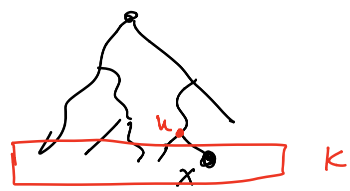
Claim: Any non-tree connects nodes at the same or at two consecutive level of BFS tree

Pf: Fix an edge (x, y). Goal: $|L(x) - L(y)| \leq 1$

Assume that x is discovered first, also say $L(x) = k$.

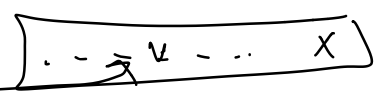
Without loss of generality



Goal: show $L(y) \leq k+1$.

BC of queue, all vertices v currently in satisfy $L(v) \leq k = L(x)$

Case 1: y discovered before processing x (say by node v).



$\Rightarrow$ $k \leq L(y) = L(v) + 1 \leq k+1$ ✓

BC y discovered after x.

Case 2: y discovered when we process $x \Rightarrow L(y) = L(x) + 1$

(x, y) will be a tree edge ✓ ∎

Claim: For all vertices v, $L(v)$ = length of shortest path from s to v.

Pf: $L(v) \leq$ shortest path & $L(v) \geq$ shortest path ✓

In BFS tree $\exists$ path of length $L(v)$ from $s \to v$.

[this is a candidate for shortest path $\Rightarrow$ shortest path $\leq L(v)$]

Now we $L(v) \leq$ shortest path.

Say $v_0, v_1, \ldots, v_k$ is the shortest path

$v_0 = s$, $v_k = v$

Goal: $L(v) = L(v_k) \leq k$.

e.g. suppose shortest path was $s, v \Rightarrow (s,v)$ an edge $\Rightarrow L(v) \le L(s)+1 = 1$

$s = v_0$, $v = v_1$

previous Lem

(v_0, v_1) an edge $\Rightarrow |L(v_0) - L(v_1)| \le 1 \Rightarrow L(v_1) \le L(v_0)+1$

(v_1, v_2) " " $\quad |L(v_1) - L(v_2)| \le 1 \Rightarrow L(v_2) \le L(v_1)+1$

$\vdots$

$L(v_k) \le L(v_{k-1})+1$

$+$

$L(v_0) = 0, \ L(v_1) \le 1, \ L(v_2) \le 2 \ldots \quad L(v_k) \le k$

In Class Exercise: Given a graph G s.t. $\deg(v) \le k$ for all v. Prove that we can color vertices of G with $k+1$ colors s.t. endpoint of any edge have distinct colors.

Pf: Do it by induction on n. Think of k as fixed.

$P(n)$: Color vertices of G with n vertices using $k+1$ colors if $\deg(v) \le k$.

Base Case $n=1$ •

IH $P(n-1)$.

IS: Given arbitrary G with n vertices s.t. $\deg(v) \le k$ for all v.

Goal: Color G with $k+1$ colors.

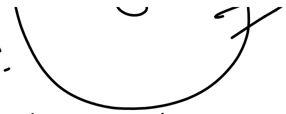
$G' = G - v$ (v arbitrary)

$\Rightarrow$ any vertex of G' has deg $\le k$.

G' has $n-1$ vertices

$\overset{\text{by IH}}{\Rightarrow}$ color G' using $k+1$ colors.

To color G: all vertices

that are in G' will have same color. 

v has $\leq k$ neighbors $\Rightarrow$ $\exists$ color not used on neighbors of v. Color v with that color. $\square$