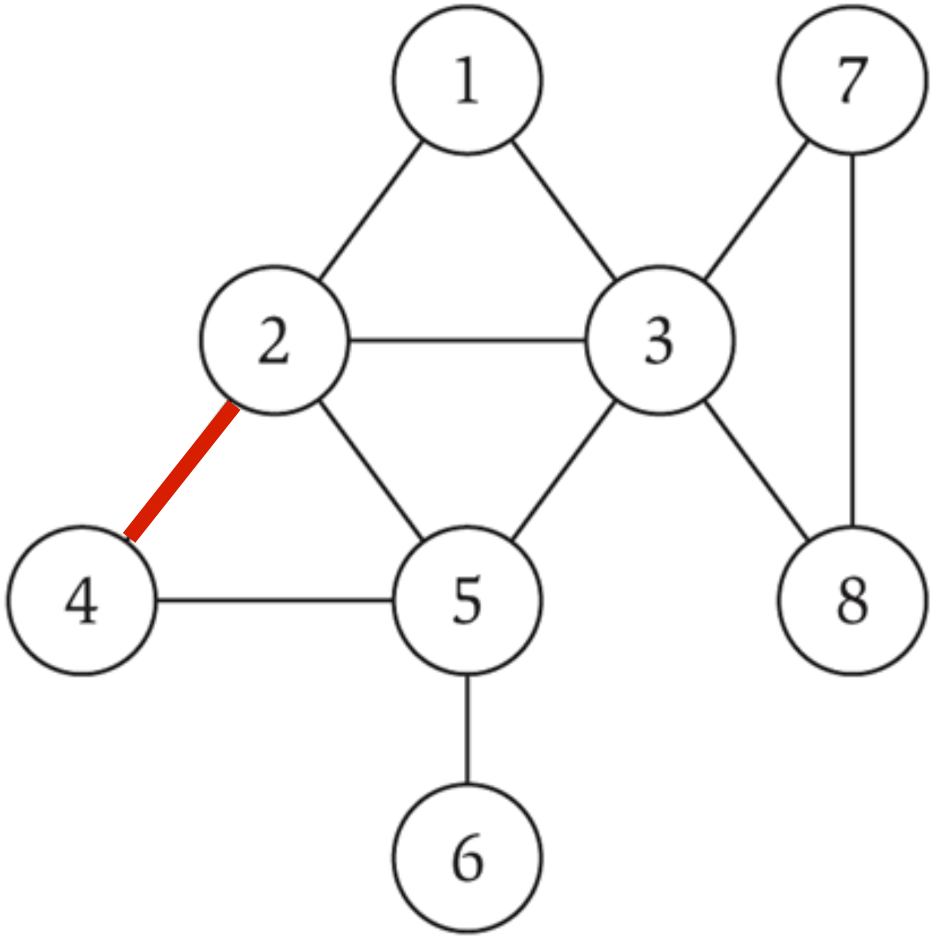
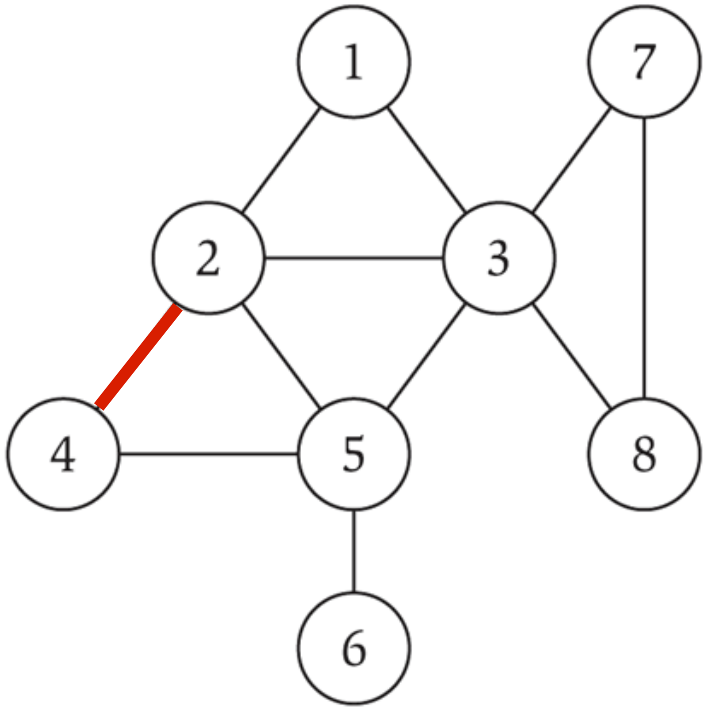
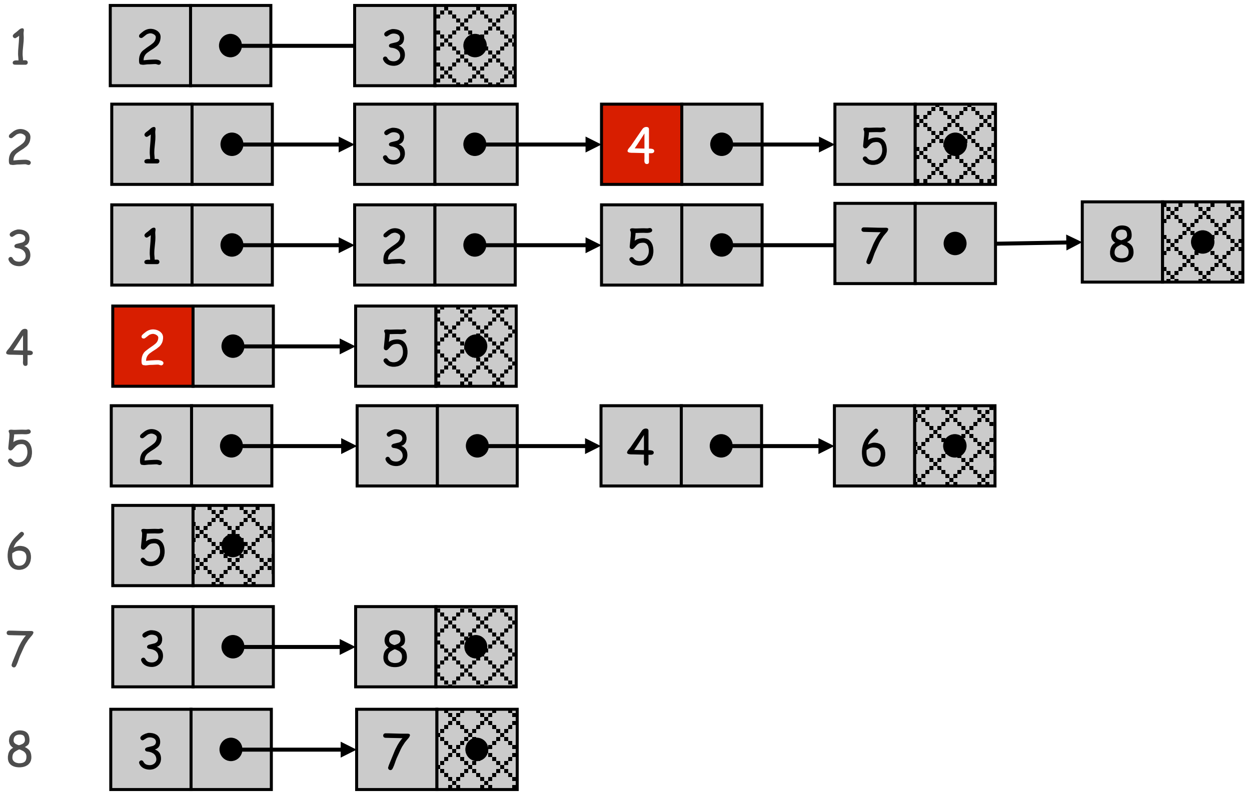
Adjacency Matrix Visualized



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

ADJACENCY LISTS VISUALIZED





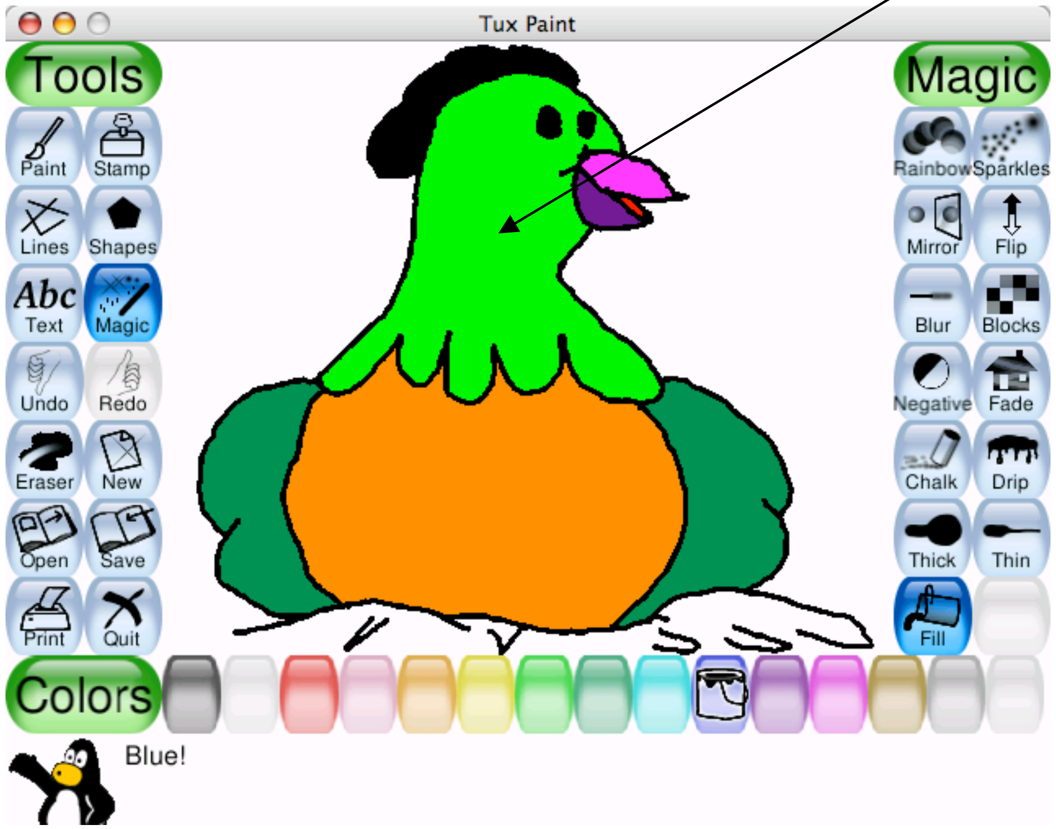
Connected Components: Practically

Flood Fill

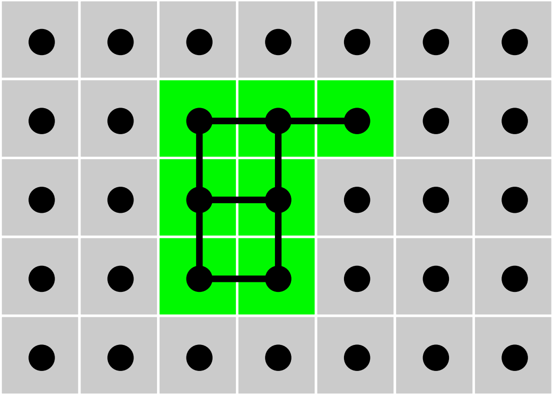
Given a lime green pixel in an image, change the color of the entire blob of neighboring lime pixels to blue.

- Node: pixel
- Edge: two neighboring lime pixels
- Blob: connected component of lime pixels

recolor lime green blob to blue



Blue!



```
DFS(u):
  mark u as explored
  for each edge (u,v):
    if v is not explored:
      //add edge (u,v) to T
      DFS(v)


return all explored nodes
```

can think of it constructing DFS tree T

```
BFS(u):
  mark u as discovered and store in queue Q
  while Q is not empty
    dequeue v from Q
    for each edge (v,w):
       if w is not discovered:
          mark w as discovered
          append w to Q
          //add edge (v,w) to T

return all discovered nodes
```

can think of it
constructing BFS
tree T

Topological order visualized