CMPS 101 HW6

Prove:

1) Graph G is a connected graph satisfying $|E(G)| \geq |V(G)| - 1$

Proof w/ induction on # of edges.

Base Step: Let $n = |V(G)|$

Let $m = |E(G)|$, if $m = 0$, G, being connected can only have 1 vertex so $n = 1$ $\therefore m \geq n - 1$ thus base case satisfied.

∎

Now let $m > 0$ and assume for any connected graph G' w/ fewer than m edges that $|E(G')| \geq |V(G')| - 1$. Now remove any edge $e \in E(G)$ denoting the resulting sub-graph as $G - e$. This leads to two cases.

Case 1: $G - e$ is connected, w/ Lemma 1 we note $m = n - 1$ and w/ induction hypothesis we get $m - 1 \geq n - 1$. And $m \geq n - 1$ as claimed

Case 2: $G - e$ is dis-connected. Meaning $G - e$ consists on two sub-graphs, K_1 and K_2, w/ both having less than m edges. Suppose K_i has m_i edges and n_i vertices $(i = 1, 2)$. The induction hypothesis shows $m_i \geq n_i - 1$ $(i = 1, 2)$. And $n = n_1 + n_2$ since no vertices were removed

$$\therefore m = m_1 + m_2 + 1 \geq (n_1 - 1) + (n_2 - 1) + 1 = n_1 + n_2 - 1 = n - 1$$

$$\therefore m \geq n - 1$$

∎

2) Show ij^{th} entry in A^d is the # of walks in G of length d from vertex i to vertex j.

Proof: Observe that $A^0 = I$, the identity matrix, which consists of 1 is each ij^{th} entry with $i=j$. And 0 when $i \neq j$. Also there is a walk of length 0 each, the trivial walk, occuring when there are no edges and zero walks when $i \neq j$. Thus base case satisfied.

Now let $d \geq 0$ and assume $(A^d)_{ij}$ is the # of walks of length d from i to j. The def. of matrix multiplication gives us.

$$(A^{d+1})_{ij} = (A^d \cdot A)_{ij} = \sum_{k=1}^{n} (A^d)_{ik} \cdot A_{kj}$$

w/ a walk from i to j of length d+1 is in two parts. ① A walk in length of from i to some intermediate vertex K, followed by part ② traversing one edge from K to j. Part ① can be represented as $(A^d)_{ik}$ ways by the induction hypothesis. Part ② only exists if is an edge b/t K & j

∴ can be completed in A_{kj} ways, 1 is there is a path or 0 if there is no edge. Summing over all possible K gives the total # of walks from i to j of length d+1. Through $(A^d)_{ik} \cdot A_{kj}$ representing the # of walks of length d+1 from i to j of length d+1, is equivalent to $(A^{d+1})_{ij}$. Thus the result now follows for all d by induction.

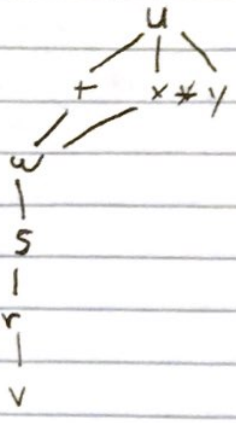
3) The distance from vertex i to vertex j is equal to $\min\{d \mid (A^d)_{ij} > 0\}$ Here $(A^d)_{ij}$ denotes the ij^{th} entry in matrix A^d.

From question 2 we know the number of walks of length d is $(A^d)_{ij}$. The distance from the vertex is the min walks of length d b/c a walk is any traversal.

4 a) Let u be the source

	adj	distance	parent
r	s v	~~4~~	s
s	r w	~~3~~	w
t	u w x	~~1~~	u
* u	t x y	0	nil
v	r	~~5~~	r
w	s t x	~~2~~	t
x	t u w y	~~1~~	u
y	u x	~~1~~	u

BFS TREE:

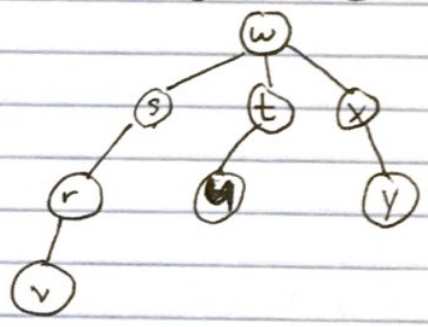


Q: ~~y~~ ~~t~~ ~~x~~ ~~r~~ ~~w~~ s ~~t~~ ~~v~~

b) Let w be the source

	adj	d	p	Q:
r	s v	2	s	~~w~~
s	r w	1	w	~~s~~
t	u w x	1	w	~~t~~
u	t x y	2	t	~~x~~
v	r	3	r	~~r~~
* w	s t x	0	nil	~~u~~
x	t u w y	1	w	~~y~~
y	u x	2	x	~~v~~

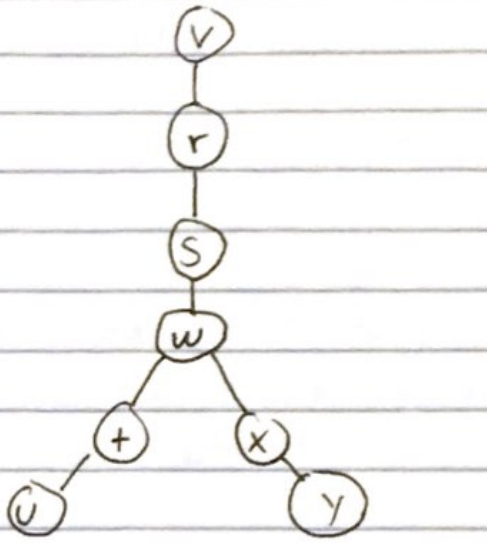
BFS TREE:



c) Let v be source

	adj	d	p	Q
r	s v	1	v	~~✓~~
s	r w	2	r	~~✓~~
t	u w x	4	w	~~✓~~
u	t x y	5	t	~~✓~~
* v	r	0	nil	~~✓~~
w	s t x	3	s	~~✓~~
x	t u w y	4	w	~~✓~~
y	u x	5	x	~~✓~~

BFS TREE:



5 what is the runtime

~~|V| = n, |E| = m~~

~~initialization would be Θ(n)~~

~~the queue operations would be O(n)~~

~~and scanning adj would be Θ(m)~~

~~adj list length = 2m if undirected, m if directed~~

~~thus running time would be Θ(n+m)~~

Answer: Since the input would be an adjacency matrix then the time it would take to go through all the edges is $O(V^2)$
thus the running time becomes $O(V+V^2)$

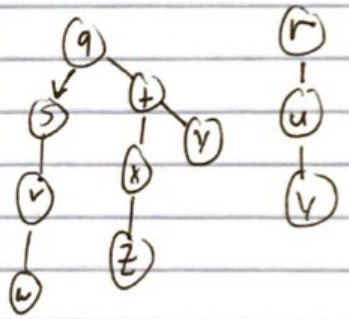
6. I would create a graph having each vertex represent a wrestler and each edge a rivalry. Graph would contain n vertices and r edges.

Next BFS would be performed to visit all the vertices. Then assign each vertex which had even distances to be "good guys" and odd to be "bad guys".
Then check each edge to verify it goes b/t a good guy & bad guy.

This will take O(n) to assign each vert to be good or bad and O(r) time to check rivalries (edges) thus O(n+r) time overall.

7.

	adj	d	f	p
q	s + w	1	16	n
r	u y	17	20	n
s	v	2	7	q
+	x y	8	15	q
u	y	18	19	r
v	w	3	6	s
w	s	4	5	q
x	z	9	12	+
y		13	14	+
z	x	10	11	x



Tree edges:
(q,s) (s,v), (v,w) (q,+)
(+,x) (x,z) (+,y) (r,u)

Back edges: (w,s) (z,x) (y,q)
forward edges: (q,w)
cross edges: (r,y) (u,y)