GT lecture notes

Graph Theory

These are the notes for graph theory course. The professor says it'll be a simple course, let's see about that. I am using Obsidian and this is an amazing markdown editor! It has a lot of community plugins. Anyways, study now... xD

Here is a somewhat detailed overview.

1. GT/Lecture 1 : Introduction to the course and grading. Defining graphs
2. GT/Lecture 2 : Something more here
3. GT/Lecture 3 : Something more

L(1)

Overview

This was the first lecture of graph theory. The professor is super old and kind lol. Course summary: Proof based and fun questions. Cutoffs are absolute. For A, aim for 80%+ (cutoff at 75%). The remaining grades are out of question anyways. A very interesting thing which came up during the meet is the page rank algorithm.

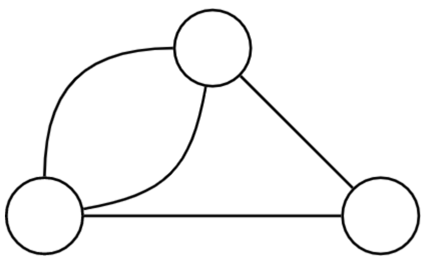
Defining graphs

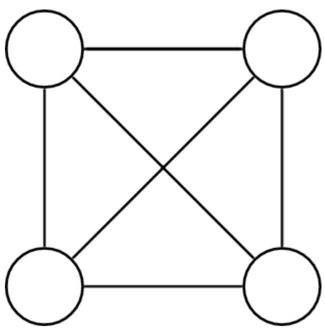
- The normal way: A graph G can be defined as a pair (V, E), where V is a set of vertices, and E is a set of edges between the vertices $E \subseteq \{(u, v) | u, v \in V\}$. If the graph is undirected, the adjacency relation defined by the edges is *symmetric*, or $E \subseteq \{\{u, v\} | u, v \in V\}$ (sets of vertices rather than ordered pairs). If the graph does not allow *self-loops*, adjacency is *irreflexive*.
- The formal way: A graph G is a triplet consisting of:

– A vertex set $V(G)$ (non empty)
– An edge set $E(G)$, disjoint from the vertex set (could be empty)
– A relation between an edge and a pair of vertices General notation: $|V(G)| = n, |E(G)| = m$. Please adhere to this xD.

A bit of imagery. Note, "we" define graphs such that the vertex set can never be empty. Also, for the scope of the course, our graph shall be finite.

- Loop: An edge whose endpoints are equal
- Multiple edges: Edges having same pair of endpoints
- Simple graph: No loop or multiple edges.





- Finite Graph: a graph whose vertex set and edge set are finite
- Null Graph: a graph whose vertex set and edge set are empty
- Adjacent vertices: v_j and v_k are adjacent if edge e_i is incident upon both v_j and v_i.
- Degree: of a vertex v_k is the number of edges incident upon it. Denoted using $d(v_k)$

Proposition 1:

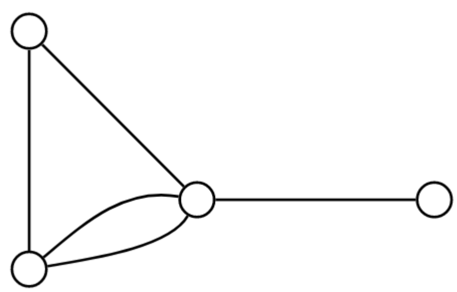
Let G be a graph. Then, the sum of degrees of the vertices is twice the number of edges, i.e.

$$\sum d(v) = 2|E(G)|, v \in V(G)$$

Proof: Each edge contributes 2 degrees (to the vertices of G). To apply it on any graph, we declare that a loop contributes 2 degrees to the vertex.

Adjacency Matrix

- Let $G = (V, E), |V| = n$ and $|E| = m$
- The adjacency matrix of G written $A(G)$, is the $n \times n$ in which entry $a_{i,j}$ is the number of edges in G with endpoints $\{v_i, v_j\}$.



$$\begin{array}{c c} & \begin{array}{cccc} w & x & y & z \end{array} \\ \begin{array}{c} w \\ x \\ y \\ z \end{array} & \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 2 & 0 \\ 1 & 2 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \end{array}$$

G a simple graph then, adjacency matrix $A(G)$ is symmetric $(0, 1)$ matrix. An **important result** for real symmetric matrices, they are orthogonally diagonalizable over $\mathbb{R}$.

Incidence Matrix

Let $G = (V, E), |V| = n$ and $|E| = m$. The incidence matrix $M(G)$ is $n \times m$ matrix in which entry $m_{i,j}$ is 1 if v_i is an endpoint of e_i and otherwise 0.

$$\begin{array}{c c} & \begin{array}{ccccc} a & b & c & d & e \end{array} \\ \begin{array}{c} w \\ x \\ y \\ z \end{array} & \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \end{array}$$

L(2)

Complement (Simple Graph)

Complement of G: The complement G' of a simple graph G:

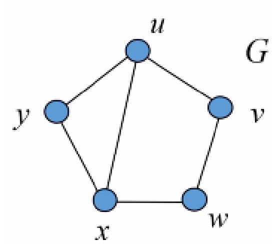
- $V(G') = V(G)$
- $E(G') = \{uv | uv \notin E(G)\}$: Every edge is determined by it's endpoints (u, v).

Subgraph

- $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$ and,
- The assignment of endpoints to edges in H is the same as in G.
- Induced subgraph: Whenever the vertices of $H(G)$ have **all** the edges in $E(G)$ corresponding to all vertices $H(G)$.

Clique and Independent Set

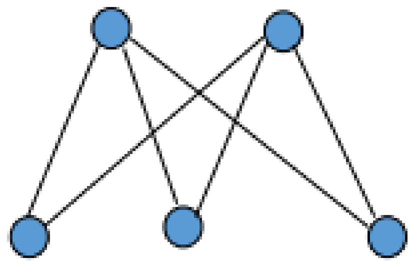
- Complete Graph: A simple graph whose vertices are pairwise adjacent.
- We use the notation K_n to denote a complete graph of n vertices.
  - $\binom{n}{2}$ edges
  - Complement of K_n is trivial (has no edges)
  - Induced subgraph: smaller or equal complete graph
- A **Clique** in a graph is a set of pairwise adjacent vertices (a complete graph). (more like a complete subgraph)
- An **Independent set** in a graph: a set of pairwise non-adjacent vertices.
- Example:
  - $\{x, y, u\}$ is a clique in G



  - $\{u, w\}, \{v, y\}$ is an independent set.
  - Question: The largest possible independent set? [Later]
  - Question: The largest clique in a graph? (Not an interesting problem though lol)

Bipartite Graphs

- A graph G is bipartite if $V(G)$ is the union of two disjoint independent sets called partite sets of G.
- Also: The vertices can be partitioned into two sets such that each set is independent
- The Matching Problem
- The Job Assignment Problem
- Complete bipartite graph (biclique) is a simple bipartite graph such that two vertices are adjacent if and only if they are in different partite sets.



A complete bipartite graph with partite sets of size r and s is denoted by $K_{r,s}$.
A graph G is k-partite if $V(G)$ is a union of k independent sets.

Path and Cycle

- Path: A sequence of distinct vertices such that two consecutive vertices are adjacent. E.g. (a,b,c,d,e)
- Cycle: A closed path (Only repeated vertex is the first which is same as last) E.g. (a,d,c,b,e,a)

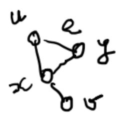
Walk and Trail

- A walk of length k is sequence of $v_0, e_1, v_1, e_2 \dots, e_k, v_k$ of vertices and edges such that $e_i = v_{i-1}v_i$ for all i.
  - A trail is a walk with NO repeated edge
  - A path is a walk with no repeated vertex
  - A U, V-walk or U, V-trail has first vertex U and last vertex V and these are the endpoints
- A walk is closed if it has length at least one and its endpoints are equal.
  - A cycle is closed trail in which "first=last" is the only repetition
  - A loop is a cycle of length one

Proposition 2

Every u, v-walk contains a u, v-path. Proof by strong induction:

- Use induction on the length of a u, v-walk W.
- Basis step: $l = 0$
  - Having no edge, W consists of a single vertex ($u = v$)
  - This vertex is a u, v-path of length 0.
- Induction step : $l \geq 1$
  - Suppose that the claim holds for walks of length less than l
  - If W has no repeated vertex, then its vertices and edges form a u, v-path. Here's an illustration

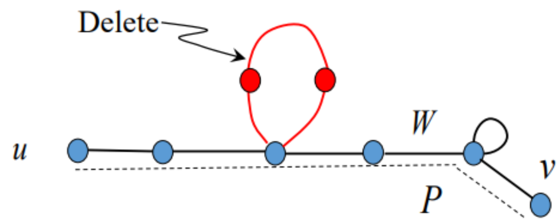


$u x y u y u x y u x v \to W$

$u x y u \; - - - - \to W'$

$\ell(W') < \ell(W) \; \ell - 2$

- If W has a repeated vertex w, then deleting the edges and vertices between appearances of w (leaving one copy of w) yields a shorter u, v-walk W' contained in W.
  - By the inductive hypothesis, W' contains a u, v-path P, and this path P is contained in W.



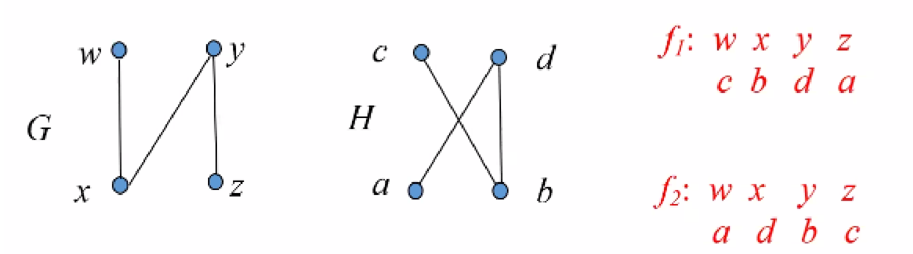
Lecture 3!

Say, "Lalisa, love me, Lalisa, love me" (Hey!)
Call me, "Lalisa, love me, Lalisa, love me" (Hey!)
Oh-ooh, aljanh-a attitude
mwol deo eojjeolagu, the loudest in the room (Hoo! Hoo!)

-- Digression

Isomorphism

An isomorphism (having the same shape/ structure preserving bijection) from a simple graph (generalizable) G to a simple graph H is a bijection $f : V(G) \to V(H)$ such that $uv \in E(G)$ if and only if $f(u)f(v) \in E(H)$



- We say "G is isomorphic to H", written as $G \equiv H$
- Isomorphism is an equivalence relation on the set of all graphs.
- How to tackle if a graph is isomorphic?

  - Use adjacency matrix representation and match (by simulataneously permuting rows and columns $R_1 \leftrightarrow R_2$ and $C_1 \leftrightarrow C_2$)
- *Section 1.1.35 has* ***special graphs*** *with small number of vertices*

Labelled and unlabelled graphs.

- We usually use letters to *label* vertices, hence the notion of labelled graph. Unlabeled graphs refer to an isomorphism class (E.g. $K_n, C_n, K_{r,s}, P_n$ (the paths with n vertices).
- No. of labelled graph given n labels is $2^{\binom{n}{2}}$ (Either choose or not choose edge).
- How many isomorphic among each other? No idea if P or NP. It is however known the subgraph isomorphism is ***NP-complete***, i.e. whether a graph contains a subgraph isomorphic to a given graph.
- Section 1.1.31 has 11 isomorphism classes of graphs with 4 vertices, whereas there are 64 labelled graphs with 4 vertices.
- Exercise: $G \equiv H$ if and only if $G' \equiv H'$

Decomposition of graphs

- A graph is **self-complementary** if it is isomorphic to it's complement.
- A **decomposition** of a graph G is a list of subgraphs of G such that each edge occurs in precisely one of the subgraphs in the list. (A vertex can appear in one or more of the subgraphs, but need not occur in all)

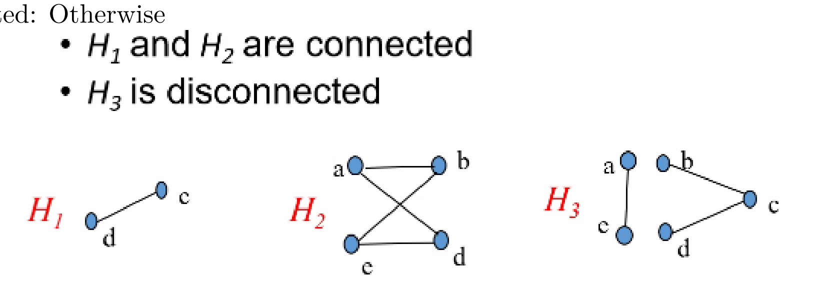
Automorphism

An automorphism of G is a permutation of $V(G)$ that is an isomorphism from G to G.

- A graph is called **vertex transistive** if for every pair $u, v \in V(G)$ there is an automorphism that maps u to v.

Connected and Disconnected

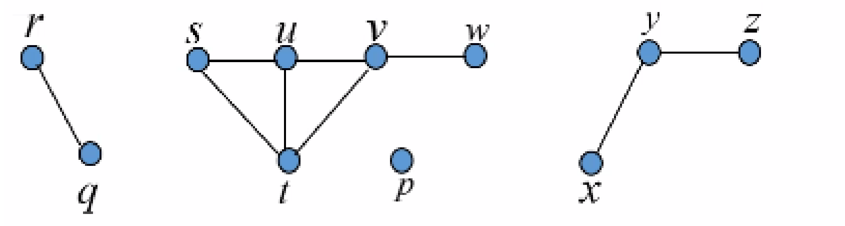
- Connected: There exists atleast one path between **any** two distinct vertices (for all pairs of vertices).
- Disconnected: Otherwise

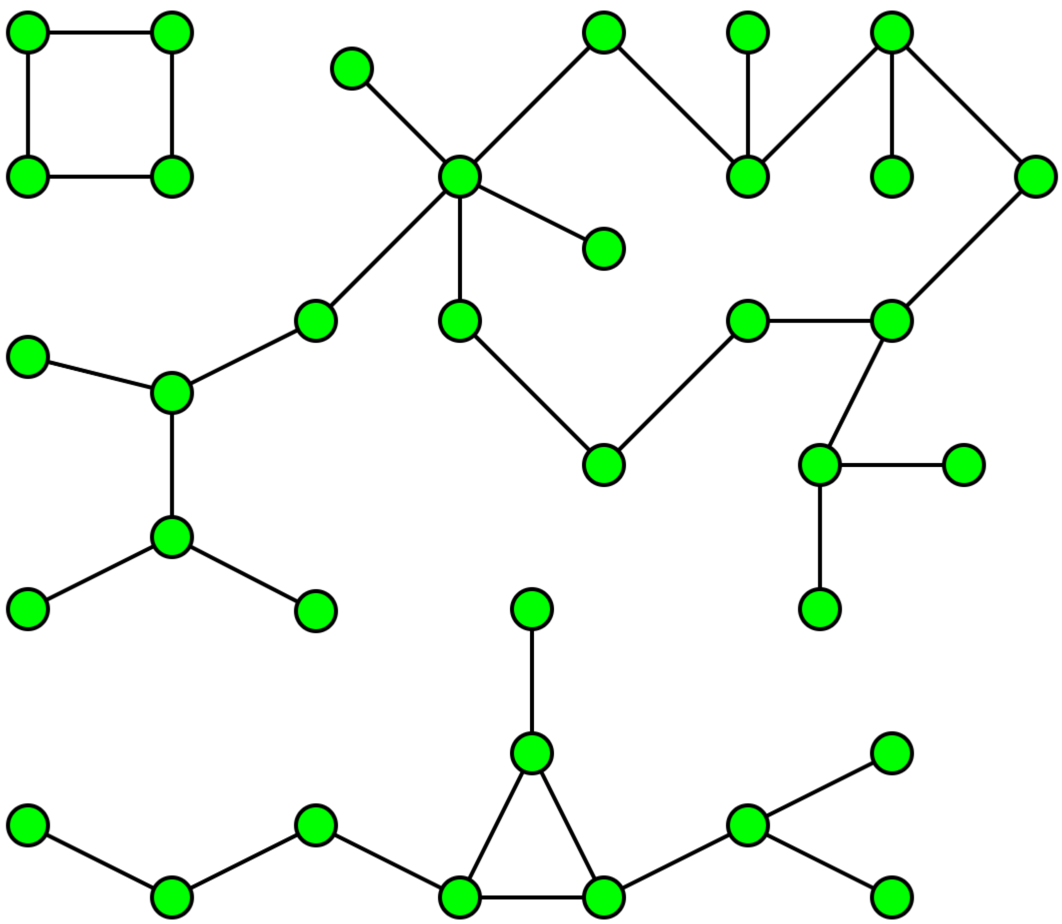


- Example:
-

Components

- The components of a graph G are its maximal connected subgraphs.
- A components is trivial if it has no edges.
- An isolated vertex is a vertex of degree 0. It is a component itself.





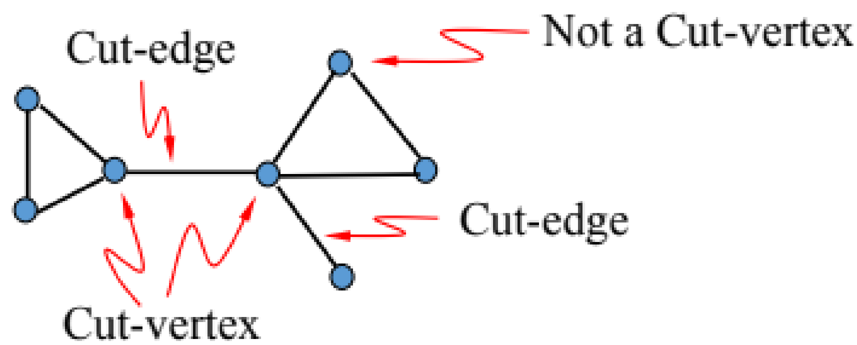
Proposition 3:

Every graph with n vertices and k edges has at least $n-k$ components. (POG)
Proof:

- An n-vertex graph with no edges has n components
- Each edge added reduces this by at most 1
- If k edges are added then the number of components is at least $n-k$

Cut-edge, Cut-vertex

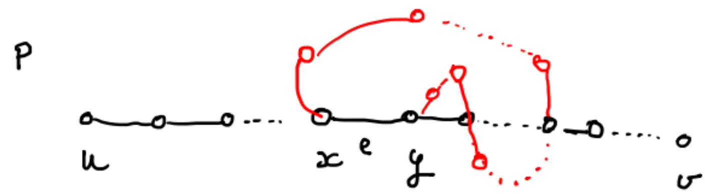
A cut-edge or cut-vertex of a graph is an edge or vertex whose deletion increases the number of components. Denote as $G-e$ or $G-v$ depending on edge or vertex of graph G to be removed.



Proposition 4:

An edge e is a cut-edge if and only if e belongs to no cycles. Proof:

- Let $e=(x,y)$ be an edge in a graph G and H be the component containing e.
  - Since deletion of e affects no other component, it suffices to prove $H-e$ is connected if and only if e belongs to a cycle. (i.e. we prove the contrapositive)
- First suppose that $H-e$ is connected.
  - This implies $H-e$ contains an x,y path
  - This path completes a cycle with e.
  - Now choose that e lies in a cycle C.
    * Choose $u,v \in V(H)$
      · Since H is connected, H has a u,v path P
    * If P does not contain e
      · Then P exists in $H-e$
    * Otherwise (P contains e)



- Suppose by symmetry that x is between u and y on P
- Now $H - e$ contains a u, x path along P, an x, y path along C, and a y, v path along P
- Putting together three paths we get a u, v walk in $H - e$
- By Proposition 2, this walk contains a path in $H - e$