GRAPH THEORY

Rui Wu
CS Department, Rutgers University
`rw761@scarletmail.rutgers.edu`

1 Graph

1.1 Basic Concepts

Graphs is a set of **Objects** and **Relations** between pairs of objects. A **Graph** $G = (V, E)$. Where: V is **Vertices/Notes**, and E is **Edges**.

If we have two nodes u and v. And between u and v, there is a edge e. We can say that e **Connects** u and v. u and v are **End Points** of e. u (v) and e are **Incident**. u and v are **Adjacent** and **Neighbors**.

And it is often convenient to consider **Directed Edges(Arcs)**. They describe **asymmetric** relations. For example, there is a flight from A to B, but not the other way around, such a graph is called **Directed**.

We should notice that if we want to describe a graph, we can write likes:
Objects:A, B, C, D, Relations:{{A, C}, {D, A}, {B, D}, {C, B}}
For directed graph:
Objects:A, B, C, D, Relations:{(A, C), (D, A), (B, D), (C, B)}

1.2 Vertex Degree

The **Degree** of a vertex is the number of its incident edges/neighbors. And the degree of a vertex v is denoted by $deg(v)$.
The **degree of a graph** is the maximum degree of its vertices. If a vertex's degree is 0, we call it as **Isolated Vertex**.
And a **Regular** graph is a graph where each vertex has the same degree(A regular graph of degree k is also called k-Regular).
The **Complement** of a graph $G = (V, E)$ is a graph $\overline{G} = (V, \overline{E})$.

1.3 Paths

A **Walk** in a graph is sequence of edges, such that each edge(except for the first one) starts with a vertex where the previous edge ended. The **Length** of a walk is the number of edges in it.
A **Path** is a walk where all edges are distinct. And a **Simple Path** is a walk where all vertices are distinct.
A **Cycle** in a graph is a path whose first vertex is the same as the last one. In particular, all the edges in a Cycle are distinct. A **Simple Cycle** is a cycle where all vertices expect for the first one are distinct(And there first vertex is taken twice).

1.4 Connectivity

A graph is called **Connected** if there is a path between every pair of its vertices. And a **Connected Component** of a graph G is a maximal connected sub-graph of G.

1.5 Directed Graphs

For directed graphs, the **In-degree** of a vertex v is the number of edges ending at v. And the **Out-degree** of a vertex v is the number of edges leaving v.

1.6 Weighted Graphs

A **Weighted Graph** associates a weight with every edge. The **Weight** of a path is the sum of weights of its edges. A **Shortest Path** between two vertices is a path of the minimum weight. And the **Distance** between two vertices is the length of a shortest path between them.

1.7 Type of Graphs

Path Graphs. The Path Graphs $P_n, n \geq 2$, consists of n vertices $v_1, ..., v_n$ and $n-1$ edges $\{v_1, v_2\}, ..., \{v_{n-1}, v_n\}$.
Cycle Graph. The Cycle Graph C_n, $n \geq 3$, consists of n vertices $v_1, ..., v_n$ and n edges $\{v_1, v_2\}, ..., \{v_{n-1}, v_n\}, \{v_n, v_1\}$.
Complete Graph. The Complete Graph(Clique) $K_n, n \geq 2$, contains n vertices $v_1, ..., v_n$ and all edges between them $(n(n-1)/2edges)$

1.8 Trees

A **tree** is a connected graph without cycles. Also, you can say a tree is a connected graph on n vertices with $n-1$ edges.

1.9 Bipartite Graphs

A graph G is **Bipartite** if its vertices can be partitioned into **two disjoint sets** L and R such that every edge in G connects a vertex in L to a vertex in R. L and R are called the **parts** of G.
And in a bipartite graph, if every vertex from part L can be connected to all vertices from part R. We call it is a **Complete Bipartite Graph** $K_{num(L),num(K)}$.
For cycle graph, if even n, C_n is always bipartite. And if odd $n \geq 3$, C_n is not bipartite. The trees are always bipartite.

2 Cycles

2.1 Handshaking Lemma

Before a business meeting, some people shook hands. Then the number of people who made an odd number of handshakes is even. In graph terms: A graph has an even number of odd nodes.

Lemma: For any graph $G(V, E)$, the sum of degrees of all its nodes is twice the number of edges:

$$\sum_{v \in V} degree(v) = 2 \cdot |E|$$

Implies the handshaking lemma: if a graph had an odd number of odd nodes, then the sum of degrees would be also odd.

Double Counting technique:
- on one hand, the number of edges is equal to

the total degree of the left part. - on the other hand, it is equal to the total degree of the right part.

2.2 Lower Bound

Theorem: An undirected graph $G(V, E)$ has at least $|V| - |E|$ connected components. If a graph is connected, then $|E| \geq |V| - 1$. The theorem is useless for graphs with $|E| \geq |V|$.

2.3 Directed Acyclic Graphs

A **directed acyclic graph**, or simply a DAG, is a directed graph without cycles.

2.4 Topological Ordering

A **topological ordering** of a directed graph is an ordering of its vertices such that, for each edge (u, v), u comes before v. And every DAG has a topological ordering.

2.5 Strongly Connected Components

In a directed graph, nodes u,v are connected, if there is a path from u to v and a path from v to u. Nodes of any directed graph can be partitioned into subsets called **strongly connected components(SCCs)**. The nodes from the same SCC are connected, and nodes from different SCCs are not connected.

2.6 Eulerian Cycle

An **Eulerian cycle(or path)** visits every edge exactly once. The definition works for both directed and undirected graphs.
A connected undirected graph contains an Eulerain cycle, if and only if the degree of every node is even. And a strongly connected directed graph contains an Eulerian cycle, if and only if, for every nodes, its in-degree is equal to its out-degree.

2.7 Hamiltonian Cycle

A **Hamiltonian cycle** visits every node of a graph exactly once.

3 Graph Classes

3.1 Spanning Trees

A **Spanning Tree** of a graph G, is a subgraph of G which is a tree and contains all vertices of G. A **Minimum Spanning Tree** of a weighted graph G is a spanning tree of the smallest weight.

Kruskal's Algorithm: Start with an empty graph T, and repeat $n - 1$ times: add to T an edge of the smallest weight which doesn't create a cycle T.

3.2 Bipartite Graphs

Theorem of Characterization: A graph is $Bipartite$ if and only if it has no cycles of odd length.

3.3 Matchings

A **Matching** in a graph is a set of edges without common vertices. And a **Maximal Matching** is a matching which cannot be extended to a large matching. A **Maximum Matching** is a matching of the largest size.

Hall's Theorem: Let $G = (V, E)$ be a graph, and SV be a subset of vertices. The **Neighborhood N(S)** of S is the set of all vertices connected to at least one vertex in S.x