SCHOOL OF SCIENCE AND HUMANITIES

DEPARTMENT OF MATHEMATICS

ADVANCED GRAPH THE

UNIT -I

GRAPHS AND DIGRAPHS

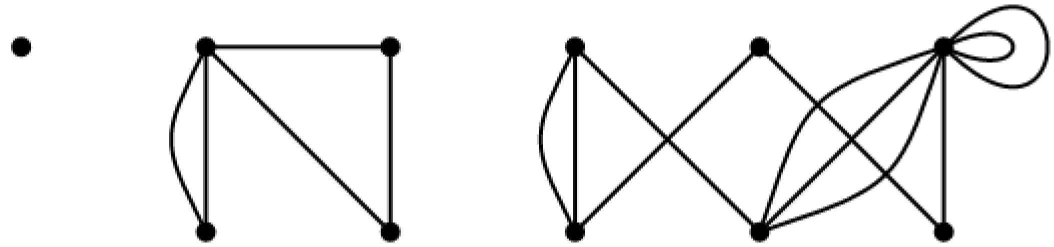
INTRODUCTION:

A graph G consists of a pair (V, E), where V is the set of vertices and E the set of edges. We write $V(G)$ for the vertices of G and $E(G)$ for the edges of G

When necessary to avoid ambiguity, as when more than one graph is under discussion.

If no two edges have the same endpoints we say there are no **multiple edges**, and if no edge has a single vertex as both endpoints we say there are no **loops**. A graph with no loops and no multiple edges is a **simple graph**. A graph with no loops, but possibly with multiple edges is a **multigraph**. The **condensation** of a multigraph is the simple graph formed by eliminating multiple edges that is, removing all but one of the edges with the same endpoints. To form the condensation of a graph, all loops are also removed. We sometimes refer to a graph as a **general graph** to emphasize that the graph may have loops or multiple edges.

DEGREE SEQUENCE: The **degree sequence** of a graph is a list of its degrees; the order does not matter, but usually we list the degrees in increasing or decreasing order. The degree sequence of the graph is



A general graph: it is not connected and has loops and multiple

listed clockwise starting at the upper left, is 0,4,2,3,2,8,2,4,3,2,2. We typically denote the degrees of the vertices of a graph by d_i, $i=1, 2,\ldots,n$, where n is the number of vertices. Depending on context, the subscript i may match the subscript on a vertex, so that d_i is the degree of v_i, or the subscript may indicate the position of d_i in an increasing or decreasing list of the degrees; for example, we may state that the degree sequence is $d_1\le d_2\le\cdots\le d_n$.

DIGRAPHS:

A **directed graph**, also called a **digraph**, is a graph in which the edges have a direction. This is usually indicated with an arrow on the edge; more formally, if v and w are vertices, an edge is an unordered pair $\{v,w\}$, while a directed edge, called an **arc**, is an ordered pair (v,w) or (w,v). The arc (v,w) is drawn as an arrow from v to w. If a graph contains both arcs (v, w) and (w,v), this is not a "multiple edge", as the arcs are distinct. It is possible to have multiple arcs, namely, an arc (v,w)

may be included multiple times in the multiset of arcs. As before, a digraph is called simple if there are no loops or multiple arcs.

We denote by E^-_v

The set of all arcs of the form (w,v), and by E^+_v the set of arcs of the form (v,w). The **indegree** of v, denoted $d^-(v)$, is the number of arcs in E^-_v, and the **outdegree**, $d^+(v)$, is the number of arcs in E^+_v. If the vertices are $v_1,v_2,\ldots,v_n$, the degrees are usually denoted $d^-_1,d^-_2,\ldots,d^-_n$ and $d^+_1,d^+_2,\ldots,d^+_n$. Note that both $\sum_{i=0}^{n} d^-_i$ and $\sum_{i=0}^{n} d^+_i$ count the number of arcs exactly once, and of course $\sum_{i=0}^{n} d^-_i=\sum_{i=0}^{n} d^+_i$. A **walk** in a digraph is a sequence $v_1,e_1,v_2,e_2,\ldots,v_{k-1},e_{k-1},v_k$ such that $e_k=(v_i,v_{i+1})$; if $v_1=v_k$, it is a closed walk or a circuit. A **path** in a digraph is a walk in which all vertices are distinct. It is not hard to show that, as for graphs, if there is a walk from v to w then there is a path from v to w.

Definition 1. A **network** is a digraph with a designated **source** s and **target** $t\ne s$. In addition, each arc e has a positive capacity, $c(e)$.

Networks can be used to model transport through a physical network, of a physical quantity like oil or electricity, or of something more abstract, like information.

Definition .2 A **flow** in a network is a function f from the arcs of the digraph to R, with $0\le f(e)\le c(e)$ for all e, and such that

$$\sum_{e \in E_v^+} f(e) = \sum_{e \in E_v^-} f(e),$$

For all v other than s and t.

Definition 3. A **cut** in a network is a set C of arcs with the property that every path from s to t uses an arc in C, that is, if the arcs in C are removed from the network there is no path from s to t. The capacity of a cut, denoted $c(C)$, is

$$\sum_{e \in C} c(e).$$

A minimum cut is one with minimum capacity. A cut C is minimal if no cut is properly contained in C.

Definition 4. If a graph G is connected, any set of vertices whose removal disconnects the graph is called a **cutset**. G has connectivity k if there is a cutset of size k but no smaller cutset. If there is no cutset and G has at least two vertices, we say G has connectivity $n-1$; if G has one vertex, its connectivity is undefined. If G is not connected, we say it has connectivity 0. G is k-connected if the connectivity of G is at least k. The connectivity of G is denoted $\kappa(G)$.

Definition 5. If a graph G is connected, any set of edges whose removal disconnects the graph is called a **cut**. G has edge connectivity k if there is a cut of size k but no smaller cut; the edge connectivity of a one-vertex graph is undefined. G is k-edge-connected if the edge connectivity of G is at least k. The edge connectivity is denoted $\lambda(G)$.

Any connected graph with at least two vertices can be disconnected by removing edges: by removing all edges incident with a single vertex the graph is disconnected. Thus, $\lambda(G) \le \delta(G)$, where $\delta(G)$ is the minimum degree of any vertex in G. Note that $\delta(G) \le n-1$, so $\lambda(G) \le n-1$

Removing a vertex also removes all of the edges incident with it, which suggests that $\kappa(G) \le \lambda(G)$. This turns out to be true, though not as easy as you might hope. We write $G-v$ to mean G with vertex v removed, and $G-\{v1,v2,\ldots,vk\}$ to mean G with all of $\{v1,v2,\ldots,vk\}$ removed, and similarly for edges.

Theorem 3. $\kappa(G) \le \lambda(G)$

Proof. We use induction on $\lambda=\lambda(G)$

If $\lambda=0$, G is disconnected, so $\kappa=0$. If $\lambda=1$, removal of edge e with endpoints v and w disconnects G. If v and w are the only vertices of G, G is $K2$ and has connectivity 1.

Otherwise, removal of one of v and w disconnects G, so $\kappa=1$As a special case we note that if $\lambda=n-1$
then $\delta=n-1$, so G is K_n and $\kappa=n-1$Now suppose $n-1>\lambda=k>1$, and removal of edges $e_1,e_2,\ldots,e_k$ disconnects G. Remove edge e_k with endpoints v and w to form G_1 with $\lambda(G_1)=k-1$. By the induction hypothesis, there are at most $k-1$ vertices v_1, $v_2,\ldots,v_j$ such that $G_2=G_1-\{v_1,v_2,\ldots,v_j\}$ is disconnected. Since $k<n-1$, $k-1\leq n-3$, and so G_2 has at least 3vertices.

If both v and w are vertices of G_2, and if adding e_k to G_2 produces a connected graph G_3, then removal of one of v and w will disconnect G_3 forming G_4, and $G_4=G-\{v_1,v_2,\ldots,v_j,v\}$ or $G_4=G-\{v_1,v_2,\ldots,v_j,w\}$, that is, removing at most k vertices disconnects G. If v and w are vertices of G_2 but adding e_k does not produce a connected graph, then removing $v_1,v_2,\ldots,v_j$ disconnects G. Finally, if at least one of v and w is not in G_2, then $G_2=G-\{v_1,v_2,\ldots,v_j\}$ and the connectivity of G is less than k. So in all cases, $\kappa\leq k$.

Theorem 4 If G has at least three vertices, the following are equivalent:

1. G is 2-connected

2. G is connected and has no cutpoint

3. For all distinct vertices u, v, w in G there is a path from u to v that does not contain w

Proof. 1⇒3 Since G is 2-connected; G with w removed is a connected graph G'. Thus, in G' there is a path from u to v, which in G is a path from u to v avoiding w

3⇒2 If G has property 3 it is clearly connected. Suppose that w is a cutpoint, so that $G'=G-w$ is disconnected. Let u and v be vertices in two different components of G', so that no path connects them in G'. Then every path joining u to v in G must use w, a contradiction.

2⇒1 Since G has at least 3 vertices and has no cutpoint, its connectivity is at least 2, so it is 2-connected by definition.

UNIT-II

Trees

Definition: A connected graph G is a **tree** if it is **acyclic**, that is, it has no cycles. More generally, an acyclic graph is called a **forest**.

Note that the definition implies that no tree has a loop or multiple edges.

Theorem 1: Every tree T is bipartite.

Proof. Since Thas no cycles, it is true that every cycle of T has even length.

Definition: A vertex of degree one is called a **pendant** vertex, and the edge incident to it is a pendant edge. □

Theorem 2: Every tree on two or more vertices has at least one pendant vertex.

Proof. We prove the contra positive. Suppose graph G has no pendant vertices. Starting at any vertex v, follow a sequence of distinct edges until a vertex repeats; this is possible because the degree of every vertex is at least two, so upon arriving at a vertex for the first time it is always possible to leave the vertex on another edge. When a vertex repeats for the first time, we have discovered a cycle. This theorem often provides the key step in an induction proof, since removing a pendant vertex (and its pendant edge) leaves a smaller tree.

Theorem 3 : A tree on n vertices has exactly $n-1$edges.

Proof. A tree on 1 vertex has 0 edges; this is the base case. If T is a tree on $n\geq 2$ vertices, it has a pendant vertex. Remove this vertex and its pendant edge to get a tree T' on $n-1$ vertices. By the induction hypothesis, T' has $n-2$ edges; thus T has $n-1$ edges. Theorem: A tree with a vertex of degree $k\geq 1$has at least k

Pendant vertices. In particular, every tree on at least two vertices has at least two pendant vertices.

Proof. The case $k=1$is obvious. Let T be a tree with n vertices, degree sequence $\{d_i\}_{i=1}^{n}$, and a vertex of degree $k\geq 2$, and let l be the number of pendant vertices. Without loss of generality, $1=d_1=d_2=\cdots=d_l$ and $d_{l+1}=k$. Then $2(n-1)=\sum_{i=1}^{n} d_i=l+k+\sum_{i=l+2}^{n} d_i\geq l+k+2(n-l-1)$.

This reduces to $l\geq k$, as desired. If T is a tree on two vertices, each of the vertices has degree 1. If T has at least three vertices it must have a vertex of degree $k\geq 2$, since otherwise $2(n-1) =\sum_{i=1}^{n} d_i=n$, which implies $n=2$. Hence it has at least $k\geq 2$ pendant vertices. Trees are quite useful in their own right, but also for the study of general graphs.

Definition: If $G$is a connected graph on n vertices, a **spanning tree** for G is a subgraph of G that is a tree on n vertices.

Theorem 4: Every connected graph has a spanning tree.

Proof. By induction on the number of edges. If $G$is connected and has zero edges, it is a single vertex, so G

is already a tree. Now suppose G has $m\geq 1$ edges. If G is a tree, it is its own spanning tree. Otherwise, G contains a cycle; remove one edge of this cycle. The resulting graph G' is still connected and has fewer edges, so it has a spanning tree; this is also a spanning tree for G. In general, spanning trees are not unique, that is, a graph may have many spanning trees. It is possible for some edges to be in every spanning tree even if there are multiple spanning trees. For example, any pendant edge must be in every spanning tree, as must any edge whose removal disconnects the graph (such an edge is called a **bridge**.)

Corollary: If G is connected, it has at least $n-1$ edges; moreover, it has exactly $n-1$edges if and only if it is a tree.

Proof. If G is connected, it has a spanning tree, which has $n-1$ edges, all of which are edges of G. If G has $n-1$ edges, which must be the edges of its spanning tree, then G is a tree.

Theorem 5: G is a tree if and only if there is a unique path between any two vertices.

Proof. Since every two vertices are connected by a path, G is connected. For a contradiction, suppose there is a cycle in G; then any two vertices on the cycle are connected by at least two distinct paths, a contradiction. Only if: If G is a tree it is connected, so between any two vertices there is at least one path. For a contradiction, suppose there are two different paths from v to w: $v=v1$, $v2,\ldots,vk=w$and $v=w1,w2,\ldots,wl=w$. Let i be the smallest integer such that $vi\neq wi$. Then let j be the smallest integer greater than or equal to i such that $wj=vm$ for some m, which must be at least i. (Since wl $=vk$, such an m must exist.) Then $vi-1,vi,\ldots,vm=wj,wj-1,\ldots,wi-1=vi-1$ is a cycle in G, a contradiction. Definition: A **cutpoint** in a connected graph G is a vertex whose removal disconnects the graph.

Theorem: Every connected graph has a vertex that is not a cutpoint.

Proof. Remove a pendant vertex in a spanning tree for the graph.

Theorem 6. **(Menger's Theorem)** If G has at least k+1 vertex, then G is k-connected if and only if between every two vertices u and v there are k pairwise internally disjoint paths.

Proof of Menger's Theorem.
Suppose first that between every two vertices v and w in G there are k internally disjoint paths. If G is not k-connected, the connectivity of G is at most $k-1$, and because G has at least $k+1$ vertices, there is a cutset S of G with size at most $k-1$. Let v and w be vertices in two different components of $G-S$; in G these vertices are joined by k internally disjoint paths. Since there is no path from v to w in $G-S$, each of these k paths contains a vertex of S, but this is impossible since S has size less than k, and the paths share no vertices other than v and w. This contradiction shows that G is k connected. Now suppose G is k connected.

If v and w are not adjacent, $\kappa\ G(v,w)\geq k$ and by the previous theorem there are $p\ G(v,w)=\kappa\ G(v,w)$ internally disjoint paths between v and w. If v and w are connected by edge e, consider $G-e$. If there is a cutset of $G-e$ of size less than $k-1$, call it S, then either $S\cup\{v\}$ or $S\cup\{w\}$ is a cutset of G of size less than k, a contradiction. (Since G has at least $k+1$ vertices, $G-S$ has at least three vertices.) Thus, $\kappa G-e(v,w)\geq k-1$ and by the previous theorem there are at least $k-1$ internally disjoint paths between v and w in $G-e$. Together with the path v, w using edge e, these form k internally disjoint paths between v and w in G

Definition A **block** in a graph G is a maximal induced subgraph on at least two vertices without a cutpoint.

Theorem 6: If G is connected but not 2-connected, then every vertex that is in two blocks is a cutpoint of G

Proof. Suppose w is in $B1$ and $B2$, but $G-w$ is connected. Then there is a path $v1,v2,\ldots,vk$ in $G-w$, with $v1\in B1$ and $vk\in B2$. But then $G[V(B1)\cup V(B2)\cup\{v1,v2,\ldots,vk\}]$ is 2-connected and contains both $B1$ and $B2$, a contradiction.

UNIT-III

Euler and Hamiltonian Graphs

Definition: A walk in a graph is a sequence of vertices and edges, $v1,e1,v2,e2,\ldots,vk,ek,vk+1$ such that the endpoints of edge ei are vi and $vi+1$. In general, the edges and vertices may appear in the sequence more than once. If $v1=vk+1$, the walk is a **closed walk** or a **circuit**. A successful walk in Königsberg corresponds to a closed walk in the graph in which every edge is used exactly once. What can we say about this walk in the graph, or indeed a closed walk in any graph that uses every edge exactly once? Such a walk is called an **Euler circuit**. If there are no vertices of degree 0, the graph must be connected, as this one is. Beyond that, imagine tracing out the vertices and edges of the walk on the graph. At every vertex other than the common starting and ending point, we come into the vertex along one edge and go out along another; this can happen more than once, but since we cannot use edges more than once, the number of edges incident at such a vertex must be even. Already we see that we're in trouble in this particular graph, but let's continue the analysis. The common starting and ending point may be visited more than once; except for the very first time we leave the starting vertex, and the last time we arrive at the vertex, each such visit uses exactly two edges. Together with the edges used first and last, this means that the starting vertex must also have even degree. Thus, since the Königsberg Bridges graph has odd degrees, the desired walk does not exist. The question that should immediately spring to mind is this: if a graph is connected and the degree of every vertex is even, is there an Euler circuit? The answer is yes.

Theorem If G is a connected graph, then G contains an Euler circuit if and only if every vertex has even degree.

Proof. We have already shown that if there is an Euler circuit, all degrees are even. We prove the other direction by induction on the number of edges. If G has no edges the problem is trivial, so we assume that G has edges. We start by finding some closed walk that does not use any edge more than once: Start at any vertex $v0$ follow any edge from this vertex, and continue to do this at each new vertex, that is, upon reaching a vertex, choose some unused edge leading to another vertex. Since every vertex has even degree, it is always possible to leave a vertex at which we arrive, until we return to the starting vertex, and every edge incident with the starting vertex has been used. The sequence of vertices and edges formed in this way is a closed walk; if it uses every edge, we are done. Otherwise, form graph G' by removing all the edges of the walk. G' is not connected, since vertex $v0$ is not incident with any remaining edge. The rest of the graph, that is, G' without $v0$, may or may not be connected. It consists of one or more connected subgraphs, each with fewer edges than G; call these graphs $G1, G2,\ldots,Gk$. Note that when we remove the edges of the initial walk, we reduce the degree of every vertex by an even number, so all the vertices of each graph Gi have even degree. By the induction hypothesis, each Gi has an Euler circuit. These closed walks together with the original closed walk use every edge of Gexactly once. Suppose the original closed walk is $v_0,v_1,\ldots,v_m=v_0$, abbreviated to leave out the edges. Because G is connected, at least one vertex in each Gi appears in this sequence, say vertices $w_1,1\in G_1$, $w_2,_1\in G_2,\ldots,wk,Gk$, listed in the order they appear in $v0,v1,\ldots,vm$. The Euler circuits of the graphs Gi are $w1,1,w1,2,\ldots,w1,m1=w1,1w2,1,w2,2,\ldots,w2,m2=w2,wk,1,wk,2,\ldots,wk,mk=wk,1$.By pasting together the original closed walk with these, we form a closed walk in G that uses every edge exactly once: $v0,v1,\ldots,vi1=w1,1,w1,2,\ldots,w1,m1=vi1,vi1+1,\ldots,vi2=w2,1,\ldots,w2,m2=vi2,vi2+1,\ldots,vik=wk,1,\ldots,wk,mk=vik,vik+1,\ldots,vm=v0$.Now let's turn to the second interpretation of the problem: is it possible to walk over all the bridges exactly once, if the starting and ending points need not be the same? In a graph G, a walk that uses all of the edges but is not an Euler circuit is called an **Euler walk**. It is not too difficult to do an analysis much like the one for Euler circuits, but it is even easier to use the Euler circuit result itself to characterize Euler walks.

Theorem 1 A connected graph G has an Euler walk if and only if exactly two vertices have odd degree.

Proof. Suppose first that G has an Euler walk starting at vertex v and ending at vertex w. Add a new edge to the graph with endpoints v and w, forming G'. G' has an Euler circuit, and so by the previous theorem every vertex has even degree. The degrees of v and w in G are therefore odd, while all others are even. Now suppose that the degrees of v

and w in G are odd, while all other vertices have even degree. Add a new edge e to the graph with endpoints v and w, forming G'. Every vertex in G' has even degree, so by the previous theorem there is an Euler circuit which we can write as $v,e1,v2,e2,\ldots,w,e,v,$so that $v,e1,v2,e2,\ldots,w$ is an Euler walk.

Theorem 2.

A graph is eulerian if and only if it is connected and every vertex has even degree clearly, an eulerian graph must be connected. Also, if is an eulerian circuit in , then for each , we can view the edge as exiting and entering . The degree of every vertex must be even, since for each vertex, the number of edges exiting equals the number of edges entering . Furthermore, each edge incident with either exits from or enters .We now describe a deterministic process that will either (a) find an eulerian circuit, (b) show that the graph is disconnected, or (c) find a vertex of odd degree. The description is simplified by assuming that the vertices in have been labelled with the positive integers, where is the number of vertices in . Furthermore, we take .We launch our algorithm with a trivial circuit consisting of the vertex. Thereafter suppose that we have a partial circuit defined by with. The edges of the form have been traversed; while the remaining edges in (if any) have not. If the third condition for an Euler circuit is satisfied, we are done, so we assume it does not hold.We then choose the least integer for which there is an edge incident with that has not already been traversed. If there is no such integer, since there are edges that have not yet been traversed, then we have discovered that the graph is disconnected. So we may assume that the integer exists. Set. We define a sequence recursively. If, set is an edge in and has not yet been traversed. If , we take as the least positive integer in . If , then and we take and halt this subroutine. When the subroutine halts, we consider two cases. If, then and are vertices of odd degree in . So we are left to consider the case where . In this case, we simply expand our original sequence by replacing the integer by the sequence.

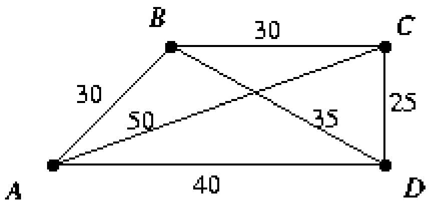
Theorem 3. A connected multigraph has an Euler path but not an Euler circuit if and only if it has exactly two vertices of odd degree.

Proof: (ONLY IF) Assume the graph has an Euler path but not a circuit. Notice that every time the path passes through a vertex, it contributes 2 to the degree of the vertex (1 when it enters, 1 when it leaves). Obviously the first and the last vertices will have odd degree and all the other vertices - even degree. (IF) Assume exactly two vertices, u and v, have odd degree. If we connect these two vertices, then every vertex will have even degree. By Theorem 1, there is an Euler circuit in such a graph. If we remove the added edge {u,v} from this circuit, we will get an Euler path for the original graph. Hence the proof.

Results: Suppose a connected graph has degree sequence *d*1, *d2*,…, *dn*. How many edges must be added to *G?*

so that the resulting graph has an Euler circuit? Explain.

Travelling salesman problem Suppose the distances between each pair of the cities *A*, *B*, *C* and *D* are given, and suppose a salesman must travel to each city exactly once, starting and ending at city *A*. Which route from city to city will minimize the total travelling distance? If we use vertices to denote cities, and put the distance between any two cities on the edge joining them, then we can represent the given knowledge by the following **weighted graph**



Before considering the above two problems in detail, we first introduce below a few related basic terminology.

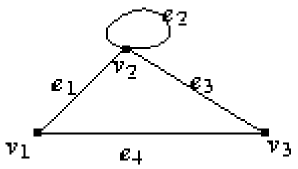
Eulerian and Hamiltonian Paths and Circuits

- A **circuit** is a walk that starts and ends at a same vertex, and contains no repeated edges.
- An **Eulerian circuit** in a graph *G* is a circuit that includes all vertices and edges of *G*. A graph which has an Eulerian circuit is an **Eulerian graph**.
- A **Hamiltonian circuit** in a graph *G* is a circuit that includes every vertex (except first/last vertex) of *G* exactly once.

- An **Eulerian path** in a graph *G* is a walk from one vertex to another, that passes through all vertices of *G* and traverses exactly once every edge of *G*. An Eulerian path is therefore not a circuit.
- A **Hamiltonian path** in a graph *G* is a walk that includes every vertex of *G* exactly once. A Hamiltonian path is therefore not a circuit.

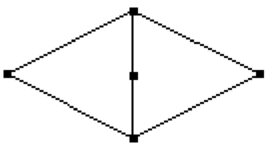
Examples

1. In the following graph



(a)
Walk $v_1e_1v_2e_3v_3e_4v_1$, loop $v_2e_2v_2$ and vertex v_3 are all circuits, but vertex v_3 is a trivial circuit.
(b)
$v_1e_1v_2e_2v_2e_3v_3e_4v_1$ is an Eulerian circuit but not a Hamiltonian circuit.
(c)
$v_1e_1v_2e_3v_3e_4v_1$ is a Hamiltonian circuit, but not an Eulerian circuit.

2. K_3 is an Eulerian graph, K_4 is not Eulerian.
3. Graph



has an Eulerian path but is not Eulerian.

Euler's Theorem Let *G* be a connected graph.

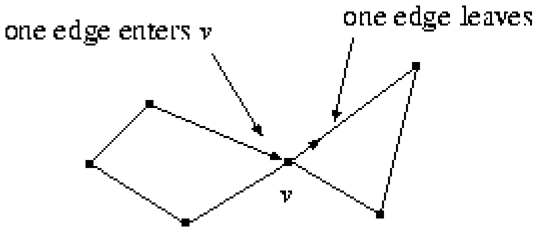
(i)
G is Eulerian, i.e. has an Eulerian circuit, if and only if every vertex of *G* has even degree.
(ii)
G has an Eulerian path, but not an Eulerian circuit, if and only if *G* has exactly two vertices of odd degree. The Eulerian path in this case must start at any of the two odd-degree vertices and finish at the other vertex.

Proof We only consider the case (i).

(a)

We first show G is Eulerian implies all vertices have even degree.

Let C be an Eulerian (circuit) path of G and v an arbitrary vertex. Then each edge in C that enters v must be followed by an edge in C that leaves v. Thus the total number of edges incident at v must be even.



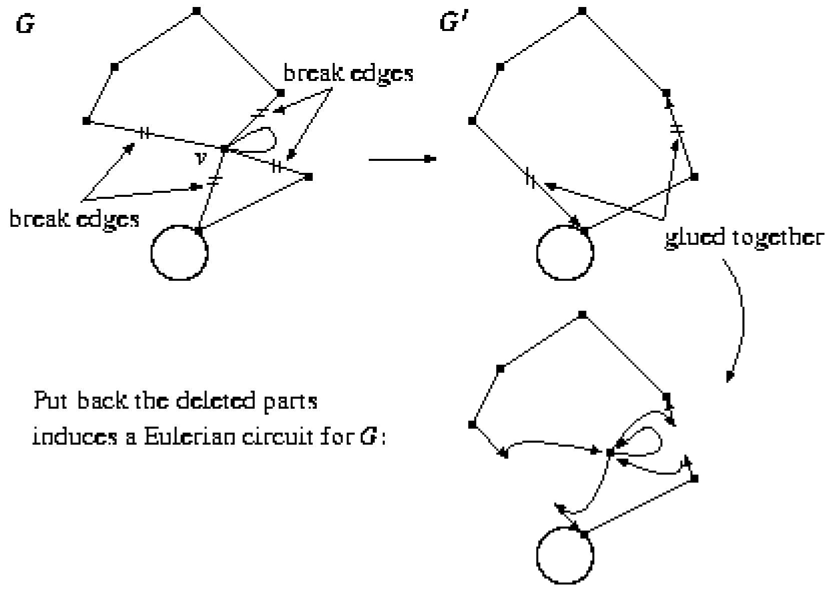
(b)

We then show by induction that G is Eulerian if all of its vertices are of even degree.

Let S_n be the statement that connected graph of n vertices must be Eulerian if its every vertex has even degree.

For $n=1$, G is either a single vertex or a single vertex with loops. Hence S_1 is true because an Eulerian circuit can be obtained by traversing all loops (if any) one by one.

For inductions we now assume S_k is true, and G has $k+1$ vertices. Select a vertex v of G. We form a subgraph G' with one vertex less as follows: remove all loops of v and break all remaining edges incident at v; remove v and connect in pairs the broken edges in such a way G remains connected. Since the degrees of the vertices remain even when G is reduced to G', the induction assumption implies the existence of an Eulerian circuit of G'. The Eulerian circuit of G can thus be constructed by traversing all loops (if any) at v and then the Eulerian circuit of G' starting and finishing at v. Hence G is Eulerian and S_{k+1} is true, implying S_n is true for all $n \geq 1$. For clarity and intuitiveness, the induction step is exemplified by the following graphs



Examples

4. Due to the above Euler's theorem, the seven bridge problem described earlier has no solution, i.e. the graph in the problem does not have an Eulerian circuit because all vertices there *(A, B, C, D)* have odd degrees.
5. As for the travelling salesman problem, we need to find all the Hamiltonian circuits for the graph, calculate the respective total distance and then choose the shortest route.

route	total distance
$A \to B \to C \to D \to A$	$30 + 30 + 25 + 40 = 125$
$A \to B \to D \to C \to A$	140
$A \to C \to B \to D \to A$	155
$A \to C \to D \to B \to A$	140
$A \to D \to B \to C \to A$	155
$A \to D \to C \to B \to A$	125

6. Hence the best route is either *ABCDA* or *ADCBA*.

Fleury's Algorithm for Finding Eulerian Path or Circuit

(i)

If there are odd degree vertices (there then must be exactly two if an Eulerian path is to exist), choose one. Travel over any edge whose removal will not result in breaking the graph into disconnected components.

(ii)

Rub out the edge (or colour the edge if you like) you have just traversed, and then travel over any remaining edge whose removal will not result in breaking the remaining subgraph into disconnected components.

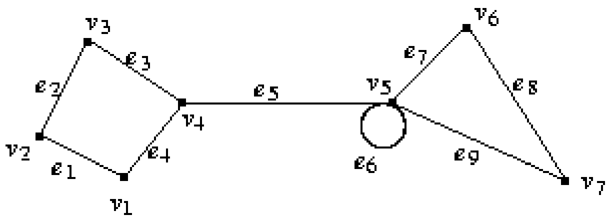
(iii)

Repeat (ii) until other edges are rubbed out or coloured.

You may consult for further details, if you wish the book by John E Munro, *Discrete Mathematics for Computing*, Thomas Nelson, 1992.

Example

Find an Eulerian path for the graph *G* below

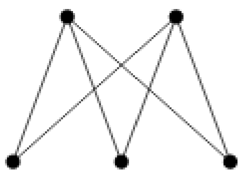


We start at v_5 because $\delta(v_5) = 5$ is odd. We can't choose edge e_5 to travel next because the removal of e_5 breaks *G* into 2 connected parts. However we can choose e_6 or e_7 or e_9. We choose e_6. *One* Eulerian path is thus .

UNIT-IV

Planarity and Connectivity

Whether is it possible to draw a graph so that none of the edges cross? If this *is* possible, we say the graph is *planar* (since you can draw it on the *plane*).Notice that the definition of planar includes the phrase "it is possible to." This means that even if a graph does not look like it is planar, it still might be. Perhaps you can redraw it in a way in which no edges cross. For example, this is a planar graph:



The graphs are the same, so if one is planar, the other must be too. However, the original drawing of the graph was not a *planar representation* of the graph.

When a planar graph is drawn without edges crossing, the edges and vertices of the graph divide the plane into regions. We will call each region a *face*. The graph above has 3 faces (yes, we *do* include the "outside" region as a face). The number of faces does not change no matter how you draw the graph (as long as you do so without the edges crossing), so it makes sense to ascribe the number of faces as a property of the planar graph.
WARNING: you can only count faces when the graph is drawn in a planar way. For example, consider these two representations of the same graph:

Euler's Formula for Planar Graphs. For any connected planar graph with v vertices, e edges and faces, we have
$v-e+f=2$.Not all graphs are planar. If there are too many edges and too few vertices, then some of the edges will need to intersect. The smallest graph where this happens is K5.

Proof. The proof is by contradiction. So assume that K5is planar. Then the graph must satisfy Euler's formula for planar graphs. K5 has 5 vertices and 10 edges, so we get$5-10+f=2$, Which says that if the graph is drawn without any edges crossing, there would be $f=7$faces. Now consider how many edges surround each face. Each face must be surrounded by at least 3 edges. Let B be the total number of boundaries around all the faces in the graph. Thus we have that $3f\leq B$. But also $B=2e$, since each edge is used as a boundary exactly twice. Putting this together we get$3f\leq 2e$.But this is impossible, since we have already determined that $f=7$ and $e=10$, and $21\not\leq 20$. This is a contradiction so in fact K5 is not planar.

Graphs formed from maps in this way have an important property: they are **planar**.

Definition - A graph *G* is planar if it can be represented by a drawing in the plane so that no edges cross. The number of colors needed to properly color any map is now the number of colors needed to color any planar graph. This problem was first posed in the nineteenth century, and it was quickly conjectured that in all cases four colors suffice. This was finally proved in 1976 with the aid of a computer. In 1879, Alfred Kempe gave a proof that was widely known, but was incorrect, though it was not until 1890 that this was noticed by Percy Heawood, who modified the proof to show that five colors suffice to color any planar graph. We will prove this Five Color Theorem, but first we need some other results. We assume all graphs are simple. Theorem (Euler's Formula) supposes G is a connected planar graph, drawn so that no edges cross, with n vertices and m edges, and that the graph divides the plane into r regions. Then $r=m-n+2$.

Proof. The proof is by induction on the number of edges. The base case is $m=n-1$, the minimum number of edges in a connected graph on n vertices. In this case G is a tree, and contains no cycles, so the number of regions is 1, and indeed $1=(n-1)-n+2$Now suppose G has more than $n-1$ edges, so it has a cycle. Remove one edge from a cycle forming G', which is connected and has $r-1$ regions, n vertices, and $m-1$ edges. By the induction hypothesis $r-1=(m-1)-n+2$, which becomes $r=m-n+2$ when we add 1 to each side.

Lemma: Suppose G is a simple connected planar graph, drawn so that no edges cross, with $n\geq 3$ vertices and m edges, and that the graph divides the plane into r regions. Then $m\leq 3n-6$

Proof. Let fi be the number of edges that adjoin region number i; if the same region is on both sides of an edge, that edge is counted twice. We call the edges adjoining a region the boundary edges of the region. Since G is simple and $n\geq 3$, every region is bounded by at least 3 edges. Then $\sum ri=1fi=2m$, since each edge is counted twice, once for the region on each side of the edge. From $r=m-n+2$ we get $3r=3m-3n+6$, and because $fi\geq 3$, $3r\leq\sum ri=1fi=2m$, so $3m-3n+6\leq 2m$, or $m\leq 3n-6$ as desired.

Theorem 5. K_5 is not planar.

Proof. K_5 has 5 vertices and 10 edges, and $10\nleq 3\cdot 5-6$, so by the lemma, $K5$ is not planar.

Lemma: If G is planar then G has a vertex of degree at most 5.

Proof. We may assume that G is connected (if not, work with a connected component of G). Suppose that $d(vi)>5$ for all vi. Then $2m=\sum ni=1\ d(vi)\geq 6n$. By lemma $3n-6\geq m$ so $6n-12\geq 2m$. Thus $6n\leq 2m\leq 6n-12$, a contradiction.

Theorem :(Five Color Theorem) Every planar graph can be colored with 5 colors.

Proof. The proof is by induction on the number of vertices n; when $n\leq 5$this is trivial. Now suppose G is planar on more than 5 vertices; by lemma some vertex v has degree at most 5. By the induction hypothesis, $G-v$ can be colored with 5 colors. Color the vertices of G, other than v, as they are colored in a 5-coloring of $G-v$. If $d(v)\leq 4$, then v can be colored with one of the 5 colors to give a proper coloring of G with 5 colors. So we now suppose $d(v)=5$. If the five neighbours of v are colored with four or fewer of the colors, then again v can be colored to give a proper coloring of G with 5 colors. Now we suppose that all five neighbours of v have a different color five neighbours of v colored with 5 colors: v_1 is red, v_2 is purple, v_3 is green, v_4 is blue, $v_5$is orange. Suppose that in G there is a path from $v1$ to $v3$, and that the vertices along this path are alternately colored red and green; call such a path a red-green alternating path. Then together with v, this path makes a cycle with $v2$ on the inside and $v4$ on the outside, or vice versa. This means there cannot be a purple-blue alternating path from $v2$ to $v4$. Supposing that $v2$ is inside the cycle, we change the colors of all vertices inside the cycle colored purple to blue, and all blue vertices are recolored purple. This is still a proper coloring of all vertices of G except v, and now no neighbor of v is purple, so by coloring v purple we obtain a proper coloring of GIf there is no red-green alternating path from $v1$to $v3$, then we recolor vertices as follows: Change the color of $v1$ to green. Change all green neighbors of $v1$ to red. Continue to change the colors of vertices from red to green or green to red until there are no conflicts, that is, until a new proper coloring is obtained. Because there is no red-green alternating path from $v1$ to $v3$, the color of $v3$ will not change. Now no neighbor of v is colored red, so by coloring v red we obtain a proper coloring of G.

Example: If G is Kn, $PG(k)=k(k-1)(k-2)$ $(k-n+1)$, namely, the number of permutations of k things taken n at a time. Vertex 1 may be colored any of the k colors, vertex 2 any of the remaining $k-1$ colors, and so on. Note that when $k<n$, $PG(k)=0$

Example: If G has n vertices and no edges, $PG(k)=kn$.

Given PG it is not hard to compute $\chi(G)$; for example, we could simply plug in the numbers 1,2,3,… for k until $PG(k)$ is non-zero. This suggests it will be difficult (that is, time consuming) to compute PG. We can provide an easy mechanical procedure for the computation, quite similar to the algorithm we presented for computing $\chi(G)$.Suppose G has edge $e=\{v,w\}$, and consider $PG-e(k)$, the number of ways to color $G-e$ with k colors. Some of the colorings of $G-e$ are also colorings of G, but some are not, namely, those in which v and w have the same color. How many of these are there? From our discussion of the algorithm for $\chi(G)$ we know this is the number of colorings of G/e. Thus, $PG(k)=PG-e(k)-PG/e(k)$.Since $G-e$ and G/e both have fewer edges than G, we can compute PG by applying this formula recursively. Ultimately, we need only compute PG f or graphs with no edges, which is easy. Since $PG(k) = kn$ when G has no edges, it is then easy to see, and to prove by induction, that PG is a polynomial.

Theorem: For all G on n vertices, PG is a polynomial of degree n, and PG is called the **chromatic polynomial** of G

Proof. The proof is by induction on the number of edges in G When G has no edges, otherwise, by the induction hypothesis, $PG-e$

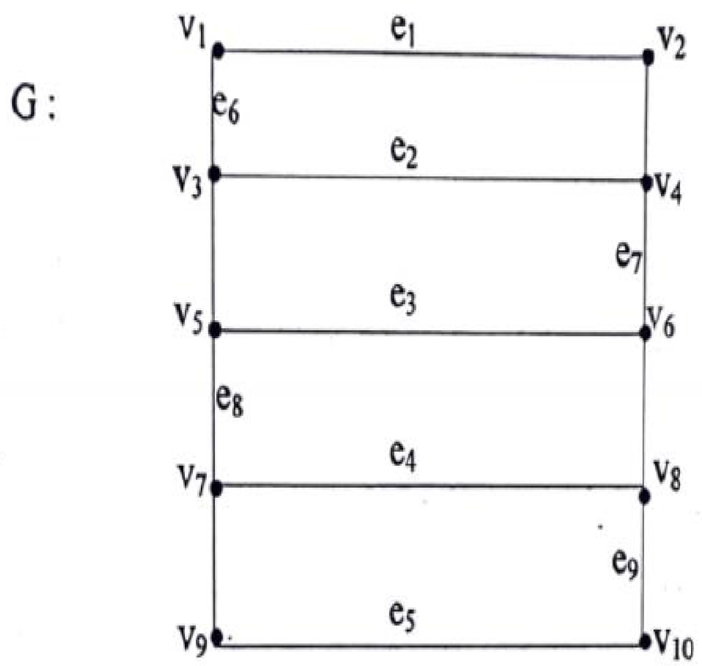
is a polynomial of degree n and PG/e is a polynomial of degree $n-1$, so $PG=PG-e-PG/e$ is a polynomial of degree n.

The chromatic polynomial of a graph has a number of interesting and useful properties.

UNIT-V Matching and Covering

Definition: A subset M of E is called a matching in G if no two of the edges in M are adjacent. The two ends of an edge in M are said to be matched under M.

Example: In the graph G of figure the sets $M_1 = \{e_6, e_8\}$,



$M_3 = \{e_6, e_7, e_8, e_9\}$ and $M_3 = \{e_1, e_2, e_3, e_4, e_5\}$ are all matchings.

Definition: A matching M saturates a vèrtex v if one edge of M is incident with v. Also, we say v is M-saturated. Otherwise, v is M-unsaturated.

Example: In the graph G of figure , vi is both M_1-saturated and M_2-saturated; v_4 is M_2-saturated but M_1-unsaturated; but M_3 saturates every vertex of G.

Definition: If M is a matching in G such that every vertex of G is M-saturated then M is called a perfect matching.

Example: The matching M, of G of figure is a perfect matching where as M_1 and M_2 are not perfect.

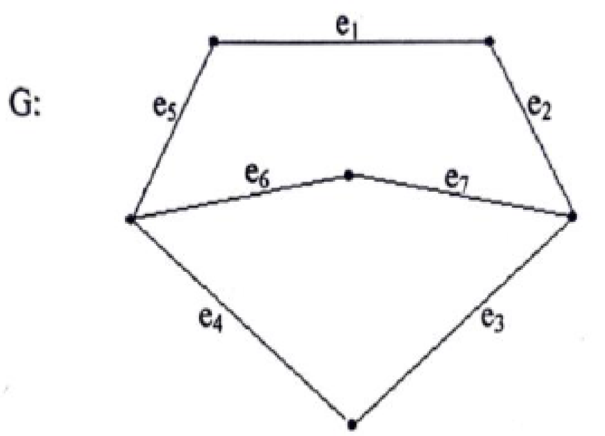
Note: If G has a perfect matching, then p is even.

Definition: A matching M is called a maximal matching of G if there is no matching M' of G such that $M' \supset M$.

Remark: Note that two maximal matchings need not have same Cardinality.

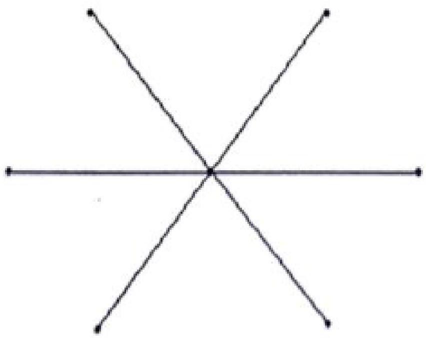
Example: In the graph G , $M_1 = (e_1, e_6, e_3\}$ and $M_2=\{e_5, e_3\}$ are maximal matchings.

Definition: A matching M of G is called a Maximum matching if G has no matching M' with $|M'| > |M|$. The number of edges in a maximum matching of G is called as the matching number of G.



We note that $M_1 = \{e_1, e_6, e_3\}$ is a maximum matching of G, but $M_2 = \{e_5, e_3\}$ is not a maximum matching, though it is a maximal matching of G. Clearly every perfect matching is maximum; but maximum matchings need not be perfect.

Example: Consider the star $K_{1,6}$ and in general $K_{1,p}$. Here any maximum matching contains only one edge and hence it is not perfect.



$K_{1,6}$

Definition: Let M be the matching in G. An M-alternating path in G is a path whose edges are alternately in E/M and M.

Example: In the graph G, if we consider the matching M= $\{e_1, e_2\}$ then the path $v_1v_2v_6v_5v_3$ is an M- alternating path.

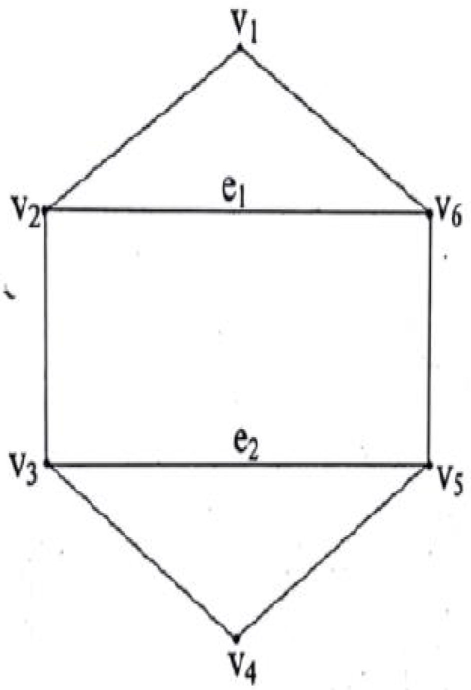
Definition: Let M be a matching in G. An M-augmenting path is an M-alternating path whose origin and terminus are M-unsaturated.

Example: In the graph G, if we consider the matching M=$\{e_1 , e_2\}$ then the path $v_1v_2v_6v_5v_3$ is an M-augmenting path.

Note: 1. In M-augmenting path initial and final edges are in E\M.

2. An M- alternating path whose initial and final edges are in E\M, need not be an M-augmenting path.

G:



A matching M in G is a maximum matching if and only if G contains no M-augmenting path.

Proof: Let M be a maximum matching in G. We prove that G has no M-augmenting path. Suppose not, let G have a M-augmenting path, $v_0e_1v_1e_2v_2 \ldots v_{2m}\, e_{2m+1}v_{2m+1}$. We note that such a path is of odd length. Now we define set M' ⊆E by,

$M' = \{M - \{e_2, e_4, \ldots, e_{2m}\}\} \cup \{ e_1, e_3, \ldots, e_{2m+1} \}$.

Then M' is a matching in G and $|M'| = |M|+1$. This is a contradiction to the fact that M is maximum matching. Hence, G has no M-augmenting path.

Conversely, let G has no M-augmenting path. We prove that M is a maximum matching in G. Suppose not, let M' be a maximum matching in G.

Then, $|M'| > |M|$ (1)

Let H G[M Δ M'] where MΔM' denotes the symmetric difference of M and M'. Each vertex of H has degree either one or two in H, since it can be incident with at most one edge of M and one edge of M'. Thus each component of H is either an even cycle with edges alternately in M and M' or else a path with edges alternately in M and M'.

By (1), H contains more edges of M' than of M and so some path component P of H must contain more edges of M' than M and therefore must start and end with edges of M'. The origin and terminus of P being M' -saturated in H and of degree one, are M-unsaturated in G. Therefore, P is an M-augmenting path in G, which is a contradiction to our assumption. Hence, M is a maximum matching in G.

SYSTEM OF DISTINCT REPRESENTATIVES AND MARRIAGE PROBLEM

Let X be a non-empty finite set and S = $\{S_1,S_2,..,S_m\}$ be a family of (not necessarily distinct) non empty subset of X. If there exists a set $\{X_1, X_2, X_m\}$ of X such that x_i € S_i and $x_i \neq x_j$ if $i \neq j$ then the set $\{X_1, X_2, X_m\}$ is called a system of distinct representatives (S.D.R) of the family S.

For example, consider X= {1, 2, 3, 4, 5} and S= $\{S_1, S_2, S_3, S_4, S_5\}$ where S_1={1, 2}, S_2={1, 2, 3}, S_3= {1, 2, 3}, S_4={1, 4, 3} and S_5={1, 5}. Now, {1, 2, 3, 4, 5} is a system of distinct fepresentatives of the family S. Instead, if we take S_1={1, 2}, S_2={1, 2, 3}, S_3 = {1, 2, 3}, S_4={1,5} and S_5 ={2, 5} then S has no system of distinct representatives.

Naturally, we can identify S. with a bipartite graph with bipartition (S, X) in which S_i € S is joined to every x € X contained in S. A system of distinct representatives is then a set of m independent edges (thus each S_i is incident with one of these edges).

It is customary to formulate this problem of finding S.D.R in terms of marriage arrangements.

The Marriage Problem

Suppose there are n boys each of whom has several girlfriends, under what conditions can we marry off the boys in such a way that each boy marries one of his girl friends? We assume that only single life partner marriage is allowed. This is known as marriage problem.

In graph theoretical terms, the above problem, can be stated as follows. Construct a bipartite graph G with bipartition (X,Y) where X = $\{x_1,X_2,...,X_n\}$ represents the. set of n boys and Y = $\{y_1, y_2\ldots y_m\}$ represents their girlfriends. An edge joins a vertex x_i to a vertex y_j if and only if y_j is a girl friend of x_i. The marriage problem is then equivalent to finding conditions for the existence of a matching in G which saturates every vertex of X.

For example, suppose there are five boys b_1, b_2, b_3, b_4 and b_5 and six girls g_1, g_2, g_3, g_4, g_5 and g_6 with their relationship as follows:

$$b_1 \longrightarrow \{g_1, g_2, g_3\} = S_1$$

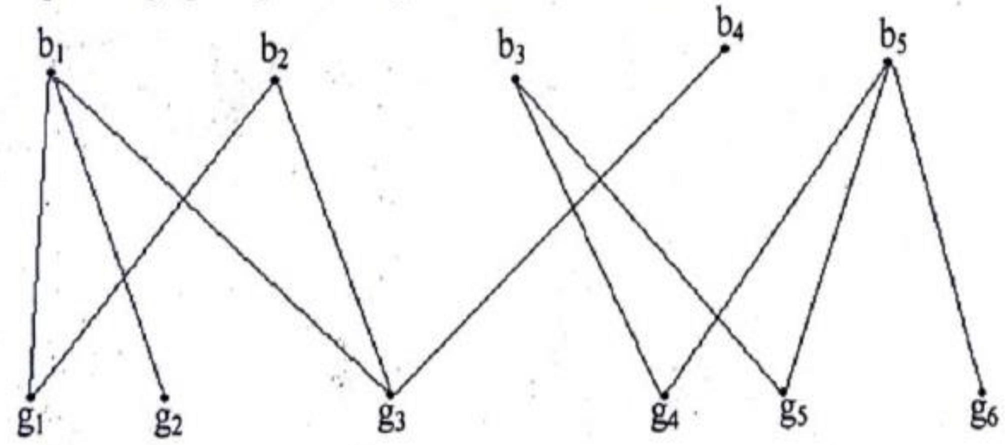
$$b_2 \longrightarrow \{g_1, g_3\} = S_2$$

$$b_3 \longrightarrow \{g_4, g_5\} = S_3$$

$$b_4 \longrightarrow \{g_3\} = S_4$$

$$b_5 \longrightarrow \{g_4, g_5, g_6\} = S_5$$

The bipartite graph representing this situation is shown



One of the solutions to this example is, b_1 to marry g_2, b_2 to marry g_1, b_3 to marry g_4, b_4 to marry g_3 and b_5 to marry g_5.

Now, we present a necessary and sufficient condition for the existence of a solution to the above marriage problem, first given by P. Hall(1935).

Theorem (Hall's Marriage theorem)

Let G be a bipartite graph with bipartition (X, Y). Then G contains a matching that saturates every vertex in X if and only if $|N(S)| \geq |S|$ for all $S \subseteq X$.

Proof: Suppose that G contains a matching M which saturates every vertex in X and let S be a subset of X. Since the vertices in S are matched under M with distinct vertices in N(S), we have $|N(S)| \geq |2S|$.

Conversely, Let G be a bipartite graph with $|N(S)| \geq |S|$ for all $S \subseteq X$. We assume that G has no matching which saturates all vertices in X. Let M* be a maximum matching in G. By our assumption, M* does not saturate all vertices in X. Let u be an M*- unsaturated vertex in X. Let Z denote the set of all vertices connected to u by M*-alternating paths. Since M* is a maximum matching in G, G has no M*-augmenting path. That is, u is the only M*-unsaturated vertex in Z. We set $S = Z \cap X$ and $T = Z \cap Y$. Clearly, the vertices in $S/\{u\}$ are matched under M* with vertices in T. So, we get $|T| = |S| - 1$ and $N(S) \supseteq T$. Since every vertex in N(S) is connected to u by an M*-alternating path, we also have $N(S) \subseteq T$ and hence $N(S) = T$. So, $|N(S)| = |T| = |S| - 1 < |S|$. This is a contradiction to the given hypothesis and hence G has a matching that saturates every vertex in X.

Now, let us reformulate the marriage theorem in terms of system of distinctive representatives.

Theorem 6.3 A family $S=\{S_1,S_2,..,S_m\}$ of sets has a system of distinctive representatives i

and only if $\left|\bigcup_{i\in F} S_i\right| \geq |F|$ for every $F \subseteq \{1,2,.., m\}$.

Corollary: If G is a k-regular bipartite graph with $k>0$, then G has a perfect matching.

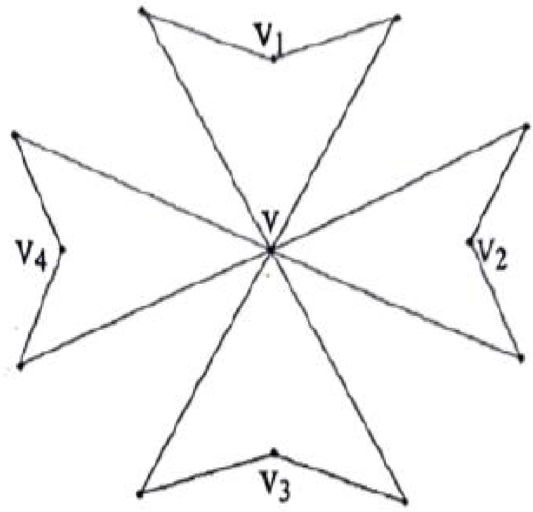
Proof: Let G be a k-regular bipartite graph with bipartition (X, Y). Since G is k-regular, $|X|=|Y|$

Now, let S be a subset of X and denote E_1 and E_2 the sets of edges incident with vertice in S and N(S) respectively. By definition of N(S), $E_1 \subseteq E_2$ and therefore $k|N(S)| = |E_2| \geq |E_1|$= $k|S|$. Therefore, $|N(S)| \geq |S|$ and hence, by theorem, that G has a matching M that saturate every vertex in X. Since $|X|=|Y|$, M is a perfect matching. Hence the corollary.

COVERING

Definition: A covering of a graph G is a subset K of V such that every edge of G has at least one end in K. For example, in the graph G of figure, the set

G_1:



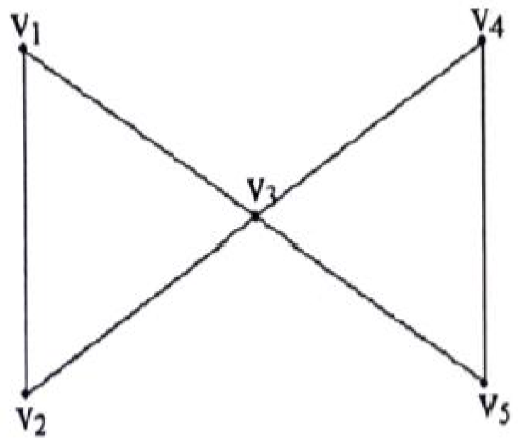
K={v, v_1, v_2, v_3, v_4} is a covering of G1.

Definition: A covering K is called a minimal covering of G if there is no covering K' of G such that K' ⊂ K.

For example, the covering K of G_1 is a minimal covering.

Definition: A covering K is called a minimum covering of G if G has no covering K' with |K'|<|K|.

G_2:



For example, $\{v_1, v_3, v_5\}$ is a minimum covering of the graph G_2. Also, we note that $\{v_1, v_2, v_4, v_5\}$ is a minimal covering but not minimum covering of G_2.

Remark: If K is a covering of G and M is a matching of G then K contains at least one end of each of the edges in M. Thus, for any matching M and any covering K, $|M| \leq |K|$. In particular, if M* is a maximum matching and K* is a minimum covering then,

$$|M^*| \leq |K^*|. \qquad (1)$$

In general, equality does not hold in (1). For example, consider the graph G_2. Here, $|M^*| = 2$ and $|K^*| = 3$. Under what conditions, does the equality hold? If G is bipartite then $|M^*| = |K^*|$. This result was proved by Konig and Egervary in 1931. Now, we present a lemma, which is useful in proving the Konig-Egervary theorem.

Lemma Let M be a matching and K be a covering such that $|M| = |K|$. Then M is a maximum matching and K is a minimum covering.

Proof: Let M* be a maximum matching and K* be a minimum covering of G. Then, $|M| \leq |M^*| \leq |K^*| \leq |K|$. Since $|M| = |K|$, in the above, equality must hold throughout and hence the lemma.

Konig-Egervary Theorem

In a bipartite graph, the number of edges in a maximum matching is equal to the number of vertices in a minimum covering.

Proof: Let G be a bipartite graph with bipartition (X, Y) and let M* be a maximum matching of G.

Suppose M* is perfect then $|X| = |Y| = |M^*|$. In this case X is a covering and the theorem holds.

So, we assume that M* is not perfect. Let U denote the set of all M*-unsaturated vertices in X and let Z be the set of all vertices connected by M*-alternating paths to vertices of U. Let $S = Z \cap X$ and $T = Z \cap Y$. Clearly every vertex in T is M*-saturated and N(S) = T (as in Hall's theorem). Define $K^* = (X/S) \cup T$. Every edge of G must have at least one of its ends in K*; otherwise, there would be an edge with one end in S and one end in YT, contradicting N(S) = T. Thus, K* is a covering of G and clearly $M''|= K^*|$. By lemma 6.5, K* is a minimum covering. Hence the theorem.

STABLE MATCHINGS

Now we turn to a special type of matchings, that is, matchings satisfying certain conditions. Matchings satisfying certain conditions are called stable matchings. In 1961, Gale and Shapley introduced stable matchings. It is customary to formulate the conditions and results in terms of marriage arrangements between boys and girls. So, naturally the corresponding graphs are simple bipartite and we consider only simple graphs, in this section. However, we have defined stable matching for a bipartite multigraph in the exercise.

Consider a bipartite graph G with bipartition (V_1, V_2) where V_1= {a, b,...} is the set of n boys and V_2= {A, B, ... } is the set of m girls. An edge a A means that the boy a knows the girl A. Suppose that each boy has an order of preferences on the set of girls he knows, and each girl has an order of preferences on the set of boys she knows. We assume that these orders are linear orders but place no other restriction on them.

Stable Matching: Given the preferences, a stable matching in G is a set M of independent edges of G such that if aB $\in$ E(G) - M, then either aA$\in$ M for some girl A preferred to B by a, or bB $\in$ M for some boy b preferred to a by B.

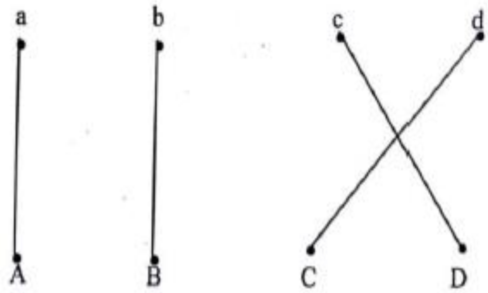
Thus, if a is not married to B, then either a is married to a girl he prefers to B, or else B is married to a boy she prefers to a. Otherwise the matching is "unstable"; a and B will leave their current partners and switch to each other.

Example. Consider a set of 4 boys {a, b, c, d}, a set of 4 girls {A, B, C, D} and their preferences as below.

Preference	1	2	3	4
a	A	B	C	D
b	A	C	B	D
c	C	D	A	B
d	C	B	A	D

	1	2	3	4
A	c	a	b	d
B	b	d	a	c
C	d	a	b	c
D	a	b	c	d

Here, the matching {aA, bB, cD, dC } is a stable matching.



Note: We have not assumed that a stable matching saturates all vertices in V_1 or V_2

Stable Matching Theorem For every assignment of preferences in a bipartite graph, there is a stable matching.

Proof: We consider a variant of the above algorithm, in which all boys and all girls act simultaneously, in rounds.

In every odd round, each boy proposes to his highest preference among those girls whom he knows and who have not yet refused him, and in every even round each girl refuses all but her highest suitor. The process ends when no girl refuses a suitor; then every girl marries her (only) suitor, if she has one. This process terminates after at most 2nm rounds, since at most m(n-1) proposals are refused, where n is the number of boys and m is the number of girls.

Since at every stage each boy proposes to at most one girl, and each girl rejects all but at most one boy, this algorithm results with a matching.

Now we prove that this matching is a stable matching. If $aB \in E(G)$-M, then either a never proposed to B, or a was refused by B during the algorithm. In the former case a marries a girl he prefers to B, as he never goes as low as B, and in the later case B refused a for a boy she prefers to a and got married. Hence this matching is a stable matching.