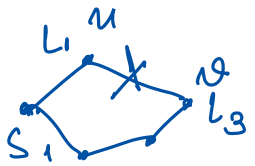
Applications (Contd.)

Testing bipartiteness using BFS:

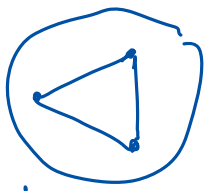
G is connected.

↳ Start with a vertex s.

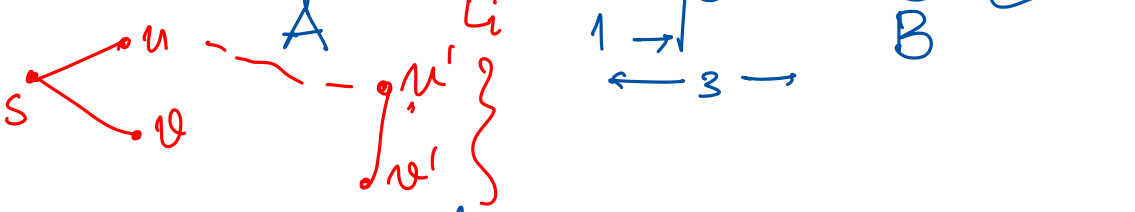


$L_0 = \{s\}$

$L_1, L_2, \ldots, L_k$



$(L_0 \cup L_2 \cup \cdots \cup L_k)$ $(L_1 \cup L_3 \cup \cdots \cup L_{k-1})$



Say $u \in L_1$ and $v \in L_3$.

Shortdistance $(s,u) = 1$
Shortdistance $(s,v) = 3$

} If $(u,v) \in E(G)$ then shortdistance $(s,v) = 2$.
This contradicts the shortest distance property of BFS.

Claim: There are no edges between any pair of vertices in part A (part B).

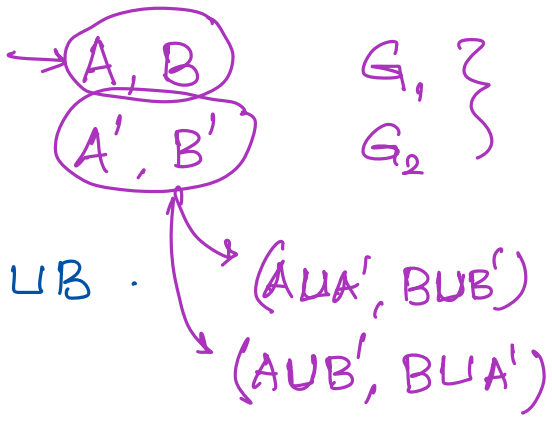
Pf:

Edges → Go across the layers of BFS → Contradicts SD property
→ Within the same layer of BFS → odd cycles.

Algo for testing bipartiteness:

Input: Graph (V, E)

Output: Graph (A, B, E) s.t $V = A \cup B$.



- Run BFS
- Club even layers into A, odd layers into B.
- If there are edges between pairs of vertices in either A or B:

  return Fail.

  Else return (A, B, E).

Connectedness for directed graphs:

A graph G is strongly connected if $\forall$ pairs $(u, v) \in G$, $\exists$ a directed path from u to v and v to u.

↳ In other words: Every vertex in the strongly connected component is part of a closed walk from s.

BFS: Start from s.

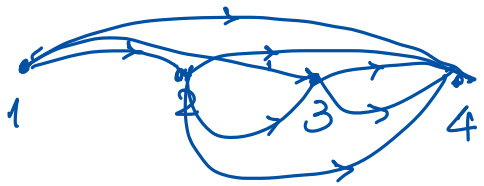
↳ BFS tree contains all nodes reachable from s.

Directed Acyclic Graphs (DAGs).

↳ Directed graph with no cycles.

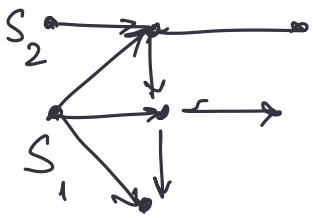
↳ ∃ a vertex with no outgoing edges.

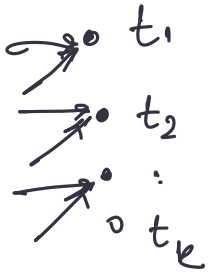
- n node graph w/ ≥ n edges ✗ ✗



no cycles

- If every vertex has an outgoing edge then over walks of long enough length, a cycle appears.

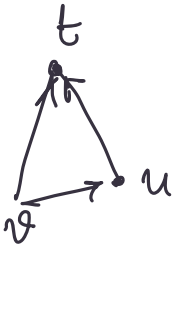




Order $\geq$: Transitive.

We say node $u \geq$ if $\exists$ directed ~~edge~~ path u to v.

$$u \geq v \;\&\; v \geq w \Rightarrow u \geq w$$



$v \leq t$

$u \leq t$

$v \leq u$.

Courses → DAG

C_1 is a prereq to C_2