S. KIMMEL

Idea, explore layers.

```
Exp[v] = 0    ∀ v∈V    // mark True when "explored"
A = {};
A.add(s)
Exp[s] = True
```

A is a queue = "First in, first out"

```
while(A is not empty) {
    v = A.pop
    For each edge (v,w)        {
       If (Ex[w] = false) { Ex[w] = True; A.add(w); }
    }
}
```

This is Breadth First Search - slowly move away in layers from initial node

S. KIMMEL

Explain: Why is runtime $O(n+m_s)$

- Looks at edges (in for loop)
- Each edge can only be examined when its adjoining vertex is popped. Each vertex can only show up in QUEUE one time. $\Rightarrow$ Each edge is examined once.
- Looking at each edge takes constant time b/c use adjacency list.

$O(m_s)$ to do while loop ← # edges connected to s

+

$O(n)$ to initialize

$=$

$O(n+m_s)$

S. KIMMEL

Alg

Pop
Edge
Edge
Edge
Pop
Edge
Edge
Edge
Edge
Edge
Pop
Edge
Edge
Pop
E
E
E
E

runtime = n + # of edges looked at

An edge  is looked at when v or u is popped from queue.

v and u can each only be in queue once. ⇒ each edge is looked at twice