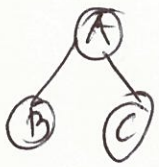
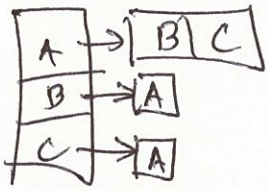
Sep 28

(Undirected) $G = (V, E)$

n_u = # neighbors of u = $|\{w \mid (u,w) \in E\}|$

(degree of u)





↳ 3 ptr +

size of lists = 2 = n_A (list of nbrs of A for A)

1 = n_B

1 = n_C

$n = 3$

$m = 2$

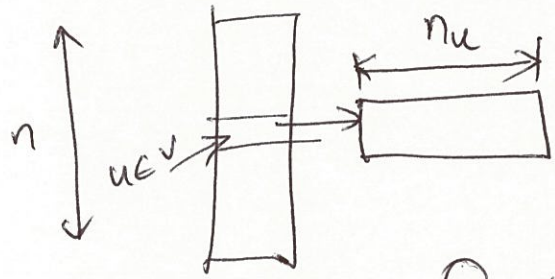
Overall = $3 + 2+1+1 = 3+4 = 3+2\cdot2$

Adj list in general

#ptrs = $|V| = n$

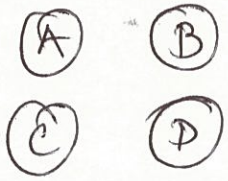
sum of list sizes = $\sum_{u \in V} n_u$

Just argued → $= 2m$



⇒ overall size = $n + 2m$ $\leq O(n^2)$

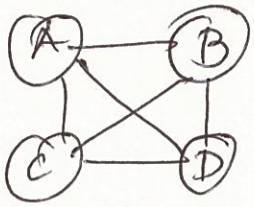
$= \Theta(n+m)$



$0 \leq m \leq \binom{n}{2}$

$= \frac{n(n-1)}{2}$

$\leq \frac{n^2}{2}$



⇒ $m \leq O(n^2)$

array of size n

```
BFS(G,s)   // G is in adjacency list rep.
0.  CC[s] ← T  and  CC[u] ← F  ∀ u ≠ s ∈ V        O(n)
1.  i ← 0                                          O(1)
2.  L_0 ← {s}        (empty set)
3.  While L_i ≠ ∅    (all s these incl)            O(n)
      3.1  L_{i+1} ← { }   // O(1)
      3.2  for u in L_i
             for (u,w) ∈ E
               If CC[w] = F
                 CC[w] ← T
                 Add w to L_{i+1}                  O(1)
      3.3  i ← i+1                                 O(n)
4.  Return CC   //   CC(s) = { w | CC[w] = T }
```

How many times this block is executed: T

Overall $= O(n) + T \cdot O(1)$

$\leq O(n+m)$ If $T \leq O(m)$