . . .
95  . . . . . . 1 . . 1 1 . . . . . 1 1 . . . . 1 . . . . .
96  . . . . . . . 1 . 1 1 . . . . . 1 1 1 1 1 1 . . . . .
97  . . . . . . . . 1 1 1 1 . . . . . 1 1 1 1 . . . . .
98  . . . . . . . . 1 1 1 1 . . . . . . . . . . . . . . .
99  . . . . . . . . . 1 . . . . . . . . . . . . . . .
100 . . . . . . . . . 1 . . . . . . . 1 1 1 1 1 1 1 1 . . . . .
101 . . . . . . . . . 1 . . . . . . . 1 1 1 1 1 1 1 1 . . . . .
102 . . . . . . . . . 1 . . . . . . . 1 1 1 . . . . . 1 1 1 1 1 1 . .
103 . . . . . . . . . 1 . . . . . . . 1 1 . . . . . . . 1 1 1 1 1 1 . .
104 . . . . . . . . . 1 1 1 . . . . . 1 1 . . . . . 3 3 . . 1 1 1 1 1 1 . .
105 . . . . . 1 1 1 1 1 1 1 . . . . . 1 1 . . . . . 3 3 . . . . 1 1 1 1 . .
106 . . . . 1 1 1 1 1 1 1 . . . . . 1 1 . . . . . 3 3 3 3 . . . . 1 1 1 1 . .
107 . . . . 1 1 1 . . . . . 1 1 1 1 . . . . . 1 1 . . . . . 3 3 3 3 3 . . . . 1 1 . .
108 . . . . 1 1 . . . . . . 1 1 1 1 . . . . . 1 1 . . . . . 3 3 3 3 3 . . . . 1 1 . .
109 . . . . 1 1 . . . . . . 1 1 1 1 . . . . . 1 1 . . . . . 3 3 3 3 3 . . . . 1 1 . .
110 . . . . 1 1 1 . . . . . 1 1 1 1 . . . . . 1 1 . . . . . 3 3 3 3 3 . . . . 1 1 . .
111 . . . . 1 1 1 1 1 1 1 . 1 1 1 . . . . . 1 1 . . . . . 3 3 . . . . . 1 1 . .
112 . . . . 1 1 1 1 1 1 1 1 1 1 1 . . . . . 1 1 . . . . . 3 3 . . . . . 1 1 1 1 . .
113 . . . . 1 1 1 1 1 1 1 1 1 1 1 . . . . . 1 1 . . . . . . . . . 1 1 1 1 1 . .
114 . . . . 1 1 . . . . . 1 1 1 1 . . . . . 1 1 1 . . . . . . . . . 1 1 1 1 1 . .
115 . . . . 1 1 . . . . . 1 1 1 1 . . . . . 1 1 1 1 1 1 1 1 1 1 1 1 1 1 . . . .
116 . . . . 1 1 . . . . . 1 1 1 1 . . . . . 1 1 1 1 1 1 1 1 1 1 1 1 1 1 . . . .
117
118  -----Printing EQ Table-----
119  1 2 3 4 5 6 7 8 9
120  1 1 2 1 1 3 1 3 1

```


Label.txt

```
1  numRows: 30 numCols: 35 minVal: 0 maxVal: 1
2  -----Printing image-----
3  30 35 0 1
4  . . . . . 1 1 1 1 1 1 1 1 1 1 . . . . .
5  . . . . . 1 1 1 1 1 1 1 1 1 1 . . . . .
6  . . . . . 1 1 1 1 1 1 1 1 1 1 . . . . .
7  . . . . . . . . 1 1 . . . . . 1 1 1 1 1 1 . . . . .
8  . . . . . . . . . 1 1 . . . . . 1 1 . . . . .
9  . . . . . . . . . 1 1 . . . . . 1 1 . . 2 2 . 1 1 . . . . .
10 . . . . . . . . . 1 1 . . . . . 1 1 . . 2 2 . 1 1 . . . . .
11 . . . . . 1 . . . 1 1 . . . . . 1 1 . . 2 2 . 1 1 . . . . .
12 . . . . . 1 . . 1 1 . . . . . 1 1 . . . . . 1 . . . . .
13 . . . . . . 1 . . 1 1 . . . . . 1 1 1 1 1 1 . . . . .
14 . . . . . . . 1 1 1 1 . . . . . 1 1 1 1 . . . . .
15 . . . . . . . . 1 1 1 1 . . . . . . . . . . . . . . .
16 . . . . . . . . . . 1 . . . . . . . . . . . . . . .
17 . . . . . . . . . 1 . . . . . 1 1 1 1 1 1 1 1 . . . . .
18 . . . . . . . . . 1 . . . . . 1 1 1 1 1 1 1 1 . . . . .
19 . . . . . . . . 1 . . . . . 1 1 1 . . . . 1 1 1 1 1 1 . . . . .
20 . . . . . . . . 1 . . . . . 1 1 . . . . . 1 1 1 1 1 1 . . . . .
21 . . . . . 1 1 1 . . . . . 1 1 . . . . . 3 3 . . 1 1 1 1 1 1 . . . . .
22 . . . . 1 1 1 1 1 1 1 . . . . . 1 1 . . . . . 3 3 . . . . 1 1 1 1 . . . . .
23 . . . 1 1 1 1 1 1 1 . . . . . 1 1 . . . . . 3 3 3 3 . . . . 1 1 1 1 . . . . .
24 . . . 1 1 1 . . . . 1 1 1 1 . . . . 1 1 . . . . 3 3 3 3 3 . . . . . 1 1 . . . . .
25 . . . 1 1 . . . . . 1 1 1 1 . . . . 1 1 . . . . 3 3 3 3 3 . . . . . 1 1 . . . . .
26 . . . 1 1 . . . . . 1 1 1 1 . . . . 1 1 . . . . 3 3 3 3 3 . . . . . 1 1 . . . . .
27 . . . 1 1 1 . . . . . 1 1 1 1 . . . . 1 1 . . . . 3 3 3 3 3 . . . . . 1 1 . . . . .
28 . . . 1 1 1 1 1 1 . 1 1 1 . . . . 1 1 . . . . 3 3 . . . . . 1 1 . . . . .
29 . . . 1 1 1 1 1 1 1 1 1 1 . . . . 1 1 . . . . 3 3 . . . . . 1 1 1 1 . . . . .
30 . . . 1 1 1 1 1 1 1 1 1 1 . . . . 1 1 . . . . . . . . . 1 1 1 1 1 . . . . .
31 . . . 1 1 . . . . . 1 1 1 1 . . . . 1 1 1 . . . . . . . . 1 1 1 1 1 . . . . .
32 . . . 1 1 . . . . . 1 1 1 1 . . . . 1 1 1 1 1 1 1 1 1 1 1 1 1 . . . . .
33 . . . 1 1 . . . . . 1 1 1 1 . . . . 1 1 1 1 1 1 1 1 1 1 1 1 1 . . . . .
34
```

```
1  CCProperties
2  30 35 0 1
3  3
4  1
5  319
6  0 3
7  29 32
8  2
9  6
10 5 26
11 7 27
12 3
13 32
14 17 21
15 25 25
16
```

```

1  -----entering connected 8 method-----
2  After Connected8Pass1, newLabel = 9
3  numRows: 30 numCols: 35 minVal: 0 maxVal: 1
4  . . . . . 1 1 1 1 1 1 1 1 1 1 . . . . .
5  . . . . . 1 1 1 1 1 1 1 1 1 1 . . . . .
6  . . . . . 1 1 1 1 1 1 1 1 1 1 . . . . .
7  . . . . . . . . 1 1 . . . . . 2 2 2 2 2 2 1 . . . . .
8  . . . . . . . . 1 1 . . . . . 2 2 . . . . 1 . . . . .
9  . . . . . . . . 1 1 . . . . . 2 2 . . 3 3 . 1 1 . . . . .
10 . . . . . . . . 1 1 . . . . . 2 2 . . 3 3 . 1 1 . . . . .
11 . . . . . 4 . . . 1 1 . . . . . 2 2 . . 3 3 . 1 1 . . . . .
12 . . . . . 4 . . . 1 1 . . . . . 2 2 . . . 1 . . . . .
13 . . . . . . 4 . . 1 1 . . . . . 2 2 2 2 1 1 . . . . .
14 . . . . . . . 1 1 1 1 . . . . . 2 2 1 1 . . . . .
15 . . . . . . . 1 1 1 1 . . . . . . . . . . . . . . . .
16 . . . . . . . . . 1 . . . . . . . . . . . . . . . .
17 . . . . . . . . . 1 . . . . . . 5 5 5 5 5 5 5 5 . . . . .
18 . . . . . . . . . 1 . . . . . . . 5 5 5 5 5 5 5 5 . . . . .
19 . . . . . . . . 1 . . . . . . . 5 5 5 . . . . . 5 5 5 1 1 1 . .
20 . . . . . . . . 1 . . . . . . . 5 5 . . . . . . 5 5 1 1 1 1 . .
21 . . . . . . 1 1 1 . . . . . . . 5 5 . . . . . 6 6 . . 5 1 1 1 1 1 . .
22 . . . . . 7 1 1 1 1 1 1 . . . . . 5 5 . . . . . 6 6 . . . 1 1 1 1 . .
23 . . . . . 7 1 1 1 1 1 1 . . . . . 5 5 . . . . . 8 6 6 6 . . . 1 1 1 1 . .
24 . . . . 1 1 1 . . . . 1 1 1 1 . . . 5 5 . . . 6 6 6 6 6 6 . . . 1 1 . .
25 . . . . 1 1 . . . . . 1 1 1 1 . . . 5 5 . . . 6 6 6 6 6 6 . . . 1 1 . .
26 . . . . 1 1 . . . . . 1 1 1 1 . . . 5 5 . . . 6 6 6 6 6 6 . . . 1 1 . .
27 . . . . 1 1 1 . . . . . 1 1 1 1 . . . 5 5 . . . 6 6 6 6 6 6 . . . 1 1 . .
28 . . . . 1 1 1 1 1 1 1 . 1 1 1 . . . 5 5 . . . . 6 6 . . . . 1 1 . .
29 . . . . 1 1 1 1 1 1 1 1 1 1 1 . . . 5 5 . . . . 6 6 . . . . 9 1 1 1 . .
30 . . . . 1 1 1 1 1 1 1 1 1 1 1 . . . 5 5 . . . . . . . . 9 1 1 1 1 1 . .
31 . . . . 1 1 . . . . . 1 1 1 1 . . . 5 5 5 . . . . . . . . 1 1 1 1 1 . .
32 . . . . 1 1 . . . . . 1 1 1 1 . . . 5 5 5 5 5 5 5 5 5 5 1 1 1 1 1 . .
33 . . . . 1 1 . . . . . 1 1 1 1 . . . 5 5 5 5 5 5 5 5 5 5 1 1 1 1 1 1 . .
34
35  -----Printing EQ Table-----
36  1  2  3  4  5  6  7  8  9
37  1  1  3  1  1  6  1  6  1

```

```

39 After Connected8Pass2, newLabel = 9
40 numRows: 30 numCols: 35 minVal: 0 maxVal: 1
41 . . . . . 1 1 1 1 1 1 1 1 1 1 1 . . . . .
42 . . . . . 1 1 1 1 1 1 1 1 1 1 1 . . . . .
43 . . . . . 1 1 1 1 1 1 1 1 1 1 1 . . . . .
44 . . . . . . . . . 1 1 . . . . . . . . . 1 1 1 1 1 1 . . . . .
45 . . . . . . . . . 1 1 . . . . . . . . . 1 1 . . . 1 . . . . .
46 . . . . . . . . . 1 1 . . . . . . . . . 1 1 . . 3 3 . 1 1 . . . . .
47 . . . . . . . . . 1 1 . . . . . . . . . 1 1 . . 3 3 . 1 1 . . . . .
48 . . . . . 1 . . . . 1 1 . . . . . . . . . 1 1 . . 3 3 . 1 1 . . . . .
49 . . . . . 1 . . . . 1 1 . . . . . . . . . 1 1 . . . . 1 . . . . .
50 . . . . . 1 . . . . 1 1 . . . . . . . . . 1 1 1 1 1 1 . . . . .
51 . . . . . . . . . 1 1 1 1 . . . . . . . . . 1 1 1 1 . . . . .
52 . . . . . . . . . 1 1 1 1 . . . . . . . . . . . . . . . . . . . . .
53 . . . . . . . . . . . 1 . . . . . . . . . . . . . . . . . . . . .
54 . . . . . . . . . . 1 . . . . . . . . . 1 1 1 1 1 1 1 1 1 . . . . .
55 . . . . . . . . . . 1 . . . . . . . . . 1 1 1 1 1 1 1 1 1 . . . . .
56 . . . . . . . . . 1 . . . . . . . . . 1 1 1 . . . . . 1 1 1 1 1 1 . .
57 . . . . . . . . . 1 . . . . . . . . . 1 1 . . . . . . . 1 1 1 1 1 1 . .
58 . . . . . . . . . 1 1 1 . . . . . . . . . 1 1 . . . . 6 6 . . 1 1 1 1 1 1 . .
59 . . . . . 1 1 1 1 1 1 1 1 . . . . . . . . . 1 1 . . . . 6 6 . . 1 1 1 1 . .
60 . . . . 1 1 1 1 1 1 1 1 . . . . . . . . . 1 1 . . . . 6 6 6 6 . . . . 1 1 1 1 . .
61 . . . . 1 1 1 . . . . . 1 1 1 1 . . . . . . . . . 6 6 6 6 6 6 . . . . 1 1 . .
62 . . . . 1 1 . . . . . 1 1 1 1 . . . . . . . . . 6 6 6 6 6 6 . . . . 1 1 . .
63 . . . . 1 1 . . . . . 1 1 1 1 . . . . . . . . . 6 6 6 6 6 6 . . . . 1 1 . .
64 . . . . 1 1 1 . . . . . 1 1 1 1 . . . . . . . . . 6 6 6 6 6 6 . . . . 1 1 . .
65 . . . . 1 1 1 1 1 1 1 1 . . . . . . . . . 1 1 . . . . 6 6 . . . . 1 1 . .
66 . . . . 1 1 1 1 1 1 1 1 1 1 1 1 . . . . . . . . . 6 6 . . . . 1 1 1 1 . .
67 . . . . 1 1 1 1 1 1 1 1 1 1 1 1 . . . . . . . . . . . . 1 1 1 1 1 1 . .
68 . . . . 1 1 . . . . . 1 1 1 1 . . . . . . . . . . . . . . 1 1 1 1 1 1 . .
69 . . . . 1 1 . . . . . 1 1 1 1 . . . . . 1 1 1 1 1 1 1 1 1 1 1 1 1 1 . .
70 . . . . 1 1 . . . . . 1 1 1 1 . . . . . 1 1 1 1 1 1 1 1 1 1 1 1 1 1 . .
71
72 -----Printing EQ Table-----
73 1 2 3 4 5 6 7 8 9
74 1 1 3 1 1 6 1 6 1
75

```

```

76 ----Running Managing EQ ARRAY----
77 -----Printing EQ Table-----
78 1 2 3 4 5 6 7 8 9
79 1 1 2 1 1 3 1 3 1
80
81 In Connected8, after manageEQarr, trueNumCC = 3
82 Entering connectPass3 Method
83 ---leaving connectPass3 Method---
84

```

CONN 4 Data2.txt CONVERSION

PrettyPrint.txt

```
1  ----Printing Original Table-----
2  numRows: 30 numCols: 35 minVal: 0 maxVal: 1
3  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
4  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  1 1 1 1 1 . . . . . 1 1 1 1 1 1 1 1 1 1 1 . . . . 1 1 . 1 1 1 1
6  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
7  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
8  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
9  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
10 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
11 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
12 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
13 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
14 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
15 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
16 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
17 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
18 1 1 1 1 1 1 1 1 . 1 1 1 1 1 1 1 1 . . . 1 1 1 1 1 1 . . . . . 1 1
19 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
20 1 1 1 1 1 1 . . . 1 1 1 1 1 1 1 . . 1 1 1 1 1 . . 1 1 . . . . . 1 1
21 1 1 1 1 . . . . . 1 1 1 1 1 . . 1 1 1 1 1 . . 1 1 1 1 . . . . . 1 1
22 1 1 1 . . . . . 1 1 1 1 1 . . 1 1 1 . . . . 1 1 1 1 . . . . . 1 1
23 1 1 1 . . . 1 1 1 . . . . 1 1 1 . . 1 1 1 . . . . 1 1 1 1 1 . . 1 1
24 1 1 1 . . 1 1 1 1 1 . . . . 1 1 . . 1 1 1 . . . . 1 1 1 1 1 . . 1 1
25 1 1 1 . . 1 1 1 1 1 . . . . 1 1 . . 1 1 1 . . . . 1 1 1 1 1 . . 1 1
26 1 1 1 . . . 1 1 1 1 . . . . 1 1 . . 1 1 1 . . . . 1 1 1 1 1 . . 1 1
27 1 1 1 . . . . . 1 . . . 1 1 1 . . 1 1 1 1 1 . . 1 1 1 1 1 1 . . 1 1
28 1 1 1 . . . . . . . . . 1 1 1 . . 1 1 1 1 1 . . 1 1 1 1 . . . . 1 1
29 1 1 1 . . . . . . . . . 1 1 1 . . 1 1 1 1 1 1 1 1 1 1 . . . . . 1 1
30 1 1 1 . . 1 1 1 1 . . . . 1 1 1 . . . 1 1 1 1 1 1 1 1 1 . . . . . 1 1
31 1 1 1 . . 1 1 1 1 . . . . 1 1 1 . . . . . . . . . . . . . . 1 1 1
32 1 1 1 . . 1 1 1 1 . . . . 1 1 1 1 . . . . . . . . . . . . . . 1 1 1
```

```
34 After Connected4Pass1, newLabel = 22
35 numRows: 30 numCols: 35 minVal: 0 maxVal: 1
36 1 1 1 1 1 . . . . . . . . . . 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
37 1 1 1 1 1 . . . . . . . . . . 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
38 1 1 1 1 1 . . . . . . . . . . 2 2 2 2 2 2 2 2 2 2 2 2 . . . . 2 2 . 2 2 2 2
39 1 1 1 1 1 1 1 1 1 . . 3 3 3 3 2 2 2 2 2 2 2 2 2 . . . . . . 4 2 2 2 2
40 1 1 1 1 1 1 1 1 1 . . 3 3 3 3 2 2 2 2 2 2 2 2 2 . . 5 5 5 5 . 6 4 2 2 2 2
41 1 1 1 1 1 1 1 1 1 . . 3 3 3 3 2 2 2 2 2 2 2 2 . . 7 5 . . 5 . . 4 2 2 2 2
42 1 1 1 1 1 1 1 1 1 . . 3 3 3 3 2 2 2 2 2 2 2 2 . . 7 5 . . 5 . . 4 2 2 2 2
43 1 1 1 1 1 . 1 1 1 . . 3 3 3 3 2 2 2 2 2 2 2 . . 7 5 . . 5 . . 4 2 2 2 2
44 1 1 1 1 1 1 . 1 1 . . 3 3 3 3 2 2 2 2 2 2 2 2 . . 5 5 5 5 . 8 4 2 2 2 2
45 1 1 1 1 1 1 1 . 1 . . 3 3 3 3 2 2 2 2 2 2 2 2 . . . . . . 8 4 2 2 2 2
46 1 1 1 1 1 1 1 1 . . . . 3 3 3 2 2 2 2 2 2 2 2 2 . . . . . 9 8 4 2 2 2 2
47 1 1 1 1 1 1 1 1 . . . . 3 3 3 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
48 1 1 1 1 1 1 1 1 1 . 3 3 3 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
49 1 1 1 1 1 1 1 1 1 1 . 10 3 3 3 2 2 2 . . . . . . . . . 2 2 2 2 2 2 2 2
50 1 1 1 1 1 1 1 1 1 . 11 10 3 3 3 2 2 . . . . . . . . . 2 2 2 2 2 2 2 2
51 1 1 1 1 1 1 1 1 . 12 11 10 3 3 3 2 . . . 13 13 13 13 13 13 . . . . . 2 2
52 1 1 1 1 1 1 1 . 14 12 11 10 3 3 3 2 . . 15 15 13 13 13 13 13 13 . . . . . 2 2
53 1 1 1 1 1 1 . . . 12 11 10 3 3 3 2 . . 15 15 13 13 13 . . 13 13 . . . . . 2 2
54 1 1 1 1 . . . . . . . 10 3 3 3 2 . . 15 15 13 13 13 . . 13 13 13 13 . . . 2 2
55 1 1 1 . . . . . . . 16 10 3 3 3 2 . . 15 15 13 . . . . 13 13 13 13 . . . 2 2
56 1 1 1 . . . 17 17 17 . . . . 3 3 2 . . 15 15 13 . . . . . 13 13 13 13 13 . . 2 2
57 1 1 1 . . 18 17 17 17 17 . . . . 3 2 . . 15 15 13 . . . . . 13 13 13 13 13 . . 2 2
58 1 1 1 . . 18 17 17 17 17 . . . . 3 2 . . 15 15 13 . . . . . 13 13 13 13 13 . . 2 2
59 1 1 1 . . . 17 17 17 17 . . . . 3 2 . . 15 15 13 . . . . . 13 13 13 13 13 . . 2 2
60 1 1 1 . . . . . . 17 . . . 19 3 2 . . 15 15 13 13 13 . . 20 13 13 13 13 13 . . 2 2
61 1 1 1 . . . . . . . . . 19 3 2 . . 15 15 13 13 13 . . 20 13 13 13 . . . . 2 2
62 1 1 1 . . . . . . . . . 19 3 2 . . 15 15 13 13 13 13 13 13 13 13 . . . . 2 2
63 1 1 1 . . 21 21 21 21 . . . . 19 3 2 . . 15 13 13 13 13 13 13 13 13 . . . . 2 2
64 1 1 1 . . 21 21 21 21 . . . . 19 3 2 . . . . . . . . . . . . . . . 22 2 2
65 1 1 1 . . 21 21 21 21 . . . . 19 3 2 2 . . . . . . . . . . . . . . . 22 2 2
```

```

67  -----Printing EQ Table-----
68  1  2  3  4  5  6  7  8  9  10 11 12 13 14 15 16 17 18 19 20 21 22
69  1  2  2  2  5  4  5  2  2  3 10 11 13 12 13 10 17 17 3 13 21 2
70
71  After Connected4Pass2, newLabel = 22
72  numRows: 30 numCols: 35 minVal: 0 maxVal: 1
73  1 1 1 1 1 . . . . . 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
74  1 1 1 1 1 . . . . . 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
75  1 1 1 1 1 . . . . . 2 2 2 2 2 2 2 2 2 2 2 2 . . . . 2 2 . 2 2 2 2
76  1 1 1 1 1 1 1 1 1 . . 2 2 2 2 2 2 2 2 2 2 2 2 . . . . . 2 2 2 2 2
77  1 1 1 1 1 1 1 1 1 . . 2 2 2 2 2 2 2 2 2 2 2 2 . . 5 5 5 5 . 4 2 2 2 2
78  1 1 1 1 1 1 1 1 1 . . 2 2 2 2 2 2 2 2 2 2 2 2 . . 5 5 . . 5 . . 2 2 2 2
79  1 1 1 1 1 1 1 1 1 . . 2 2 2 2 2 2 2 2 2 2 2 2 . . 5 5 . . 5 . . 2 2 2 2
80  1 1 1 1 1 . 1 1 1 . . 2 2 2 2 2 2 2 2 2 2 2 2 . . 5 5 . . 5 . . 2 2 2 2
81  1 1 1 1 1 1 . 1 1 . . 2 2 2 2 2 2 2 2 2 2 2 2 . . 5 5 5 5 . 2 2 2 2 2
82  1 1 1 1 1 1 1 . 1 . . 2 2 2 2 2 2 2 2 2 2 2 2 . . . . . 2 2 2 2 2
83  1 1 1 1 1 1 1 1 . . . . 2 2 2 2 2 2 2 2 2 2 2 2 . . . . 2 2 2 2 2 2
84  1 1 1 1 1 1 1 1 . . . . 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
85  1 1 1 1 1 1 1 1 1 1 . 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
86  1 1 1 1 1 1 1 1 1 1 . 2 2 2 2 2 2 2 2 . . . . . 2 2 2 2 2 2 2 2
87  1 1 1 1 1 1 1 1 1 . 2 2 2 2 2 2 2 2 . . . . . 2 2 2 2 2 2 2 2
88  1 1 1 1 1 1 1 1 . 2 2 2 2 2 2 2 2 . . . 13 13 13 13 13 13 . . . . . 2 2
89  1 1 1 1 1 1 1 . 12 2 2 2 2 2 2 2 . . 13 13 13 13 13 13 13 13 13 . . . . 2 2
90  1 1 1 1 1 1 . . . 11 10 2 2 2 2 2 . . 13 13 13 13 13 . . 13 13 . . . . 2 2
91  1 1 1 1 . . . . . 2 2 2 2 2 . . 13 13 13 13 13 . . 13 13 13 13 . . . . 2 2
92  1 1 1 . . . . . 10 3 2 2 2 2 . . 13 13 13 . . . . 13 13 13 13 . . . . 2 2
93  1 1 1 . . . 17 17 17 . . . 2 2 2 . . 13 13 13 . . . . 13 13 13 13 . . 2 2
94  1 1 1 . . 17 17 17 17 . . . 2 2 . . 13 13 13 . . . . 13 13 13 13 . . 2 2
95  1 1 1 . . 17 17 17 17 17 . . . 2 2 . . 13 13 13 . . . . 13 13 13 13 . . 2 2
96  1 1 1 . . 17 17 17 17 . . . 2 2 . . 13 13 13 . . . . 13 13 13 13 . . 2 2
97  1 1 1 . . . . . 17 . . . 2 2 2 . . 13 13 13 13 13 . . 13 13 13 13 13 . . 2 2
98  1 1 1 . . . . . . . . 2 2 2 . . 13 13 13 13 13 . . 13 13 13 13 . . . . 2 2
99  1 1 1 . . . . . . . . 2 2 2 . . 13 13 13 13 13 13 13 13 13 13 . . . . 2 2
100 1 1 1 . . 21 21 21 21 . . . 2 2 2 . . 13 13 13 13 13 13 13 13 13 . . . . 2 2
101 1 1 1 . . 21 21 21 21 . . . 2 2 2 . . . . . . . . . . . . . . . 2 2 2
102 1 1 1 . . 21 21 21 21 . . . 3 2 2 2 . . . . . . . . . . . . . . . 2 2 2
103
104  -----Printing EQ Table-----
105  1  2  3  4  5  6  7  8  9  10 11 12 13 14 15 16 17 18 19 20 21 22
106  1  2  2  2  5  4  5  2  2  2  2  2 13 12 13 10 17 17 2 13 21 2
107
108  ----Running Managing EQ ARRAY----
109  -----Printing EQ Table-----
110  1  2  3  4  5  6  7  8  9  10 11 12 13 14 15 16 17 18 19 20 21 22
111  1  2  2  2  3  2  3  2  2  2  2  2  4  2  4  2  5  5  2  4  6  2
112
113
114  In Connected4, after manage EQarr, trueNumCC = 6

```



```
117 -----After Pass3-----
118 numRows: 30 numCols: 35 minVal: 0 maxVal: 1
119 1 1 1 1 1 . . . . . 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
120 1 1 1 1 1 . . . . . 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
121 1 1 1 1 1 . . . . . 2 2 2 2 2 2 2 2 2 2 2 2 . . . . 2 2 . 2 2 2 2
122 1 1 1 1 1 1 1 1 1 . . 2 2 2 2 2 2 2 2 2 2 2 2 . . . . . 2 2 2 2 2 2
123 1 1 1 1 1 1 1 1 1 . . 2 2 2 2 2 2 2 2 2 2 2 2 . . 3 3 3 3 . 2 2 2 2 2 2
124 1 1 1 1 1 1 1 1 1 . . 2 2 2 2 2 2 2 2 2 2 2 2 . . 3 3 . . 3 . . 2 2 2 2 2 2
125 1 1 1 1 1 1 1 1 1 . . 2 2 2 2 2 2 2 2 2 2 2 2 . . 3 3 . . 3 . . 2 2 2 2 2 2
126 1 1 1 1 1 . 1 1 1 . . 2 2 2 2 2 2 2 2 2 2 2 2 . . 3 3 . . 3 . . 2 2 2 2 2 2
127 1 1 1 1 1 1 . 1 1 . . 2 2 2 2 2 2 2 2 2 2 2 2 . . 3 3 3 3 . 2 2 2 2 2 2
128 1 1 1 1 1 1 1 . 1 . . 2 2 2 2 2 2 2 2 2 2 2 2 . . . . . 2 2 2 2 2 2
129 1 1 1 1 1 1 1 1 . . . . 2 2 2 2 2 2 2 2 2 2 2 2 2 2 . . . . 2 2 2 2 2 2
130 1 1 1 1 1 1 1 1 . . . . 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
131 1 1 1 1 1 1 1 1 1 1 1 . 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
132 1 1 1 1 1 1 1 1 1 1 1 . 2 2 2 2 2 2 2 2 2 2 . . . . . 2 2 2 2 2 2 2 2 2 2
133 1 1 1 1 1 1 1 1 1 . 2 2 2 2 2 2 2 2 2 . . . . . 2 2 2 2 2 2 2 2 2 2
134 1 1 1 1 1 1 1 1 . 2 2 2 2 2 2 2 2 . . . 4 4 4 4 4 4 . . . . . 2 2
135 1 1 1 1 1 1 1 . 2 2 2 2 2 2 2 2 . . 4 4 4 4 4 4 4 4 4 4 . . . . . 2 2
136 1 1 1 1 1 1 . . . 2 2 2 2 2 2 2 . . 4 4 4 4 4 . . 4 4 . . . . . 2 2
137 1 1 1 1 . . . . . 2 2 2 2 2 . . 4 4 4 4 4 . . 4 4 4 4 . . . . . 2 2
138 1 1 1 . . . . . 2 2 2 2 2 . . 4 4 4 . . . . 4 4 4 4 . . . . . 2 2
139 1 1 1 . . 5 5 5 . . . 2 2 2 . . 4 4 4 . . . . 4 4 4 4 4 . . 2 2
140 1 1 1 . . 5 5 5 5 5 . . . 2 2 . . 4 4 4 . . . . 4 4 4 4 4 . . 2 2
141 1 1 1 . . 5 5 5 5 5 . . . 2 2 . . 4 4 4 . . . . 4 4 4 4 4 . . 2 2
142 1 1 1 . . . 5 5 5 5 . . . 2 2 . . 4 4 4 . . . . 4 4 4 4 4 . . 2 2
143 1 1 1 . . . . . 5 . . . 2 2 2 . . 4 4 4 4 4 . . 4 4 4 4 4 4 . . 2 2
144 1 1 1 . . . . . . . . 2 2 2 . . 4 4 4 4 4 . . 4 4 4 4 . . . . . 2 2
145 1 1 1 . . . . . . . . 2 2 2 . . 4 4 4 4 4 4 4 4 4 4 . . . . . 2 2
146 1 1 1 . . 6 6 6 6 . . . 2 2 2 . . 4 4 4 4 4 4 4 4 4 4 . . . . . 2 2
147 1 1 1 . . 6 6 6 6 . . . 2 2 2 . . . . . . . . . . . . . . . 2 2 2
148 1 1 1 . . 6 6 6 6 . . . 2 2 2 2 . . . . . . . . . . . . . . . 2 2 2
149
150 -----Printing EQ Table-----
151 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22
152 1 2 2 2 3 2 3 2 2 2 2 2 4 2 4 2 5 5 2 4 6 2
```

Label.txt

```

1  numRows: 30 numCols: 35 minVal: 0 maxVal: 1
2  -----Printing image-----
3  30 35 0 1
4  1 1 1 1 1 1 . . . . . 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
5  1 1 1 1 1 1 . . . . . 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
6  1 1 1 1 1 1 . . . . . 2 2 2 2 2 2 2 2 2 2 2 2 . . . . 2 2 . 2 2 2 2
7  1 1 1 1 1 1 1 1 1 1 . . 2 2 2 2 2 2 2 2 2 2 2 2 2 2 . . . . . 2 2 2 2
8  1 1 1 1 1 1 1 1 1 1 . . 2 2 2 2 2 2 2 2 2 2 2 2 2 2 . . 3 3 3 3 . 2 2 2 2 2
9  1 1 1 1 1 1 1 1 1 1 . . 2 2 2 2 2 2 2 2 2 2 2 2 2 . . 3 3 . . 3 . . 2 2 2 2
10 1 1 1 1 1 1 1 1 1 1 . . 2 2 2 2 2 2 2 2 2 2 2 2 2 . . 3 3 . . 3 . . 2 2 2 2
11 1 1 1 1 1 1 . 1 1 1 . . 2 2 2 2 2 2 2 2 2 2 2 2 . . 3 3 . . 3 . . 2 2 2 2
12 1 1 1 1 1 1 . 1 1 . . 2 2 2 2 2 2 2 2 2 2 2 2 2 . . 3 3 3 3 . 2 2 2 2 2
13 1 1 1 1 1 1 1 1 . 1 . . 2 2 2 2 2 2 2 2 2 2 2 2 2 . . . . . 2 2 2 2 2
14 1 1 1 1 1 1 1 1 1 . . . . 2 2 2 2 2 2 2 2 2 2 2 2 2 . . . . 2 2 2 2 2
15 1 1 1 1 1 1 1 1 1 . . . . 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
16 1 1 1 1 1 1 1 1 1 1 1 . 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
17 1 1 1 1 1 1 1 1 1 1 1 1 . 2 2 2 2 2 2 2 2 2 . . . . . 2 2 2 2 2 2 2
18 1 1 1 1 1 1 1 1 1 1 . 2 2 2 2 2 2 2 2 2 . . . . . 2 2 2 2 2 2 2
19 1 1 1 1 1 1 1 1 1 . 2 2 2 2 2 2 2 2 2 . . . 4 4 4 4 4 4 . . . . . 2 2
20 1 1 1 1 1 1 1 . 2 2 2 2 2 2 2 2 2 . . 4 4 4 4 4 4 4 4 . . . . . 2 2
21 1 1 1 1 1 1 . . . 2 2 2 2 2 2 2 . . 4 4 4 4 4 . . 4 4 . . . . . 2 2
22 1 1 1 1 . . . . . 2 2 2 2 2 . . 4 4 4 4 4 . . 4 4 4 4 . . . . . 2 2
23 1 1 1 . . . . . 2 2 2 2 2 . . 4 4 4 . . . . 4 4 4 4 . . . . . 2 2
24 1 1 1 . . . 5 5 5 . . . 2 2 2 . . 4 4 4 . . . . 4 4 4 4 4 . . 2 2
25 1 1 1 . . 5 5 5 5 5 . . . 2 2 . . 4 4 4 . . . . 4 4 4 4 4 . . 2 2
26 1 1 1 . . 5 5 5 5 5 . . . 2 2 . . 4 4 4 . . . . 4 4 4 4 4 . . 2 2
27 1 1 1 . . . 5 5 5 5 . . . 2 2 . . 4 4 4 . . . . 4 4 4 4 4 . . 2 2
28 1 1 1 . . . . . 5 . . . 2 2 2 . . 4 4 4 4 4 . . 4 4 4 4 4 . . 2 2
29 1 1 1 . . . . . . . . 2 2 2 . . 4 4 4 4 4 . . 4 4 4 4 . . . . . 2 2
30 1 1 1 . . . . . . . . 2 2 2 . . 4 4 4 4 4 4 4 4 4 4 . . . . . 2 2
31 1 1 1 . . 6 6 6 6 . . . . 2 2 2 . . 4 4 4 4 4 4 4 4 . . . . . 2 2
32 1 1 1 . . 6 6 6 6 . . . . 2 2 2 . . . . . . . . . . . . . 2 2 2
33 1 1 1 . . 6 6 6 6 . . . . 2 2 2 2 . . . . . . . . . . . . . 2 2 2

```

Property.txt

```
1  CCProperties
2  30 35 0 1
3  6
4  1
5  179
6  0 0
7  29 10
8  2
9  358
10 0 8|
11 29 34
12 3
13 17
14 4 24
15 8 28
16 4
17 109
18 15 18
19 27 30
20 5
21 18
22 20 5
23 24 9
24 6
25 12
26 27 5
27 29 8
28
```

```

1  -----entering connected 4 method-----
2
3  After Connected4Pass1, newLabel = 22
4  numRows: 30 numCols: 35 minVal: 0 maxVal: 1
5  1 1 1 1 1 1 . . . . . 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
6  1 1 1 1 1 . . . . . 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
7  1 1 1 1 1 . . . . . 2 2 2 2 2 2 2 2 2 2 2 . . . . 2 2 . 2 2 2 2
8  1 1 1 1 1 1 1 1 1 . . 3 3 3 3 2 2 2 2 2 2 2 2 2 . . . . . 4 2 2 2
9  1 1 1 1 1 1 1 1 1 . . 3 3 3 3 2 2 2 2 2 2 2 2 . . 5 5 5 5 . 6 4 2 2 2
10 1 1 1 1 1 1 1 1 1 . . 3 3 3 3 2 2 2 2 2 2 2 . . 7 5 . . 5 . . 4 2 2 2
11 1 1 1 1 1 1 1 1 1 . . 3 3 3 3 2 2 2 2 2 2 2 . . 7 5 . . 5 . . 4 2 2 2
12 1 1 1 1 1 . 1 1 1 . . 3 3 3 3 2 2 2 2 2 2 2 . . 7 5 . . 5 . . 4 2 2 2
13 1 1 1 1 1 1 . 1 1 . . 3 3 3 3 2 2 2 2 2 2 2 . . 5 5 5 5 . 8 4 2 2 2
14 1 1 1 1 1 1 1 . 1 . . 3 3 3 3 2 2 2 2 2 2 2 . . . . . 8 4 2 2 2
15 1 1 1 1 1 1 1 1 . . . . 3 3 3 3 2 2 2 2 2 2 2 2 2 . . . 9 8 4 2 2 2
16 1 1 1 1 1 1 1 1 1 . . . . 3 3 3 3 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
17 1 1 1 1 1 1 1 1 1 1 1 . 3 3 3 3 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
18 1 1 1 1 1 1 1 1 1 1 . 10 3 3 3 2 2 2 2 . . . . . 2 2 2 2 2 2 2 2
19 1 1 1 1 1 1 1 1 1 . 11 10 3 3 3 2 2 2 . . . . . 2 2 2 2 2 2 2 2
20 1 1 1 1 1 1 1 1 1 . 12 11 10 3 3 3 2 . . . 13 13 13 13 13 13 . . . . 2 2
21 1 1 1 1 1 1 1 1 . 14 12 11 10 3 3 3 2 . . 15 15 13 13 13 13 13 13 . . . . 2 2
22 1 1 1 1 1 1 . . . 12 11 10 3 3 3 2 . . 15 15 13 13 13 . . 13 13 . . . . 2 2
23 1 1 1 1 . . . . . 10 3 3 3 2 . . 15 15 13 13 13 . . 13 13 13 13 . . . . 2 2
24 1 1 1 . . . . . . 16 10 3 3 3 2 . . 15 15 13 . . . . 13 13 13 13 . . . . 2 2
25 1 1 1 . . . 17 17 17 . . . 3 3 2 . . 15 15 13 . . . . 13 13 13 13 . . 2 2
26 1 1 1 . . 18 17 17 17 17 . . . 3 2 . . 15 15 13 . . . . 13 13 13 13 . . 2 2
27 1 1 1 . . 18 17 17 17 17 . . . 3 2 . . 15 15 13 . . . . 13 13 13 13 . . 2 2
28 1 1 1 . . 17 17 17 17 . . . 3 2 . . 15 15 13 . . . . 13 13 13 13 . . 2 2
29 1 1 1 . . . . . 17 . . . 19 3 2 . . 15 15 13 13 13 . . 20 13 13 13 13 . . 2 2
30 1 1 1 . . . . . . . 19 3 2 . . 15 15 13 13 13 . . 20 13 13 13 . . . . 2 2
31 1 1 1 . . . . . . . 19 3 2 . . 15 15 13 13 13 13 13 13 13 13 . . . . 2 2
32 1 1 1 . . 21 21 21 21 . . . 19 3 2 . . 15 13 13 13 13 13 13 13 13 . . . . 2 2
33 1 1 1 . . 21 21 21 21 . . . 19 3 2 . . . . . . . . . . . . . . 22 2 2
34 1 1 1 . . 21 21 21 21 . . . 19 3 2 2 . . . . . . . . . . . . . . 22 2 2
35
36 -----Printing EQ Table-----
37 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22
38 1 2 2 2 5 4 5 2 2 3 10 11 13 12 13 10 17 17 3 13 21 2
39

```

```
#include <iostream>
#include <fstream>
#include <sstream>
#include <vector>
#include <cstring>
#include <algorithm>

using namespace std;

struct Property {
```

```

    int label;
    int numPixels;
    int minR;
    int minC;
    int maxR;
    int maxC;

    Property() {
        label = 0;
        numPixels = 0;
        minR = 0;
        minC = 0;
        maxR = 0;
        maxC = 0;
    }
    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 ccLabel {
public:
    int numRows;
    int numCols;
    int minVal;
    int maxVal;
    int newLabel;
    int trueNumCC;
    int newMin;
    int newMax;
    int** zeroFramedAry;
    int NonZeroNeighborAry[5];

```

```

int* EQAry;
char option;
Property* CCproperty;

ccLabel(int rows, int cols, int min, int max, char conver) {
    numRows = rows;
    numCols = cols;
    minVal = min;
    maxVal = max;
    newLabel = 0;
    newMin = 0;
    newMax = 0;
    option = conver;

    zeroFramedAry = new int* [numRows + 2];
    for (int i = 0; i < numRows + 2; i++) {
        zeroFramedAry[i] = new int[numCols + 2];
        for (int j = 0; j < numCols + 2; j++) {
            zeroFramedAry[i][j] = 0;
        }
    }

    EQAry = new int[(int)(numRows * numCols) / 4];
    for (int i = 0; i < (numRows * numCols) / 4; i++) {
        EQAry[i] = i;
        // cout<<EQAry[i];
    }

    //Not Sure what to do here come back to it silly goose

}

void zero2D(int** arr, int r, int c){
    for (int i = 0; i < r; i++) {
        for (int j = 0; j < c; j++) {

```



```

        arr[i][j] = 0;
    }
}

}

void negative1D(int *arr, int size){
    for(int i = 0; i < size; i++){
        arr[i] = -1;
    }
}

void loadImage(ifstream *infile, ofstream *outfile){

    string line;
    int val;
    int r=0;

    while (getline(*infile, line)){
        int c=1;
        stringstream set(line);
        while (set >> val){
            // cout << val << " ";
            zeroFramedAry[r][c] = val;
            c++;
        }
        // cout << endl;
        r++;
        // // cout <<endl;
    }
}

void connectPass3(ofstream *out){
    *out << "Entering connectPass3 Method"<<endl;

    //step1

    for (int i = 1; i <= trueNumCC; i++)

```

```

{
    CCproperty[i].label = i;
    CCproperty[i].numPixels = 0;
    CCproperty[i].minR = numRows;
    CCproperty[i].maxR = 0;
    CCproperty[i].minC = numCols;
    CCproperty[i].maxC = 0;

}

int r=1;

//step 4
while(r<numRows+1){
    int c=1;
    while (c<numCols+1){
        int pix = zeroFramedAry[r][c];

        //step 3
        if(pix > 0){
            zeroFramedAry[r][c] = EQAry[pix];
            int k = zeroFramedAry[r][c];
            CCproperty[k].numPixels++;

            if(r<CCproperty[k].minR)
                CCproperty[k].minR = r;

            if(r>CCproperty[k].maxR)
                CCproperty[k].maxR = r;

            if(c<CCproperty[k].minC)
                CCproperty[k].minC = c;

            if(c>CCproperty[k].maxC)
                CCproperty[k].maxC = c;

        }
    }
}

```

```

        c++;
    }

    r++;
}

*out << "---leaving connectPass3 Method---" << endl << endl;

}

void conversion(){

    for (int i = 1; i < numRows+1; i++){
        for (int j = 1; j < numCols+1; j++){
            zeroFramedAry[i][j] = (zeroFramedAry[i][j]+1)%2;
        }
    }

}

void printProperty(ofstream *property){

    //Image heade
    *property << numRows << " " << numCols << " " << minVal << " " << maxVal << endl;
    //number of CCs
    // *property << "trueNumCC: ";
    *property << trueNumCC << endl;

    for (int i = 1; i <= trueNumCC; i++){

        // *property << "Label: ";
        *property << CCproperty[i].label << endl;

        // *property << "numPix: ";
        *property << CCproperty[i].numPixels << endl;
    }
}

```

```

// *property<<"minRow and MinCol ";
*property<<CCproperty[i].minR-1<<" "<<CCproperty[i].minC-1<<endl;

// *property<<"maxRow and maxCol ";
*property<<CCproperty[i].maxR-1<<" "<<CCproperty[i].maxC-1<<endl;
}

}

void imgReformat(ofstream *output){

    *output<<"numRows: " << numRows <<" numCols: " << numCols <<" minVal: " << minVal <<" maxVal: " <<
maxVal<<endl;

    int r =1;

    //step 12
        //step 4

    while (r<numRows+1){
        //step 10
        int c =1;
        while (c<numCols+1){

            //step 5
            if(zeroFramedAry[r][c]==0)
                *output<<".";
            else{
                *output<< zeroFramedAry[r][c];
            }

            //step 8
            if(zeroFramedAry[r][c] < 10){
                *output<<" ";
            }
            else{

```

```

        *output<< " ";

    }

    //step 9
    c++;
}
*output<< endl;
//step 11
r++;
}

}

void printArr(ofstream *out){

    *out<<"____PRINTING ZERO FRAMED ARR____"<<endl;

    for (int i = 0; i < numRows+2; i++){
        for (int j = 0; j < numCols+2; j++){
            *out<<zeroFramedAry[i][j]<< " ";
        }
        *out<<endl;
    }
    *out<<endl<<endl<<endl;
}

void drawBoxes(){
    //step 1
    int index = 1;

    // cout<<"trueNumCC is " << trueNumCC<<endl;
    while (index <=trueNumCC){
        // cout<<"index is " << index<<endl;
        //step 2
        int minRow = CCproperty[index].minR,
            minCol = CCproperty[index].minC,
            maxRow = CCproperty[index].maxR,

```

```

        maxCol = CCproperty[index].maxC,
        label = CCproperty[index].label;

// cout<<"\tCCproperty["<<index<<"].minR+1: "<< CCproperty[index].minR <<endl;
// cout<<"\tCCproperty["<<index<<"].minC+1: "<< CCproperty[index].minC <<endl;
// cout<<"\tCCproperty["<<index<<"].maxR+1: "<< CCproperty[index].maxR <<endl;
// cout<<"\tCCproperty["<<index<<"].maxC+1 "<< CCproperty[index].maxC <<endl;
// cout<<"\tCCproperty[numPixels] "<< CCproperty[index].numPixels <<endl;
// cout<<"\tCCproperty["<<index<<"].label "<< CCproperty[index].label<<endl;


//step 3
for (int r= minRow; r < maxRow+1; r++){
    zeroFramedAry[r][minCol] = label;
    zeroFramedAry[r][maxCol] = label;
}

for (int c = minCol; c < maxCol+1; c++){

    zeroFramedAry[minRow][c] = label;
    zeroFramedAry[maxRow][c] = label;
}


//step 4
index++;
}
}

void printImg(ofstream *out){

*out<<"-----Printing image-----"<<endl;
*out<<numRows<<" " << numCols <<" " << minVal <<" " << maxVal<<endl;
for(int i = 1; i < numRows+1; i++){
    for (int j = 1; j < numCols+1; j++){

        if(zeroFramedAry[i][j]==0)
            *out<<".";
        else

```

```

        *out<< zeroFramedAry[i][j];

    if(zeroFramedAry[i][j] < 10){
        *out<< " ";
    }
    else{
        *out<< " ";
    }
}
*out<<endl;
}
*out<<endl<<endl<<endl;
}

void printEQarr(ofstream *out){

    *out<<endl<<"-----Printing EQ Table-----"<<endl;
    for (int i = 1; i <= newLabel; i++){
        if(i<10)
            *out<< i << " ";
        else
            *out<< i << " ";
    }
    *out<<endl;

    for (int i = 1; i <= newLabel; i++){
        if(EQAry[i]<10)
            *out<<EQAry[i] << " ";
        else
            *out<<EQAry[i] << " ";
    }
    *out<<endl<<endl;

}

void printEQAll(){

```

```

cout<<endl<<"-----Printing EQ all Table-----"<<endl;
for (int i = 0; i < (int)(numRows * numCols) / 4; i++){
    if(i<10)
        cout<< i << " ";
    else
        cout<< i << " ";
}
cout<<endl;

for (int i = 0; i < (int)(numRows * numCols) / 4; i++){
    if(EQary[i]<10)
        cout<<EQary[i] << " ";
    else
        cout<<EQary[i] << " ";
}
cout<<endl<<endl;
}

```

```

bool case2(int a, int b, int c, int d, int r, int c){

```

```

    // if(r==5 && c==4){
    // cout<<"\ta:" << a<<endl;
    // cout<<"\tb:" << b<<endl;
    // cout<<"\tc:" << c<<endl;
    // cout<<"\td:" << d<<endl;
    // cout<<endl<<endl<<endl;
    // }

```

```

int arr[] = {a,c,b,d};
int last_non_0 = 0,
    zeroCount = 0,
    sum =0;

```



```

for (int i = 0; i < 4; i++){

    if(arr[i]!=0){
        last_non_0 = arr[i];
    }
    if(arr[i]==0){
        zeroCount++;
    }
    sum+=arr[i];
}

// cout<<a<<b<<c<<d<<sum<<sum/(4-zeroCount)<<last_non_0<<endl;
// cout<< (sum/(4-zeroCount)==last_non_0)<<"Reee"<<endl;

// if(r==5 && c==4){
// cout<<"sum is: " << sum<<endl;
// cout<<"zeroCOunt is: " << zeroCount<<endl;
// cout<<"last nonzero is: " << last_non_0<<endl;
// cout<<"4-zeroCount:" << 4-zeroCount<<endl;
// cout<<"sum/4-zeroCount:" << sum/(4-zeroCount)<<endl;
// cout<<"\ta:" << a<<endl;
// cout<<"\tb:" << b <<endl;
// cout<<"\tc:" << c<<endl;
// cout<<"\td:" << d<<endl;

// }

return (double) sum/(4-zeroCount)==last_non_0;

// return false;
}

int case3(int a, int b, int c, int d){

```

```

int arr[] = {a,b,c,d};

int min = 99999;

for (int i = 0; i < 4; i++)
{
    if(arr[i]<min && arr[i]!=0){
        min = arr[i];
    }
}

// cout<<"min is: " << min<<endl;
// cout<<"\ta:" << a<<endl;
// cout<<"\tb:" << b<<endl;
// cout<<"\tc:" << c<<endl;
// cout<<"\td:" << d<<endl;

return min;
}

bool case2Pass2(int e,int f,int g,int h,int pix,int r,int c){

int arr[] = {e,f,g,h,pix};
int last_non_0 = 0,
    zeroCount = 0,
    sum =0;

for (int i = 0; i < 4; i++){

    if(arr[i]!=0){
        last_non_0 = arr[i];
    }

    if(arr[i]==0){
        zeroCount++;
    }
}

```

```

        sum+=arr[i];
    }

    return (double) sum/(5-zeroCount)==last_non_0;

}

int case3Pass2(int e,int f,int g,int h,int pix,int r,int o){

    int arr[] = {e,f,g,h,pix};

    int min = 99999;

    for (int i = 0; i < 4; i++)
    {
        if(arr[i]<min && arr[i]!=0){
            min = arr[i];
        }
    }

    // cout<<"min is: " << min<<endl;
    // cout<<"\ta:" << a<<endl;
    // cout<<"\tb:" << b<<endl;
    // cout<<"\tc:" << c<<endl;
    // cout<<"\td:" << d<<endl;

    return min;
}

void connect8pass1(){

    //step 0
    newLabel = 0;
    int pix, i=1;

```

```

//step 4
for (int r = 1; r < numRows+1; r++){
    for (int c = 1; c < numCols+1; c++){

        //step 1
        pix = zeroFramedAry[r][c];

        int a = zeroFramedAry[r-1][c-1];
        int b = zeroFramedAry[r-1][c];
        int cl = zeroFramedAry[r-1][c+1];
        int d = zeroFramedAry[r][c-1];

        // step 2
        if(pix>0){

            //case 1
            if( a==0 && b==0 && cl==0 && d==0){
                newLabel++;
                pix = newLabel;
                zeroFramedAry[r][c] = pix;

                //step 3
                // cout<<"case 1"<<endl<< zeroFramedAry[r][c] <<" Row and Col: ("
<<r<<","<<c<<")"<<"====>"<<endl;

                EQAry[i] = pix;
                i++;

                // cout<<zeroFramedAry[r][c] <<endl;

                // cout<<"\ta:" << a<<endl;
                // cout<<"\tb:" << b<<endl;
                // cout<<"\tc:" << cl<<endl;
                // cout<<"\td:" << d<<endl;

            }

            //case 2 this is not right come back to it
            else if(case2(a,b,cl,d, r, c)==true){
                // cout<<"case2"<<endl;

```

```

        // if(r==5 && c==4){
        //     cout<<"case 2"<<endl<< zeroFramedAry[r][c] <<"  Row and Col: ("
<<r<<","<<c<<")"<<"====>"<<endl;

        //     cout<<"\ta:" << a<<endl;
        //     cout<<"\tb:" << b<<endl;
        //     cout<<"\tc:" << cl<<endl;
        //     cout<<"\td:" << d<<endl;
        //     cout<<"\t\tMIN is: "<<case3(a,b,cl,d)<<endl;
        // }
        pix = case3(a,b,cl,d);

        zeroFramedAry[r][c] = pix;

        // if(r==5 && c==4)
        // cout<<zeroFramedAry[r][c] <<endl;
    }

//case 3
else{
    // cout<<"case 3"<<endl<< zeroFramedAry[r][c] <<"  Row and Col: ("
<<r<<","<<c<<")"<<"====>"<<endl;

    pix = case3(a,b,cl,d);
    zeroFramedAry[r][c] = pix;
    // cout<<zeroFramedAry[r][c] <<endl;

    //     cout<<"\ta:" << a<<endl;
    //     cout<<"\tb:" << b<<endl;
    //     cout<<"\tc:" << cl<<endl;
    //     cout<<"\td:" << d<<endl;

    // //step 3
    // // EQAry[i] = pix;
    // cout<<"I: "<< i<<endl;

    EQAry[a] = pix;
    EQAry[b] = pix;
    EQAry[c] = pix;
    EQAry[d] = pix;

```

```
}
```

```
}
```

```
}
```

```
}
```

```
}
```

```
void connect8pass2(){
```

```
    int pix, i=1;
```

```
    EQAry[0] = 0;
```

```
    for (int r = numRows; r > 0; r--){
```

```
        for (int c = numCols; c > 0; c--){
```

```
            //step 1
```

```
            pix = zeroFramedAry[r][c];
```

```
            int e = zeroFramedAry[r][c+1];
```

```
            int f = zeroFramedAry[r+1][c-1];
```

```
            int g = zeroFramedAry[r+1][c];
```

```
            int h = zeroFramedAry[r+1][c+1];
```

```
            // step 2
```

```
            if(pix>0){
```

```
                //case 1
```

```
                if(e==0 && f==0 && g==0 && h==0){
```

```

        //do nothing
    }

    //case 2 this is not right comeback to it
    else if(case2Pass2(e,f,g,h,pix,r,c)){
        // zeroFramedAry[r][c] = pix;

    }

    //case 3
    else{
        int minL = case3Pass2(e,f,g,h,pix,r,c);

        if(pix > minL){
            EQAry[pix] = minL;
            pix = minL;
            zeroFramedAry[r][c] = pix;
        }
    }

}

//step 3
pix = EQAry[pix];
zeroFramedAry[r][c] = pix;

}

}

}

void connect4pass1(){
    //step 0
    newLabel = 0;
    int pix;

```

```

int i=1;

//step 4
for (int r = 1; r < numRows+1; r++){
    for (int c = 1; c < numCols+1; c++){
        //step 1
        pix = zeroFramedAry[r][c];

        int b = zeroFramedAry[r-1][c];
        int d = zeroFramedAry[r][c-1];

        // step 2
        if(pix>0){

            //case 1
            if( b==0 && d==0){
                newLabel++;
                pix = newLabel;
                zeroFramedAry[r][c] = pix;
                //step 3
                // i++;
                EQAry[i] = pix;
                // cout<<" case 1 "<<endl;
                // cout<<"EQ[i] = " << EQAry[i];
                // cout<<" i = " << i<<endl;
                i++;

                // cout<<"case 1"<<endl;

            }

            //case 2 this is not right come back to it
            else if(b==d || b==0 || d==0){
                pix = max(b,d);
                zeroFramedAry[r][c] = max(b,d);
                // cout<<" case 2 "<<endl;
                // cout<<"EQ[i] = " << EQAry[i];

```



```

        // cout<<" i = " << i<<endl;

    }

    //case 3
    else if(b!=d){
        pix = min(d,b);
        zeroFramedAry[r][c] = pix;

        //step 3

        // i++;

        EQAry[i] = pix;
        EQAry[max(b,d)] = pix;

        // cout<<" case 3 "<<endl;

        // cout<<"EQ[i] = " << EQAry[i];

        // cout<<" i = " << i<<endl;

        // i++;

    }

    //step 3

    //  cout<<"ROW: "<<r<<" COL: "<<c<<"  "<<zeroFramedAry[r][c]<<endl;

    }

}

}

}

// cout<<endl<<"I AM DONE" << endl;

}

void connect4pass2(){

    //step 0

    int pix, i=1;

```

```

//step 4
for (int r = numRows; r > 0; r--){
    for (int c = numCols; c > 0; c--){

        //step 1
        pix = zeroFramedAry[r][c];

        int e = zeroFramedAry[r][c+1];
        int g = zeroFramedAry[r+1][c];

        // step 2
        if(pix>0){

            //case 1
            if(e==0 && g==0){
                //do nothing
            }

            //case 2 this is not right comeback to it
            else if( (e==g && g==pix) || (e==pix && g==0) || (g==pix && e==0) || (e==0 || g==0)){
                zeroFramedAry[r][c] = pix;
            }

            //case 3
            else{
                int minL = min(e,g);

                if(pix > minL){
                    EQAry[pix] = minL;
                    pix = minL;
                    zeroFramedAry[r][c] = pix;
                }
            }
        }
    }
}

```

```

        //step 3
        pix = EQAry[pix];
        zeroFramedAry[r][c] = pix;

    }

}

}

int manageEq(){

    // step 0
    int readLabel = 0;

    // step 1
    int index = 1;

    // step 4
    while (index <= newLabel){
        // step 2
        if(index != EQAry[index]){

            EQAry[index] = EQAry[EQAry[index]];

        }else{
            readLabel++;
            EQAry[index] = readLabel;
        }
        //step 3
        index++;
    }

    return readLabel;
}

```

```
}
```

```
void connected4(ofstream *pretty, ofstream *debug){

    *debug << "-----entering connected 4 method-----"<<endl;
    *pretty<<"----Printing Original Table-----"<<endl;
    imgReformat(pretty);

    //step 1
    connect4pass1();
    *debug << endl<<"After Connected4Pass1, newLabel = " << newLabel<<endl;
    *pretty << endl<< "After Connected4Pass1, newLabel = " << newLabel<<endl;
    imgReformat(pretty);
    printEQarr(pretty);

    imgReformat(debug);
    printEQarr(debug);

    //step 2
    connect4pass2();
    *debug << "After Connected4Pass2, newLabel = " << newLabel<<endl;
    *pretty << "After Connected4Pass2, newLabel = " << newLabel<<endl;
    imgReformat(pretty);
    printEQarr(pretty);

    imgReformat(debug);
    printEQarr(debug);

    //step 3
    *debug << "----Running Managing EQ ARRAY---- ";
    *pretty << "----Running Managing EQ ARRAY---- ";
    trueNumCC = manageEq();
    printEQarr(pretty);
    printEQarr(debug);
    newMin = 0;
    newMax = trueNumCC;
    CCproperty = new Property[trueNumCC+1];
```

```

* debug << endl << "In Connected4, after manage EQarr, trueNumCC = " << trueNumCC << endl;
* pretty << endl << "In Connected4, after manage EQarr, trueNumCC = " << trueNumCC << endl << endl;

//step 4
connectPass3(debug);

//step 5
* debug << endl << "-----After Pass3-----" << endl;
* pretty << endl << "-----After Pass3-----" << endl;
imgReformat(pretty);
imgReformat(debug);

//step 6
printEQarr(pretty);
printEQarr(debug);

//step 7
* debug << "Leaving connected4 method" << endl;

}

void connected8(ofstream *pretty, ofstream *debug){

* debug << "-----entering connected 8 method-----" << endl;
* pretty << "----Printing Original Table-----" << endl;

//step 1
connect8pass1();
* debug << "After Connected8Pass1, newLabel = " << newLabel << endl;
* pretty << endl << "After Connected4Pass1, newLabel = " << newLabel << endl;
imgReformat(pretty);
printEQarr(pretty);

imgReformat(debug);
printEQarr(debug);

//step 2

```

```

connect8pass2();

* debug << "After Connected8Pass2, newLabel = " << newLabel<<endl;
* pretty << "After Connected4Pass2, newLabel = " << newLabel<<endl;

imgReformat(pretty);
printEQarr(pretty);


imgReformat(debug);
printEQarr(debug);


//step 3
* debug << "----Running Managing EQ ARRAY---- ";
* pretty << "----Running Managing EQ ARRAY---- ";

trueNumCC = manageEq();
printEQarr(pretty);
printEQarr(debug);

newMin = 0;
newMax = trueNumCC;
CCproperty = new Property[trueNumCC+1];
* debug << "In Connected8, after manageEQarr, trueNumCC = " << trueNumCC <<endl;
* pretty <<endl<< "In Connected4, after manage EQarr, trueNumCC = " << trueNumCC <<endl<<endl;


//step 4
connectPass3(debug);


//step 5
* debug <<endl<< "-----After Pass3-----"<<endl;
* pretty <<endl<< "-----After Pass3-----"<<endl;

imgReformat(pretty);
imgReformat(debug);


//step 6
printEQarr(pretty);
printEQarr(debug);

//step 7
* debug << "Leaving connected8 method" <<endl;
}

```

```
};
```

```
int main(int argc, const char* argv){
```

```
    ifstream infile;
```

```
    infile.open(argv[1]);
```

```
    int connectness = stoi(argv[2]);
```

```
    char option = *argv[3];
```

```
    ofstream pretty, label, property, debug;
```

```
    pretty.open(argv[4]); //pretty print
```

```
    label.open(argv[5]); //label print
```

```
    property.open(argv[6]); //property print
```

```
    debug.open(argv[7]); //debug print
```

```
    int numRows, numCols, minVal, maxVal, newLabel = 0;
```

```
    infile >> numRows >> numCols >> minVal >> maxVal;
```

```
    ccLabel *proj = new ccLabel(numRows, numCols, minVal, maxVal ,option);
```

```
    //step 2
```

```
    proj->loadImage(&infile, &pretty);
```

```
    //step 3
```

```
    if(option == 'y' || option == 'Y'){
```

```
        proj->conversion();
```

```
    }
```

```
    //step 4
```

```

if(connectness == 4){
    proj->connected4(&pretty, &debug);
}

//step 5
if(connectness == 8){
    proj->connected8(&pretty, &debug);
}

//step 6
label << "numRows: " << numRows<< " numCols: " << numCols<< " minVal: " << minVal << " maxVal: " <<
maxVal<<endl;

//step 7
proj->printImg(&label);

//step 8
property << "CCProperties"<<endl;

pretty <<endl<<"Number of Conected Components: " << proj->>trueNumCC<<endl<<endl;

//step 9
proj->drawBoxes();
debug<<endl<<"-----Result of Drawing Boxes-----"<<endl;
pretty<<endl<<"-----Result of Drawing Boxes-----"<<endl;
proj->imgReformat(&debug);
proj->printProperty(&property);

//step 10
proj->imgReformat(&pretty);

//step 11

proj->connect8pass1();

//step 12

```



```
pretty.close(); //pretty print  
label.close(); //label print  
property.close(); //property print  
debug.close();
```

```
}
```


