Question solution:

In this problem, I adopted a recursive algorithm.

First, I surround the pixels of the input image with zeros to prevent boundary problems. And I define a counter to record the group number, the counter starts with 2 to distinguish from the value of pixel which is 1.

Then, I traverse the pixel matrix from left to right, from top to bottom. When find a pixel which value is 1 (that means find a new component), I replace the value of the pixel with the value of counter and use a recursive algorithm to find its 4-connectivity or 8-connectivity connected components (the values of these components are also be replaced with the value of counter). After all the components are found, the value of the counter is increased by 1. And so on.

Finally, I subtract 1 from all group numbers as the counter starts with 2.

Results:

For 4-connectivity:

0 0 0 0 1 1 0 0 0 2 0 3 0 4 4 4 0 0 5 5
6 0 7 0 0 0 0 0 2 2 0 0 0 0 4 0 0 8 0 0
0 7 7 7 7 7 7 0 2 0 9 0 10 0 0 0 0 0 11 0
0 0 0 0 7 7 0 0 0 0 9 0 0 0 11 11 11 0 11 11
12 0 0 12 0 0 0 0 0 0 9 9 9 0 11 11 11 0 11 11
12 12 12 12 0 13 0 0 0 0 0 0 0 14 0 0 11 11 0 11
12 12 0 0 0 0 15 0 0 0 0 16 16 0 0 0 0 11 0 11
0 12 0 0 0 15 15 0 0 0 0 16 16 0 0 0 0 11 11 11
0 0 17 17 17 0 0 18 0 18 0 0 16 0 0 0 11 11 11 0
19 0 17 0 17 0 18 18 18 18 0 0 16 0 20 0 0 0 0 21

For 8-connectivity:

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