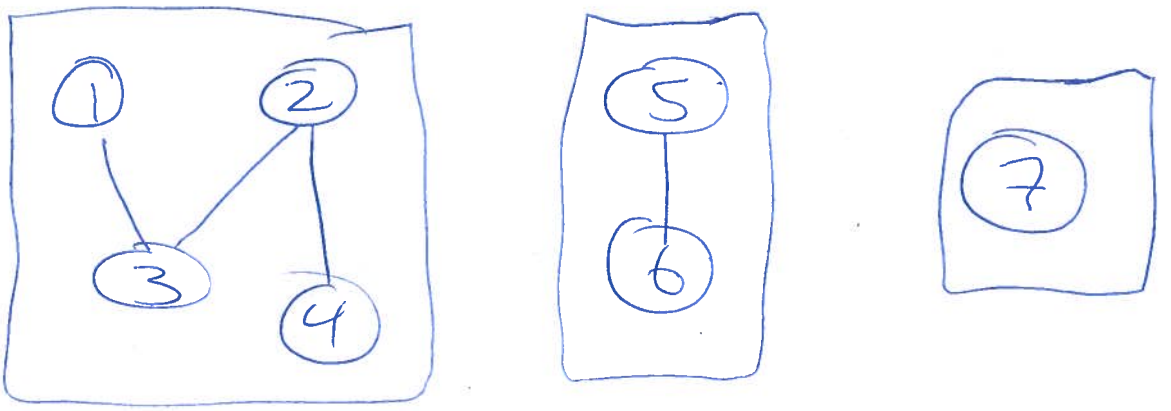

3 connected components

Run BFS(1) = {[1], [2, 3], [4]}
(L_0 L_2 L_1 L_3)

BFS(5) = {[5], [6]}
(L_0 L_1)

BFS(7) = {[7]}
(L_0)

$n_{BLUE} = 2$

$n_{GREEN} = 1$

$n_{RED} = 1$

If $u \in V$, n_u is the number of edges u has. (# of neighbors of u)

Recall that input is your graph and vertex s.

G is an adjacency list

Size of input is $n + \sum_{u \in V} n_u = O(n+m)$

(m is $O(n^2)$)

```
BFS(s)

CC[s] ← 1,  CC[u] ← 0 ∀u ≠ s

L[0] ← {s}

i ← 0

while (L[i] ≠ ∅)
    L[i+1] = ∅
    For every u ∈ L[i]
        For every (u,w) ∈ E
            If CC[w] = 0        // If w is unvisited
                CC[w] ← 1       // set w as visited
                Add w to L[i+1]

    i ← i+1

End
```