$\Rightarrow \exists (x,y) \in E$ s.t $x \in R^*$, $y \notin R^*$

$\Rightarrow$ y should have been added to R by Explore

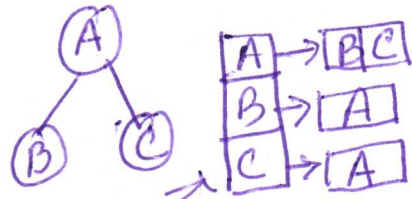
$\Rightarrow$ Explore should not have terminated.

$\Rightarrow$ contradict (*) □

Main1

Undirected $G = (V, E)$

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



$n_A = 2$

$n_B = 1$

$n_C = 1$

Total size = 3 ptrs + sum of list sizes.

list sizes: $2 = n_A$ for A

$1 = n_B$ for B

$1 = n_C$ for C

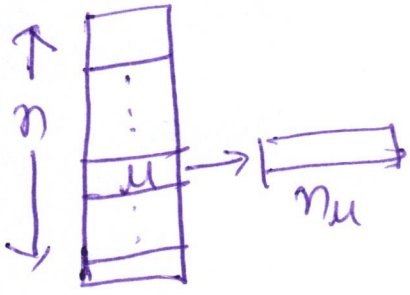
$\Rightarrow 3 + n_A + n_B + n_C$

$= 3 + 2 + 1 + 1 = 7$

Adj. list in general:

ptrs = $|V| = n$

sum of list sizes: $\sum_{u \in V} n_u = 2m$ between n and n^2

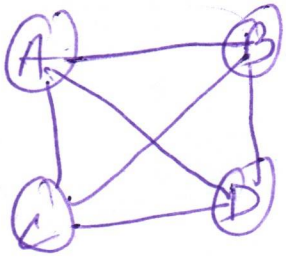


$\Rightarrow$ Overall size = $n + 2m = O(n+m)$

$0 \le m \le \binom{n}{2} = \frac{n(n-1)}{2} \le n^2$

(A) (B)

(C) (D)



BFS(G, s) // G is in adj. list format

0. cc[s] ← T and cc[v] ← F ∀ v ≠ s ∈ V
   O(1) ↑ assignment — O(n)

O(n)
1. i ← 0 // O(1)
2. L_i ← {s} // O(1)

3. while $L_i \neq \emptyset$ ← empty set // T_1 be the # of iterations
   3.1 L_{i+1} ← { } // O(1)
   3.2 for all u in L_i
       for (u, w) ∈ E // T_{12} be the times that algo. gets here
         if cc[w] = F // (T_{123} be the times algo. gets here)
           cc[w] ← T
           add w to L_{i+1} — O(1)
   3.3 i++ // O(1)

4. Return cc // connected component
   ↑ O(n) of s = {w | cc[w] = T}

Total runtime: $O(n) + T_1 \cdot O(1) + T_{123} \cdot O(1) + O(n)$

$\leq O(n) + O(T_{123})$ ← Goal: $T_{123} \leq O(m)$

⇒ $O(n) + O(m)$

$O(n+m)$