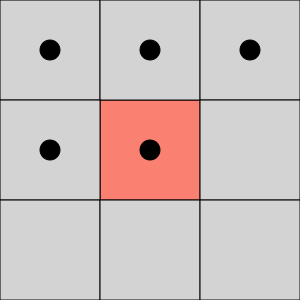
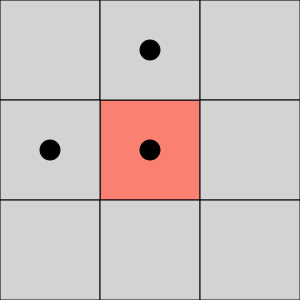
**Connected Component Analysis**

Connected Component Analysis can be of two types

* 4 Connected Neighbour
* 8 Connected Neighbour



4 Connectivity 8 Connectivity

**Algorithms**

### **One component at a time:** It is based on graph traversal methods. Starting from one pixel and propagating through its children.

1. Start from the first pixel in the image. Set current label to 1. Go to (2).
2. If this pixel is a foreground pixel and it is not already labeled, give it the current label and add it as the first element in a queue, then go to (3). If it is a background pixel or it was already labeled, then repeat (2) for the next pixel in the image.
3. Pop out an element from the queue, and look at its neighbors (based on any type of connectivity). If a neighbor is a foreground pixel and is not already labeled, give it the current label and add it to the queue. Repeat (3) until there are no more elements in the queue.
4. Go to (2) for the next pixel in the image and increment the current label by 1.

### **Two-pass:**

**Connectivity checks:**

1. Does the pixel to the left (West) have the same value as the current pixel?
   1. **Yes** – We are in the same region. Assign the same label to the current pixel
   2. **No** – Check next condition
2. Do both pixels to the North and West of the current pixel have the same value as the current pixel but not the same label?
   1. **Yes** – We know that the North and West pixels belong to the same region and must be merged. Assign the current pixel the minimum of the North and West labels, and record their equivalence relationship
   2. **No** – Check next condition
3. Does the pixel to the left (West) have a different value and the one to the North the same value as the current pixel?
   1. **Yes** – Assign the label of the North pixel to the current pixel
   2. **No** – Check next condition
4. Do the pixel's North and West neighbors have different pixel values than the current pixel?
   1. **Yes** – Create a new label id and assign it to the current pixel

**Implementation:**

**On the first pass:**

1. Iterate through each element of the data by column, then by row (Raster Scanning)
2. If the element is not the background
   1. Get the neighboring elements of the current element
   2. If there are no neighbors, uniquely label the current element and continue
   3. Otherwise, find the neighbor with the smallest label and assign it to the current element
   4. Store the equivalence between neighboring labels

**On the second pass:**

1. Iterate through each element of the data by column, then by row
2. If the element is not the background
   1. Relabel the element with the lowest equivalent label

