Module 5
INTRODUCTION

This topic is about a branch of discrete mathematics called graph theory. Discrete mathematics – the study of discrete structure (usually finite collections) and their properties include combinatorics (the study of combination and enumeration of objects) algorithms for computing properties of collections of objects, and graph theory (the study of objects and their relations).

Many problem in discrete mathematics can be stated and solved using graph theory therefore graph theory is considered by many to be one of the most important and vibrant fields within discrete mathematics.

Many problem in discrete mathematics can be stated and solved using graph theory therefore graph theory is considered by many to be one of the most important and vibrant fields within discrete mathematics.

DISCOVERY

It is no coincidence that graph theory has been independently discovered many times, since it may quite properly be regarded as an area of applied mathematics .Indeed the earliest recorded mention of the subject occurs in the works of Euler, and although the original problem he was considering might be regarded as a some what frivolous puzzle, it did arise from the physical world.

Kirchhoff's investigations of electric network led to his development of the basic concepts and theorems concerning trees in graphs. While Cayley considered trees arising from the enumeration of organic chemical isomer's. Another puzzle approach to graphs was proposed by Hamilton. After this, the celebrated four color conjecture came into prominence and has been notorious ever since. In the present century, there have already been a great many rediscoveries of graph theory which we can only mention most briefly in this chronological account.

WHY STUDY GRAPH?

The best way to illustrate the utility of graphs is via a "cook's tour" of several simple problem that can be stated and solved via graph theory. Graph theory has many practical applications in various disciplines including, to name a few, biology, computer science, economics, engineering, informatics, linguistics, mathematics, medicine, and social science, (As will become evident after reading this chapter) graphs are excellent modeling tools, we now look at several classic problems.

We begin with the bridges of Konigsberg. This problem has a historical significance, as it was the first problem to be stated and then solved using what is now known as graph theory. Leonard euler fathered graph theory in 1973 when his general solution to such problems was published euler not only solved this particular problem but more importantly introduced the terminology for graph theory.

1. **THE KONIGSBERG BRIDGE PROBLEM**

Euler (1707-- 1782) became the father of graph theory as well as topology when in 1736 he settled a famous unsolved problem of his day called the Konigsberg bridge problem. The city of

Konigsberg was located on the Pregel river in Prussia, the city occupied two island plus areas on both banks. These region were linked by seven bridges as shown in fig(1.1).

The problem was to begin at any of the four land areas, walk across each bridge exactly once and return to the starting point one can easily try to solve this problem empirically but all attempts must be unsuccessful, for the tremendous contribution of Euler in this case was negative.

In proving that the problem is unsolvable, Euler replaced each land area by a point and each bridge by a line joining the corresponding points these by producing a “graph” this graph is shown in fig(1.2) where the points are labeled to correspond to the four land areas of fig(1.1) showing that the problem is unsolvable is equivalent to showing that the graph of fig(1.2) cannot be traversed in a certain way.

In proving that the problem is unsolvable, Euler replaced each land area by a point and each bridge by a line joining the corresponding points these by producing a “graph” this graph is shown in fig(1.2) where the points are labeled to correspond to the four land areas of fig(1.1) showing that the problem is unsolvable is equivalent to showing that the graph of fig(1.2) cannot be traversed in a certain way.

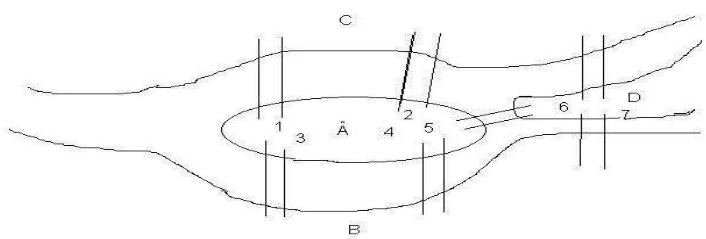


Figure1.1: A park in Konigsberg 1736

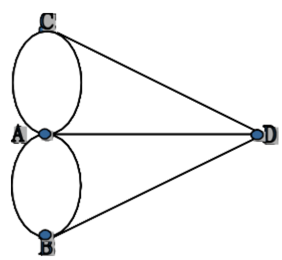


Figure1.2: The Graph of the Konigsberg bridge problem

Rather than treating this specific situation, Euler generalized the problem and developed a criterion for a given graph to be so traversable; namely that it is connected and every point is incident with an even number of lines. While the graph in fig(1.2) is connected, not every point incident with an even number of lines.

2. ELECTRIC NETWORKS

Kirchhoffs developed the theory of trees in 1847 in order to solve the system of simultaneous linear equations linear equations which gives the current in each branch and around each circuit of an electric network..

Although a physicist he thought like a mathematician when he abstracted an electric network with its resistances, condensers, inductances, etc, and replaced it by its corresponding combinatorial structure consisting only of points and lines without any indication of the type of electrical element represented by individual lines. Thus, in effect, Kirchhoff replaced each electrical network by its underlying graph and showed that it is not necessary to consider every cycle in the graph of an electric network separating in order to solve the system of equation.

Instead, he pointed out by a simple but powerful construction, which has since became std procedure, that the independent cycles of a graph determined by any of its "spanning trees" will suffice. A contrived electrical network N, its underlying graph G, and a spanning tree T are shown in fig(1.3)

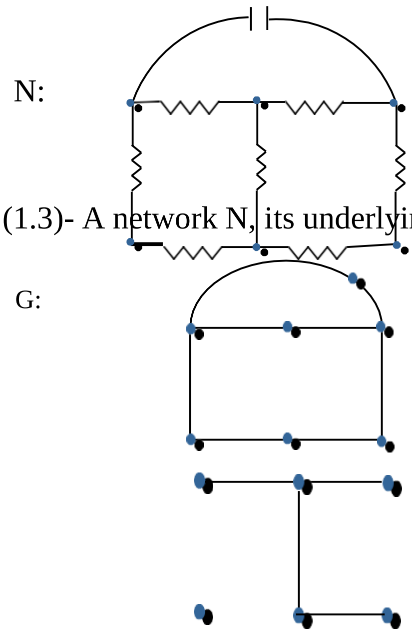
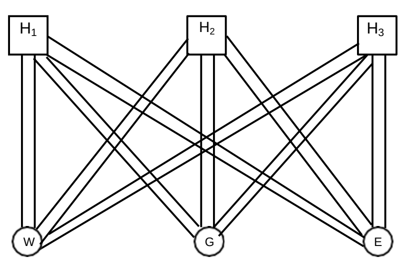


Fig (1.3)- A network N, its underlying graph G, and a spanning tree T

3. **UTILITIES PROBLEM**

These are three houses fig(1.4) H_1, H_2, and H_3, each to be connected to each of the three utilities water(w), gas(G), and electricity(E)- by means of conduits, is it possible to make such connection without any crossovers of the conduits?

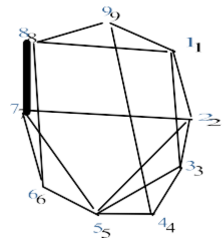


Fig(1.4)- three – utilities problem

Fig(1.4) shows how this problem can be represented by a graph – the conduits are shown as edges while the houses and utility supply centers are vertices

4. **SEATING PROBLEM**

Nine members of a new club meet each day for lunch at a round table they decide to sit such that every members has different neighbors at each lunch



Fig(1.5) – Arrangements at a dinner table

How many days can this arrangement lost?

This situation can be represented by a graph with nine vertices such that each vertex represent a member, and an edge joining two vertices represents the relationship of sitting next to each other. Fig(1.5) shows two possible seating arrangement – these are 1 2 3 4 5 6 7 8 9 1 (solid lines), and 1 3 5 2 7 4 9 6 8 1 (dashed lines) it can be shown by graph – theoretic considerations that there are only two more arrangement possible. They are 1 5 7 3 9 2 8 4 6 1 and 1 7 9 5 8 3 6 2 4 1. In general it can be shown that for n people the number of such possible arrangements is (n-1)/2, if n is odd. (n-2)/2, if n is even

WHAT IS A GRAPH?

A linear graph (or simply a graph) G = (V,E) consists of a set of objects V = {v_1, v_2,…..} called vertices, and another set E = {e_1, e_2,…..} whose elements are called edges, such that each edge e_k is identified with an unordered pair (v_i , v_j) of vertices. The vertices v_i , v_j associated with edge e_k are called the end vertices of e_k . The most common representation of a graph is by means of a diagram, in which the vertices are represented as points and each edge as a line segment joining its end vertices

The object shown in fig (a)

The Object Shown in Fig.(a)

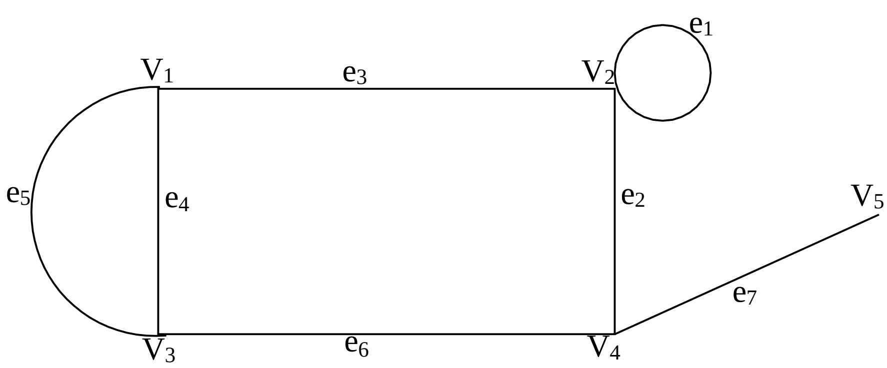
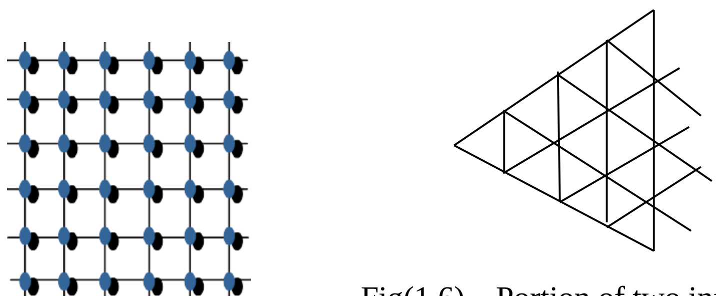


Fig (a) – Graph with five vertices and seven edges

Observe that this definition permits an edge to be associated with a vertex pair (v_i , v_j) such an edge having the same vertex as both its end vertices is called a self-loop. Edge e_1 in fig (a) is a self-loop. Also note that the definition allows more one edge associated with a given pair of vertices, for example, edges e_4 and e_5 in fig (a), such edges are referred to as 'parallel edges'. A graph that has neither self-loops nor **parallel edges** is called a '**simple graph**'.

FINITE AND INFINITE GRAPHS

Although in our definition of a graph neither the vertex set V nor the edge set E need be finite, in most of the theory and almost all application these sets are finite. A graph with a finite number of vertices as well as a finite number of edge is called a '**finite graph**': otherwise it is an **infinite graph**. The graphs in fig (a), (1.2), are all examples of finite graphs. Portions of two infinite graphs are shown below



Fig(1.6) – Portion of two infinite graphs

INCIDENCE AND DEGREE

When a vertex v_i is an end vertex of same edge e_j , v_i and e_j are said to be incident with (on or to) each other. In fig (a), for examples, edges e_2, e_6 and e_7 are incident with vertex v_4. Two nonparallel edges are said to be adjacent if there are incident on a common vertex. For example, e_2 and e_7 in fig (a) are adjacent. Similarly, two vertices are said to be adjacent if they are the end vertices of the same edge in fig (a), v_4 and v_5 are adjacent, but v_1 and v_4 are not.

The number of edges incident on a vertex v_i , with self-loops counted twice, is called the degree, d (v_i), of vertex v_i , in fig (a) for example $d(v_1) = d(v_2) = d(v_3) = 3$, $d(v_2) = 4$ and $d(v_5) = 1$. The degree of a vertex is same times also referred to as its valency.

Let us now considered a graph G with e edges and n vertices $v_1, v_2, \ldots v_n$ since each edge contributes two degrees

The sum of the degrees of all vertices in G is twice the number of edges in G that is

$$\sum_{i=1}^{n} d(v_i) = 2e ----(1.1)$$

Taking fig (a) as an example, once more $d(v_1) + d(v_2) + d(v_3) + d(v_4) + d(v_5) = 3 + 4 + 3 + 3 + 1 = 14$ = twice the number of edges.

From equation (1.1) we shall derive the following interesting result.

THEOREM 1.1

"The number of vertices of odd degree in a graph is always even".

Proof : If we consider the vertices with odd and even degree separately, the quantity in the left side of equation (1.1) can be expressed as the sum of two sum, each taken over vertices of even and odd degree respectively, as follows.

$$\sum_{i=1}^{n} d(v_i) = \sum_{even} d(vj) + \sum_{odd} d(v_k) \quad ---(1.2)$$

Since the left hand side in equation (1.2) is even, and the first expression on the right hand side is even (being a sum of even numbers), the second expression must also be even

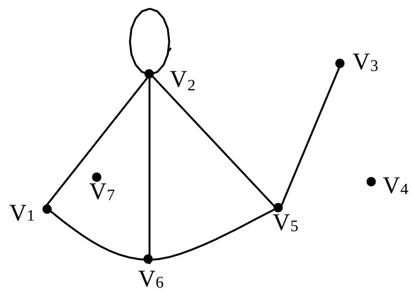
$$\sum_{odd} d(v_k) \quad = an - even - number ----(1.3)$$

Because in equation (1.3) each d(v_k) is odd, the total number of terms in the sum must be even to make the sum an even number. Hence the theorem.

A graph in which all vertices are of equal degree is called a **'regular graph'** (or simply a regular).

DEFINITION:

ISOLATED VERTEX , PENDANT VERTEX AND NULL GRAPH



Fig(1.7) – Graph containing isolated vertices, series edges, and a pendent vertex.

A vertex having no incident edge is called an '**isolated vertex**'. In other words, isolated vertices are vertices with zero degree. Vertices v_4 and v_7 in fig(1.7), for example, are isolated vertices a vertex of degree one is called a pendent vertex or an end vertex v_3 in fig(1.7) is a pendent vertex. Two adjacent edges are said to be in series if their common vertex is of degree two in fig(1.7), the two edges incident on v_1 are in series.

In the definition of a graph G = (V,E), it is possible for the edge set E to be empty. Such a graph , without any edges is called a '**null graph'**. In other words, every vertex in a null graph is an isolated vertex. A null graph of six vertices is shown in fig (1.8). Although the edge set E may empty the vertex set V must not be empty; otherwise there is no graph. In other words, by definition, a graph must have atleast one vertex

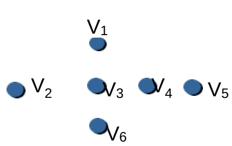


Fig 1.8: Null graph of Six Vertices

A BRIEF HISTORY OF GRAPH THEORY

As mentioned before, graph theory was born in 1736 with Euler's paper in which he solved Konigsberg bridge problem. For the next 100 years nothing more was done in the field.

In 1847,G.R.Kirchhoff (1824-1887) developed the theory of trees for their applications in Electrical network. Ten years later, A. Cayley (1821-1895) discovered trees while he was trying to enumerate the isomers of saturated hydrocarbons C_nH_{2n+2}.

About the time of Kirchhoff and Cayley,two other milestones in graph theory were laid. One was the four-color conjecture, which states that four colors are sufficient for coloring any atlas(a map on a plane)such that the countries with common boundaries have different colors.

It is believed that A.F. Mobius (1790-1868) first presented four-color problem in one of his lectures in 1840.

About 10 years later A.De Morgan(1806-1871) discussed this problem with his fellow mathematicians in London.De Morgan's letter is the first authenticated reference to the four-color problem.The problem became well known after Cayley published it in 1879 in the first volume of the **Proceedings of the Royal Geographic Society .**To this day ,the four-color conjecture is by far the most famous unsolved problem in Graph theory. It has stimulated an enormous of research in the field.

The other milestone is due to Sir W.R. Hamilton (1805-1865). In the year 1859,he invented a puzzle and sold it for 25 guineas to a game manufacturer in Dublin. The puzzle consisted of a wooden ,regular Dodecahedron (A polyhedron with 12 faces and 20 corners, each face being a regular pentagon and three edges meeting at each corner). The corners were marked with the names of 20 important cities; London, Newyork, Delhi, Paris and so on .The object in the puzzle was to find a route along the edges of the Dodecahedron, passing through each of the 20 cities exactly once.

Although the solution of this specific problem is easy to obtain, to date no one has found a necessary and sufficient condition for the existence of such a route (called Hamiltonian circuit) in an arbitrary graph.

This fertile period was followed by half a century of relative inactivity. Then a resurgence of interest in graphs started during the 1920's.One of the pioneers in this period was D. Konig. He organized the work of other mathematicians and his own and wrote the first book on the subject which was published in 1936.

The past 30 years has been a period of intense activity in graph theory both pure and applied. A great deal of research has been done and is being done in this area. Thousands of papers have been published and more than hundred of books written during the past decade. **Among the current leaders in the field are Claude Berg, Oystein Ore, Paul Erdos, William Tutte and Frank Harary.**

DIRECTED GRAPHS AND GRAPHS:

DIRECTED GRAPHS :

Look at the diagram shown below. This diagram consists of four vertices A,B,C,D and three edges AB,CD,CA with directions attached to them .The directions being indicated by arrows.

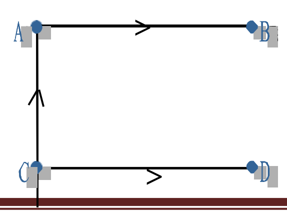


Fig.

Because of attaching directions to the edges, the edge AB has to be interpreted as an edge from the vertex A to the vertex B and it cannot be written as BA. Similarly the edge CD is from C to D and cannot be written as DC and the edge CA is from C to A and cannot be written as AC .Thus here the edges AB, CD, CA are directed edges.

The directed edge AB is determined by the vertices A and B in that order and may therefore be represented by the ordered pair (A,B). similarly, the directed edge CD and CA may be represented by the ordered pair(C,D) and (C,A) respectively. Thus the diagram in fig(1.1) consists of a nonempty set of vertices, namely {A,B,C,D} and a set of directed edges represented by ordered pairs {(A,B),(C,D),(C,A) }.Such a diagram is called a diagram of a directed graph.

DEFINITION OF A DIRECTED GRAPH :

A directed graph (or digraph) is a pair (V,E), where V is a non empty set and E is a set of ordered pairs of elements taken from the set V.

For a directed graph (V, E), the elements of V are called **Vertices** (points or nodes) and the elements of E are called **"Directed Edges".** The set V is called the **vertex set** and the set E is called the **directed** edge set

The directed graph (V,E) is also denoted

by D=(V,E) or D =D(V,E).

The geometrical figure that depicts a directed graph for which the vertex set is

V={A,B,C,D} and the edge set is

E={AB,CD,CA}={(A,B),(C,D),(C,A)}

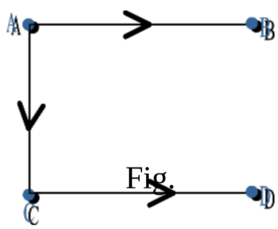


Fig.

Fig(1.2) depicts the directed graph for which the

vertex set is V={A,B,C,D} and the edge set is

E={AB,CD,AC}={(A,B),(C,D),(A,C)}.

It has to be mentioned that in a directed graph the directed edges need not be straight line segments, they can be curve lines (arcs)Also.

For example, a directed edge AB of a directed graph can be represented by an arbitrary arc drawn from the vertex A to the vertex B as shown in fig(1.3).

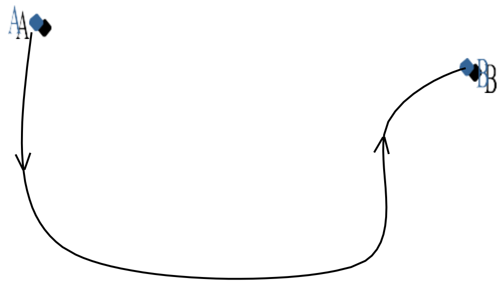


Fig.

In fig (1.1) every directed edge of a digraph (directed graph) is determined by two vertices of the diagraph- a vertex from which it begins and a vertex at which it ends. Thus ,if AB is a directed edge of a digraph D. Then it is understood that this directed edge begins at the vertex A of D and terminates at the vertex B of D. Here we say that A is the **<u>initial vertex</u>** and B is the **<u>terminal vertex</u>** of AB.

It should be mentioned that for a directed edge (in a digraph) the initial vertex and the terminal vertex need not be different. A directed edge beginning and ending at the same vertex A is denoted by AA or (A,A) and is called **<u>directed loop.</u>** The directed edge shown in Fig.(1.4) is a directed loop which begins and ends at the vertex A.

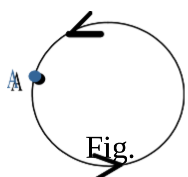
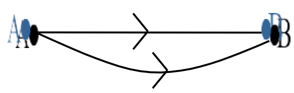
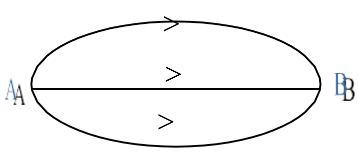


Fig.

A digraph can have more than one directed edge having the same initial vertex and the same terminal vertex. Two directed edges having the same initial vertex and the same terminal vertex are called <u>parallel directed edges</u>.

Two parallel directed edges are shown in fig(1.5)(a).





Two or more directed edges having the same initial vertex and the same terminal vertex are called **“multiple directed edges”.** Three multiple edges are shown in fig(1.5)(b).

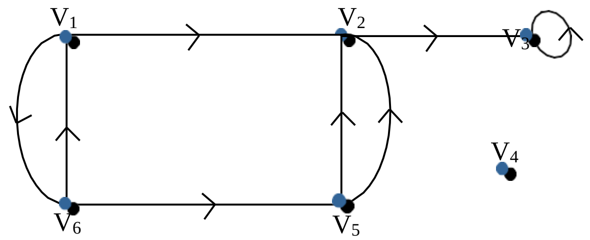
IN- DEGREE AND OUT –DEGREE

If V is the vertex of a digraph D, the number of edges for which V is the initial vertex is called the **outgoing degree** or the **out degree** of V and the number of edges for which V is the terminal vertex is called the **incoming degree** or the **in degree** of V. The out degree of V is denoted by d^+ (v) or o d (v) and the in degree of V is denoted by d^- (v) or i d (v).

It follows that

i. d^+ (v) =0, if V is a sink
ii. d^- (v) =0, if V is a source
iii. d^+ (v) = d^- (v) = 0, if V is an isolated vertex.

For the digraph shown in fig(1.6) the out degrees and the in degrees of the vertices are as given below



$d^+ (v_1) = 2$	$d^- (v_1) = 1$
$d^+ (v_2) = 1$	$d^- (v_2) = 3$
$d^+ (v_3) = 1$	$d^- (v_3) = 2$
$d^+ (v_4) = 0$	$d^- (v_4) = 0$
$d^+ (v_5) = 2$	$d^- (v_5) = 1$
$d^+ (v_6) = 2$	$d^- (v_6) = 1$

We note that ,in the above digraph, there is a directed loop at the vertex v_3 and this loop contributes a count 1 to each of $d^+ (v_3)$ and $d^- (v_3)$.

We further observe that the above digraph has 6 vertices and 8 edges and the sums of the out-degrees and in-degrees of its vertices are

$$\sum_{i=1}^{6} d^+(v_j) = 8, \quad \sum_{i=1}^{6} d^-(v_i) = 8$$

Example 1: Find the in- degrees and the out-degrees of the vertices of the digraph shown in fig (1.8)

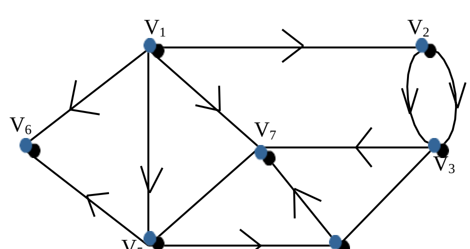


Fig.

SOLUTION:

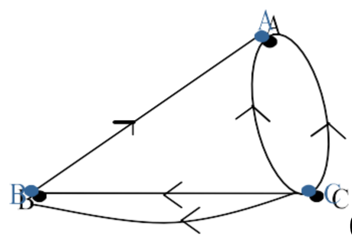
The given digraph has 7 vertices and 12 directed edges. The out-degree of a vertex is got by counting the number of edges that go out of the vertex and the in-degree of a vertex is got by counting the number of edges that end at the vertex. Thus we obtain the following data

Vertex	V_1	V_2	V_3	V_4	V_5	V_6	V_7
Out-degree	4	2	2	1	3	0	0
In-degree	0	1	2	2	1	2	4

This table gives the out-degrees and in-degrees of all the vertices. We note that v_1 is a source and v_6 and v_7 are sinks.

We also check that sum of out-degrees = sum of in – degrees = 12 = No of edges.

Example 2: Write down the vertex set and the directed edge set of each of the following digraphs.



(i)

Fig.

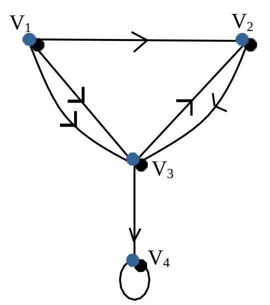


Fig. (ii)

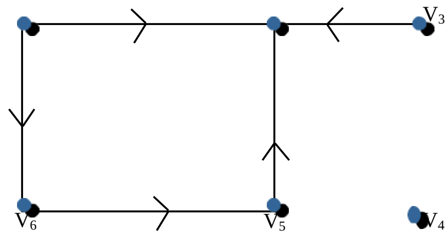
Solution of graph (i) & (ii):

i) This is a digraph whose vertex set is
   V={A,B,C} and the directed edge set
   E={(B,A),(C,A),(C,A),(C,B),(C,B)}.

ii) This is a digraph whose vertex set is
   V={V_1,V_2,V_3, V_4} and the directed edge set
   E={(V_1,V_2),(V_1,V_3),(V_1,V_3),
   (V_2,V_3),(V_3,V_2)),(V_3,V_4) ,(V_4,V_4)}.

Example 3: For the digraph shown in fig, determine the out-degrees and in-degrees of all the vertices

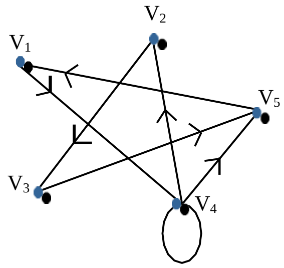
V_1 V_2



Solution: $d^-(V_1) = 0$, $d^-(v_2) = 3$, $d^-(v_3) = 0$, $d^-(v_4) = 0$, $d^-(v_5) = 1$, $d^-(v_6) = 1$
$d^+(v_1) = 2$, $d^+(v_2) = 0$, $d^+(v_3) = 1$, $d^+(v_4) = 0$, $d^+(v_5) = 1$, $d^+(v_6) = 1$

Example 4: Let D be the digraph whose vertex set
$V=\{V_1,V_2,V_3, V_4, V_5\}$ and the directed edge set is
$E =\{(V_1,V_4),(V_2,V_3), (V_3,V_5),(V_4,V_2),(V_4,V_4),(V_4,V_5),(V_5,V_1)\}$.

Write down a diagram of D and indicate the out-degrees and in-degrees of all the vertices



vertices	V_1	V_2	V_3	V_4	V_5
D^+	1	1	1	3	1
d^-	1	1	1	2	2

DEFINITION :

SIMPLE GRAPH :

A graph which does not contain loops and multiple edges is called simple graph.

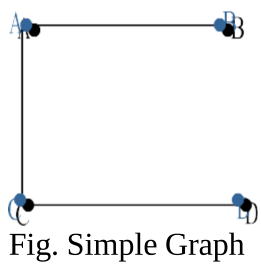


Fig. Simple Graph

LOOP FREE GRAPH.

A graph which does not contain loop is called loop free graph.

MULTIGRAPH

A graph which contain multiple edges but no loops is called <u>multigraph</u>.

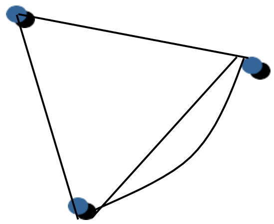
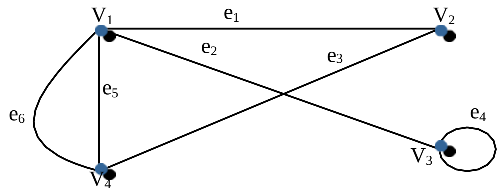


Fig. Multigraph

GENERAL GRAPH

A graph which contains multiple edges or loops (or both) is called <u>general graph.</u>



<u>COMPLETE GRAPH :</u>

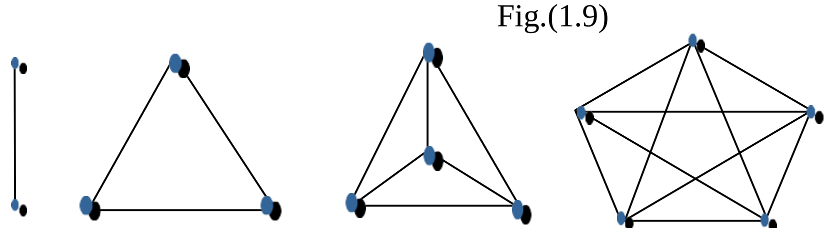
A simple graph of order ≥ 2 in which there is an edge between every pair of vertices is called a complete graph (or a full graph).

In other words a complete graph is a simple graph in which every pair of distinct vertices are adjacent.

A complete graph with $n \geq 2$ vertices is denoted by K_n .

A complete graph with 2,3,4,5 vertices are shown in fig (1.9)(a) to (1.9)(d) respectively. Of these complete graphs ,the complete graph with 5 vertices namely K_5(shown in fig.1.9 (d),is of great importance. This graph is called the Kuratowski's first graph

Fig.(1.9)



(a) : K_2 (b) : K_3 (c) : K_4 (d) : K_5

BIPARTITE GRAPH

Suppose a simple graph G is such that its vertex set V is the union of two of its mutually disjoint non-empty subsets V_1 and V_2 which are such that each edge in G joins a vertex inV_1 and a vertex inV_2.Then G is called a **bipartite graph**. If E is the edge set of this graph, the graph is denoted by G = (V_1, V_2: E), or G = G(V_1, V_2: E). The sets $V_1$and V_2 are called **bipartites** (or partitions) of the vertex set V.

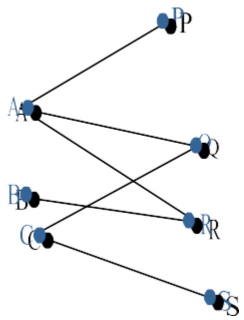


Fig. (1.10)

For example, consider the graph G in fig(1.10) for which the vertex set is
V={A,B,C,P,Q,R,S} and the edge set is
E= {AP,AQ,AR,BR,CQ,CS}. Note that the set V is the union of two of its subsets V_1={ A,B,C} and V_2={P,Q,R,S} which are such that

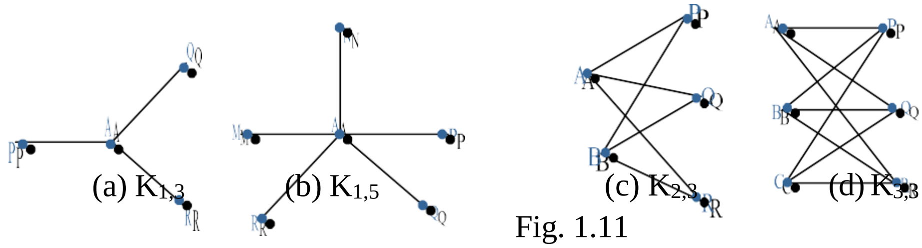
i) V_1 and V_2 are disjoint.
ii) Every edge in G joins a vertex in V_1 and a vertex in V_2.
iii) G contains no edge that joins two vertices both of which are in V_1 or V_2. This graph is a bipartite graph with V_1={ A,B,C} and V_2={P,Q,R,S} as bipartites.

COMPLETE BIPARTITE GRAPH

A bipartite graph G= {V_1, V_2 ; E} is called a complete bipartite graph, if there is an edge between every vertex in V_1 and every vertex in V_2 .

The bipartite graph shown in fig (1.10) is not a complete bipartite graph. Observe for example that the graph does not contain an edge joining A and S.

A complete bipartite graph G={ V_1, V_2 ; E} in which the bipartites $V_1$and V_2 contain r and s vertices respectively, with $r \leq s$ is denoted by $K_{r,s}$ .In this graph each of r vertices in V= is joined to each of s vertices in V_2 .Thus $K_{r,s}$ has r+ s vertices and rs edges. That is $K_{r,s}$ is of order r+s and size rs. It is therefore a (r+ s,rs) graph



(a) $K_{1,3}$ (b) $K_{1,5}$ (c) $K_{2,3}$ (d) $K_{3,3}$

Fig. 1.11

Fig 1.11 (a) to (d) depict some bipartite graphs. Observe that in fig 1.11(a),the bipartites are V_1={ A } and V_2={P,Q,R}; the vertex A is joined to each of the vertices P,Q,R by an edge. In fig 1.11(b) ,the bipartites are V_1={A} and

V_2={M,N,P,Q,R}; the vertex A is joined to each of the vertices M,N,P,Q,R by an edge. In fig 1.11(c) ,the bipartites are V_1={ A,B } and V_2={ P,Q,R}; each of the vertices A and B is joined to each of the vertices P,Q,R by an edge. In fig 1.11(d),the bipartites are V_1={ A,B,C } and V_2={P,Q,R}; each of the vertices A,B,C is joined to each of the vertices P,Q,R. Of these complete bipartite graph the graph $K_{3,3}$ shown in fig 1.11(d),is of great importance. This is known as **Kuratowski's second graph.**

Example 1. Draw a diagram of the graph G = (V,E) in each of the following cases.

a) V= { A,B,C,D} ,E={AB,AC,AD,CD}

b) V={V_1,V_2,V_3, V_4 ,V_5 },
E={V_1V_2 ,V_1V_3,V_2V_3,V_4V_5}.

c) V= {P,Q,R,S,T} ,E={PS,QR,QS}

d) V={ V_1,V_2,V_3, V_4 ,V_5,V_6},
E={V_1V_4,V_1V_6,V_4V_6,V_3V_2,V_3V_5,V_2V_5}

Solution : The required diagram are shown below

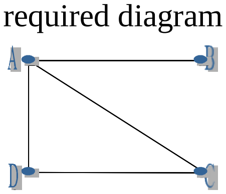


Fig: (a)

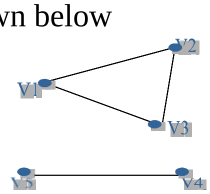


Fig: (b)

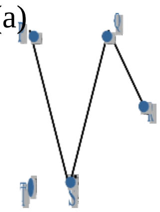


Fig: (c)

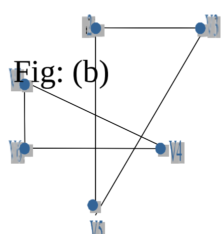
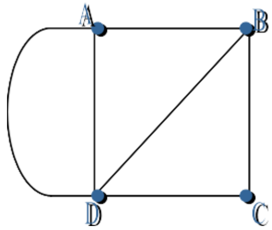
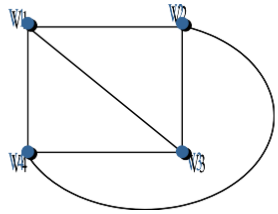


Fig: (d)

Example 2: Which of the following is a complete graph?



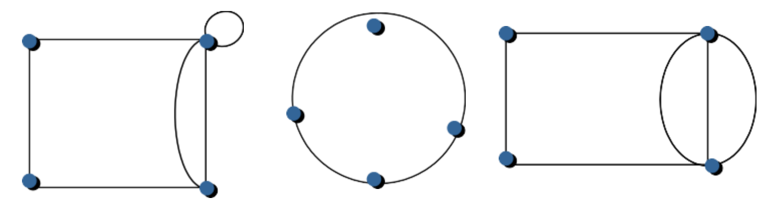
(a)



(b)

Solution: The first of the graph is not complete. It is not simple on the one hand and there is no edge between A and C on the other hand. The second of the graphs is complete. It is a simple graph and there is an edge between every pair of vertices.

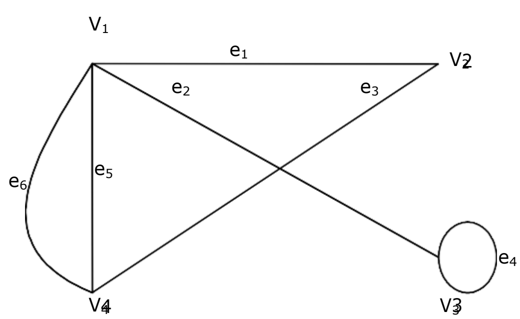
Example 3: Which of the following graphs is a simple graph? a multigraph ? a general graph ?



(i) (ii) (iii)

Solution: (i) General Graph,
(ii) Simple Graph,
(iii) Multigraph

Example 4: Identify the adjacent vertices and adjacent edges in the graph shown in Figure.



Solution :
Adjacent Vertices : V_1 & V_2, V_1 & V_3, V_1 & V_4, V_2 & V_4.
Adjacent edges : e_1 & e_2, e_1 & e_3, e_1 & e_5, e_1 & e_6, e_2 & e_4, e_2 & e_5, e_2 & e_6, e_3 & e_5, e_3 and e_6.

***VERTEX DEGREE AND HANDSHAKING PROPERTY* :**

Let G = (V,E) be a graph and V be a vertex of G. Then the number of edges of G that are incident on V (that is, the number of edges that join V to other vertices of G) with the loops counted twice is called the ***degree*** of the vertex V and is denoted by deg(v) or d(V).

The degree of the vertices of a graph arranged in non-decreasing order is called the ***degree sequence*** of the graph. Also, the minimum of the degree of a graph is called the ***degree of the graph***

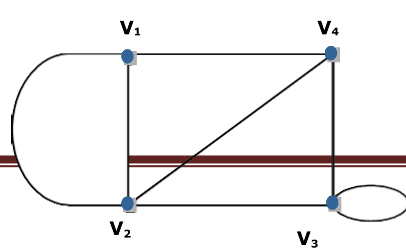
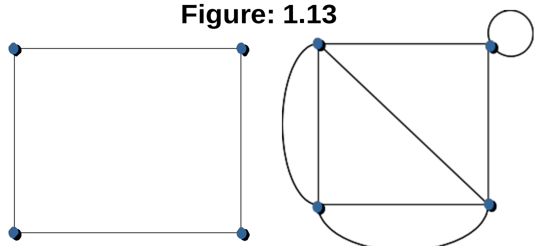


Figure (1.12)

For example, the degrees of vertices of the graph shown in fig are as given below
$d(V_1) = 3$, $d(V_2) = 4$, $d(V_3) = 4$, $d(V_4) = 3$
Therefore,the degree sequence of the graph is 3,3,4,4 and the degree of the graph is 3.
Regular Graph **:** A graph in which all the vertices are of the same degree K is called a regular graph of degree K, or a K- regular graph.In particular, a 3-regular graph is called a ***cubic graph***.
The graph shown in figures 1.13 (a) and (b) are 2- regular and 4 - regular graph respectively.

Figure: 1.13



(a) (b)

The graph shown in fig1.13 (c) is a 3-regular graph (cubic graph). This particular cubic graph, which contains 10 vertices and 15 edges, is called the ***Peterson Graph***.

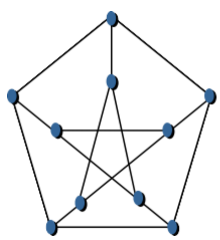


Figure (c)

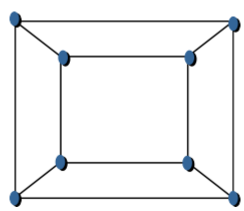


Figure (d)

The graph shown in fig (d) is a cubic graph with $8 = 2^3$ vertices. This particular graph is called the three dimentional hyper cube and is denoted by Q_3.

Handshaking property :
Let us refer back to degree of the graph shown in fig 1.14. we have, in this graph,

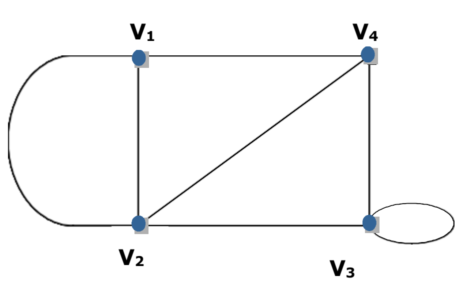


fig (1.14)

$d(V_1) = 3$, $d(V_2) = 4$, $d(V_3) = 4$, $d(V_4) = 3$

Also,the graph has 7 edges, we observe that $\deg(V_1) + \deg(V_2) + \deg(V_3) + \deg(V_4) = 14 = 2 \times 7$

Property: The sum of the degrees of all the vertices in a graph is an even number, and this number is equal to twice the number of edges in the graph.

In an alternative form, this property reads as follows:

For a graph G = (V,E) $\sum_{v \in V} \deg(v) = 2|E|$

This property is obvious from the fact that while counting the degree of vertices, each edge is counted twice (once at each end).

The aforesaid property is popularly called the **'*handshaking property*'**

Because, it essentially states that if several people shake hands, then the total number of hands shaken must be even, because just two hands are involved in each hand shake.

Theorem : In every graph the number of vertices of odd degrees is even

Proof : Consider a graph with n vertices. Suppose K of these vertices are of odd degree so that the remaining n-k vertices are of even degree. Denote the vertices with odd degree by $V_1, V_2, V_3, \ldots, V_k$ and the vertices with even degree by $v_{k+1}, v_{k+2}, \ldots, v_n$ then the sum of the degrees of vertices is

$$\sum_{i=1}^{n} \deg(v_i) = \sum_{i=1}^{k} \deg(v_i) + \sum_{i=k+1}^{n} \deg(v_i) ----(1)$$

In view of the hand shaking property, the sum on the left hand side of the above expression is equal to twice the number of edges in the graph. As such, this sum is even. Further, the second sum in the right hand side is the sum of the degrees of the vertices with even degrees. As such this sum is also even. Therefore, the first sum in the right hand side must be even; that is,

$\deg(V_1) + \deg(V_2) + \text{--} + \deg(V_k) = \text{Even}$—(ii)

But, each of $\deg(V_1), \deg(V_2), \ldots, \deg(V_k)$ is odd. Therefore, the number of terms in the left hand side of (ii) must be even; that is, K is even

Example : For the graph shown in fig 1.15 indicating the degree of each vertex and verify the handshaking property

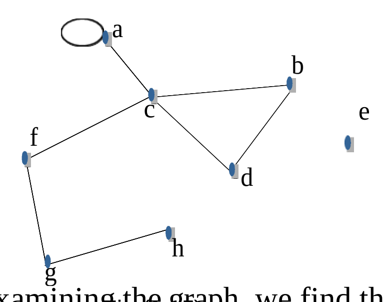


Fig (1.15)

Solution : By examining the graph, we find that the degrees of its vertices are as given below:

deg (a) = 3, deg (b) = 2, deg (c) = 4, deg (d) = 2, deg(e) = 0, deg (f) = 2, deg (g) = 2, deg (h) = 1.

We note that e is an isolated vertex and h is a pendant vertex.

Further, we observe that the sum of the degrees of vertices is equal to 16. Also, the graph has 8 edges. Thus, the sum of the degrees of vertices is equal to twice the number of edges.

This verifies the handshaking property for the given graph.

Example : For a graph with n-vertices and m edges, if δ is the minimum and Δ is the maximum of the degrees of vertices, show that $\delta \leq \frac{2m}{n} \leq \Delta$

Solution : Let $d_1, d_2, \ldots, d_n$, be the degrees of the vertices. Then, by handshaking property, we have
$d_1 + d_2 + d_3 + \text{-----} + d_n = 2m$ --------- (i)

Since $\delta = \min(d_1, d_2, \ldots, d_n)$, we have $d_1 \geq \delta$,

$d_2 \geq \delta, \ldots\ldots, d_n \geq \delta$.

Adding these n inequalities, we get

$d_1 + d_2 + \text{----} + d_n \geq n\delta$ ------(ii)

Similarly, since $\Delta = \max(d_1, d_2, \ldots, d_n)$, we get

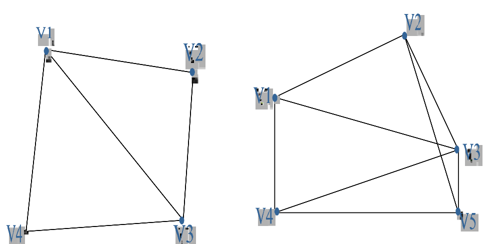
$d_1 + d_2 + \text{----} + d_n \leq n\Delta$----- (iii)

From (i), (ii) and (iii), we get $2m \geq n\delta$ and $2m \leq n\Delta$, so that $n\delta \leq 2m \leq n\Delta$,

or

$$\delta \leq \frac{2m}{n} \leq \Delta$$

SUBGRAPHS



(a) : G_1 (b) : G

Fig. (1.6)

Given two graphs G and G_1, we say that G_1 is a ***subgraph*** of G if the following conditions hold:

(1). All the vertices and all the edges of G_1 are in G.

(2). Each edges of G_1 has the same end vertices in G as in G_1.

Essentially, a subgraph is a graph which is a part of another graph. Any graph isomorphic to a subgraph of a graph G is also referred to as a subgraph of G.

Consider the two graphs G_1 and G shown in figures 1.16(a) and 1.16(b) respectively, we observe that all vertices and all edges of the graph G_1 are in the graphs G and that every edge in G_1 has same end vertices in G as in G_1.Therefore G_1 is a subgraph of G .In the diagram of G ,the part G_1 is shown in thick lines.

The following observation can be made immediately.

i) Every graph is a sub-graph of itself.

ii) Every simple graph of n vertices is a subgraph of the complete graph K_n.

iii) If G_1 is a subgraph of a graph G_2 and G_2 is a subgraph of a graph G,then G_1 is a subgraph of a graph G.

iv) A single vertex in a graph G is a subgraph of a graph G.

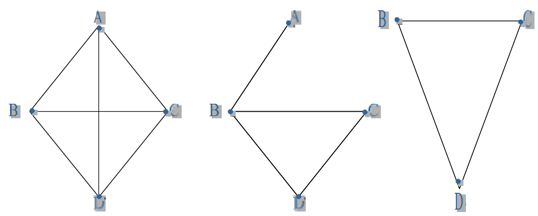
v) A single edge in a graph G together with its end vertices,is a subgraph of G

SPANNING SUBGRAPH :

Given a graph G=(V, E), if there is a subgraph $G_1=(V_1,E_1)$ of G such that $V_1=V$ then G_1 is called a spanning subgraph of G.

In other words , a subgraph G_1 of a graph G is a spanning subgraph of G whenever the vertex set of G_1 contains all vertices of G. Thus a graph and all its spanning subgraphs have the same vertex set. Obviously every graph is its own spanning subgraph.

Figure (1.17)



(a) (b) (c)

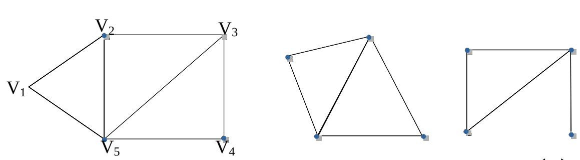
For example, for the graph shown in fig1.17(a), the graph shown in fig 1.17(b) is a spanning subgraph where as the graph shown in fig1.17(c) is a subgraph but not a spanning subgraph

INDUCED SUBGRAPH

Given a graph G=(V,E), suppose there is a subgraph $G_1=(V_1,E_1)$ of G such that every edge {A,B} of G, where AB ϵ V_1 is an edge of G_1 also .then G_1 is called an induced subgraph of G (induced by V_1) and is denoted by $< V_1 >$.

It follows that a subgraph $G_1=(V_1,E_1)$ of a graph G=(V,E) is not an induced subgraph of G, if for some A,B ϵ V_1,there is an edge{A,B} which is in G but not in G_1.

For example, for the graph shown in the figure 1.18 (a), the graph shown in the figure 1.18 (b), is an induced subgraph, induced by the set of vertices $V_1=\{v_1,v_2,v_3,v_5\}$ where as the graph shown in the figure 1.18 (c) is not an induced subgraph



(a) (b) (c)(c)

Figure 1.18 (a, b & c)

EDGE-DISJOINT AND VERTEX-DISJOINT SUBGRAPHS

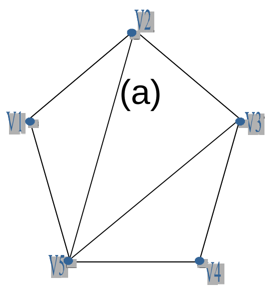
Let G be a graph and G_1 and G_2 be two subgraphs of G. then

G_1 and G_2 are said to be edge disjoint if they do not have any common edge.

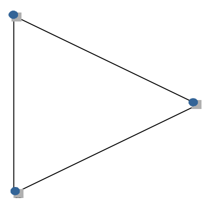
G_1 and G_2 are said to be vertex disjoint if they do not have any common edge and any common vertex.

It is to be noted that edge disjoint subgraphs may have common vertices. Subgraphs that have no vertices in common cannot possibly have edges in common.
For example ,for the graph shown in the figure 1.19 (a), the graph shown in the figure1.19 (b) and 1.19 (c) are edge disjoint but not vertex disjoint subgraphs.

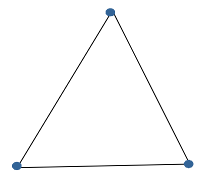
Figure 1.19:



(a)



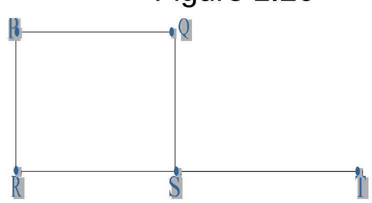
(b)



(c)

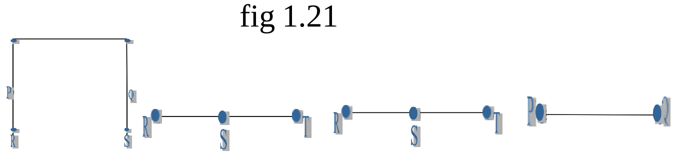
Example : For the graph shown in fig 1.20 ,find two edge-disjoint subgraphs and two vertex-disjoint subgraphs .

Figure 1.20



Solution:for the given graph, two edge-disjoint subgraphs are shown in fig 1.21(a) and two vertex-disjoint subgraphs are shown in fig 1.21(b).

fig 1.21



(a) (b)

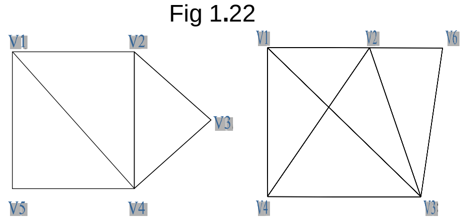
OPERATIONS ON GRAPHS

Consider two graphs $G_1=(V_1,E_1)$ and $G_2=(V_2,E_2)$ then the graph whose vertex set is $V_1 U V_2$ and edge set is $E_1 U E_2$ is called the <u>union</u> of G_1 and G_2 and is denoted by $G_1 U G_2$.
Thus $G_1 U G_2=(V_1 U V_2, E_1 U E_2)$.
Similarly, if $V_1 \cap V_2 \neq \phi$,the graph whose vertex set is $V_1 \cap V_2$ and the edge set $E_1 \cap E_2$ is called intersection of G_1 and G_2.It is denoted by $G_1 \cap G_2$.Thus $G_1 \cap G_2=(V_1 \cap V_2, E_1 \cap E_2)$, if $V_1 \cap V_2 \neq \phi$.

Next suppose we consider the graph whose vertex set is V_1UV_2 and edge set is $E_1\Delta E_2$ where $E_1\Delta E_2$ is the symmetric difference of E_1 and E_2.This graph is called the ring sum of G_1 and G_2.It is denoted by $G_1\Delta G_2$. Thus $G_1\Delta G_2 = (V_1UV_2, E_1\Delta E_2)$.

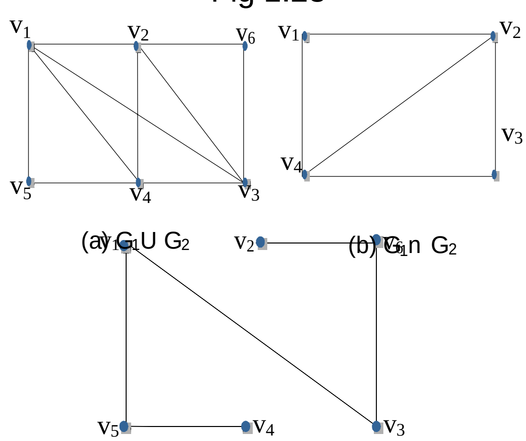
For the two graphs G_1 and G_2 shown in figures 1.22 (a) and (b), their union ,intersection and ring sum are shown in figures 1.23 (a), (b) and (c) respectively.

Fig 1.22



(a) : G_1 (b) : G_2

Fig 1.23



(a) G_1U G_2 (b) G_1n G_2

Fig 1.23: (c) $G_1\Delta G_2$

DECOMPOSITION

We say that a graph G is decomposed (or partitioned) in to two subgraphs G_1 & G_2 if G_1 Ù G_2 = G & $G_1 \cap G_2$ = null graph

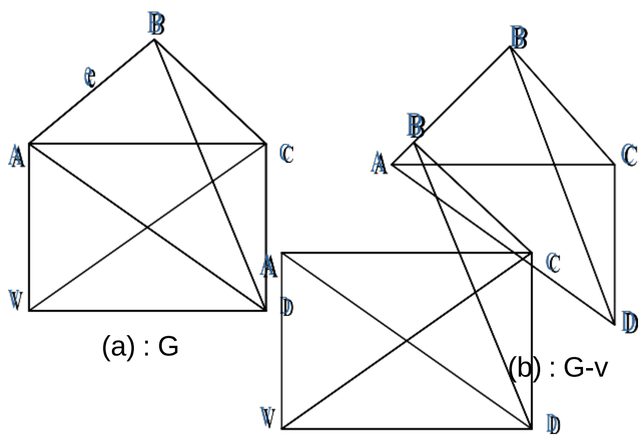
DELETION:

If V is a vertex in a graph G, then G – V denotes the subgraph of G obtained by deleting V and all edges incident in V, from G this subgraph G-u, is refered to as **vertex deleted subgraph** of G.

It should be noted that, the deletion of a vertex always results in the deletion of all edges incident on that vertex.

If e is an edge in a graph G, then G-e denotes the subgraph of G obtained by deleting e (but not its end vertices) from G. This subgraph, G-e, is referred to as **edge – deleted subgraph** of G. For the graph G shown in figure 1.24 (a), the subgraphs G-V and G-e are shown in figure 1.24 (b) and 1.24 (c) respectively.

Figure 1.24 (a, b, c)



(a) : G

(b) : G-v

(c) : G-e

COMPLEMENT OF A SUBGRAPH

Given a graph G and a subgraph G_1 of G ,the subgraph of G obtained by deleting from all the edges that belongs to G_1 is called the complement of G_1 in G;it is denoted by $G-G_1$ or G_1

In other words ,if E_1 is the set of all edges of G_1 then the complement of G_1 in G is given by $G_1 = G-E_1$.We ca~~n~~ check that $G_1=G\Delta G_1$.

For example :

Consider the graph G shown in fig 1.25(a) .Let G_1 be the subgraph of G shown by thick lines in this figure. The complement of G_1 in G, namely G_1, is as shown in ~~f~~ig 1.25(b)

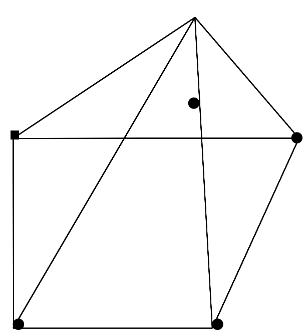


Fig. 1.25(a)

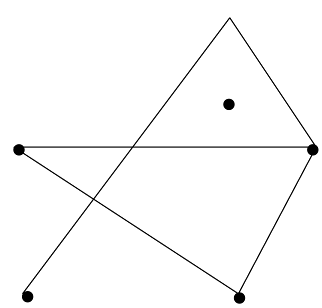


Fig.1.25 (b)

COMPLEMENT OF A SIMPLE GRAPH

Earlier we have noted that every simple graph of order n is a subgraph of the complete graph Kn .If G is a simple graph of order n ,then the complement of G in K_n is called the **complement of G**, it is denoted by G.

Thus, the complement $\overline{G}$ of a simple graph G with n vertices is that graph which is obtained by deleting those edges of Kn which belongs to G. Thus $G = K_n - G = K_n \Delta$~~G~~.

Evidently K_n, G and $\overline{G}$ have the same vertex set and two vertices are adjacent in $\overline{G}$ if and only if they are not adjacent in G. Obviously, $\overline{G}$ is also a simple graph and the complement of $\overline{G}$ is G that is $\overline{\overline{G}}$ **= G**

In fig 1.26(a), the complete graph K_4 is shown. A simple graph G of order 4 is shown in fig 1.26(b). The complement $\overline{G}$, of G is shown in fig 1.26(c).

Observe that G, $\overline{G}$ & K_4 have the same vertices and that the edges of $\overline{G}$ are got by deleting those edges from K_4 which belong to G.

Figure 1.26 a, b & c

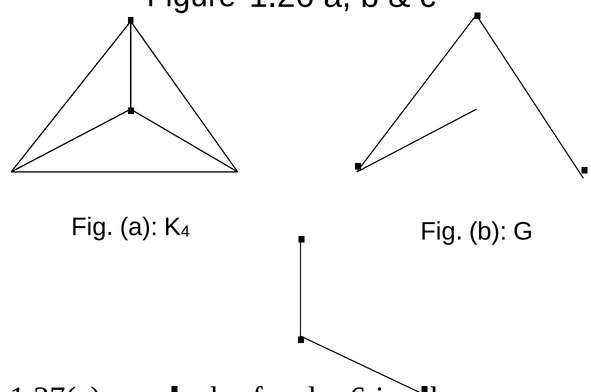


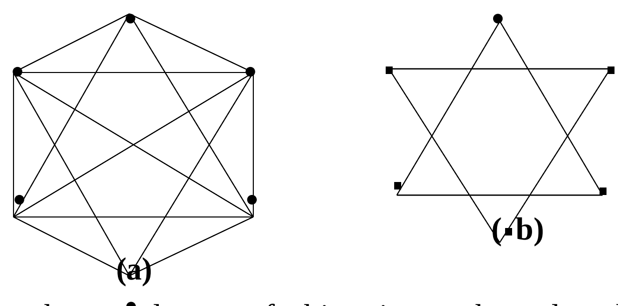
Fig. (a): K_4

Fig. (b): G

Fig. (c): $\overline{G}$

In fig1.27(a) ,a graph of order 6 is shown as a subgraph of K_6 ,the edges of G being shown in thick lines .Its complement $\overline{G}$, is shown in fig1.27(b).The graph shown in fig1.27(b) is known as **<u>David Graph</u>**.

Fig. 1.27

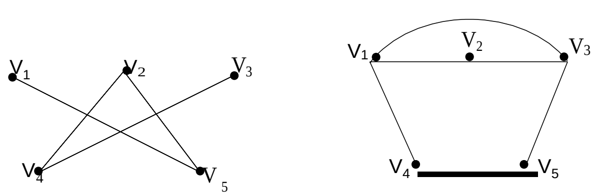


(a) **(b)**

Example 1.Show that the complement of a bipartite graph need not be a bipartite graph.

Solution: Fig 1.28(a) shows a bipartite graph which is of order **5.**The complement of this graph is shown in fig1.28(b),this is not a bipartite graph.

Fig. 1.28(a) Fig. 1.28(b)



Example 2.Let G be a simple graph of order n.If the size of graph G is 56 and size of is G 80.What is ~~n?~~

Solution: We know that $\overline{G}$ =K_n-G therefore

Size of $\overline{G}$ = (Size of K_n)- (Size of G)

Since size of K_n (ie the number of edges in K_n) is ½(n)(n-1),this yields

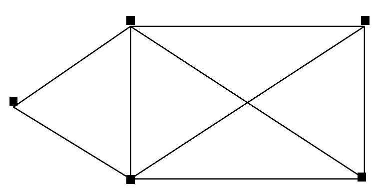
80 = ½ n(n-1) – 56

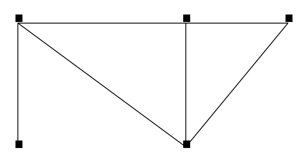
or n(n-1) = 160 + 112 = 272 = 17 x 16

thus, n = 17, (that is, G is of order 17)

Example 3: Find the union, intersection and the ring sum of the graph G1 and G2 shown below.

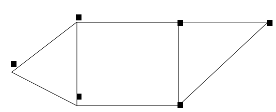
Fig. 1.29(G) Fig. 1.29($\overline{G}$)



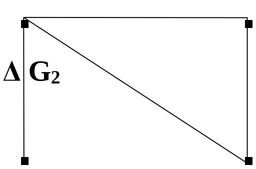


Solution :

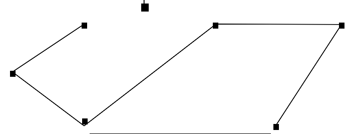
Union :- **G_1 U G_2**



Intersection : - **G_1 n G_2**



Ring Sum :- **$G_1 \Delta G_2$**



Example 4: For the graph G shown below, find G-v and G-e .

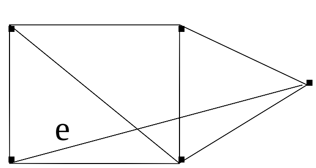
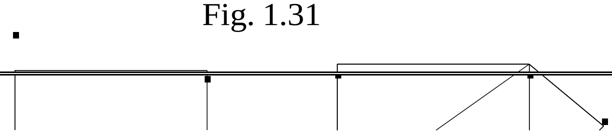


Fig 1.30

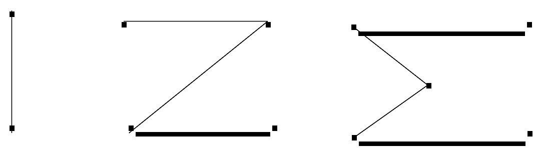
Solution :

Fig. 1.31



Example 5: Find the complement of each of the following simple graphs

Fig. 1.32



(a) (b) (c)

Example 6: Find the complement of the complete bipartite graph $K_{3,3}$

Solution :

Fig. 1.34

WALKS AND THEIR CLASSIFICATION

WALK:

Let G be a graph having atleast one edge. In G, consider a finite, alternating sequence of vertices and edges of the form v_i e_j v_{i+1} e_{j+1} v_{i+2},…..,e_k v_m which begin and ends with vertices and which is such that each edge in the sequence is incident on the vertices preceding and following it in the sequence. Such a sequence is called a **walk** in G. In a walk, a vertex or an edge (or both) can appear more than once.

The number of edges present in a walk is called its '**length**'.

For example : Consider the graph shown below;

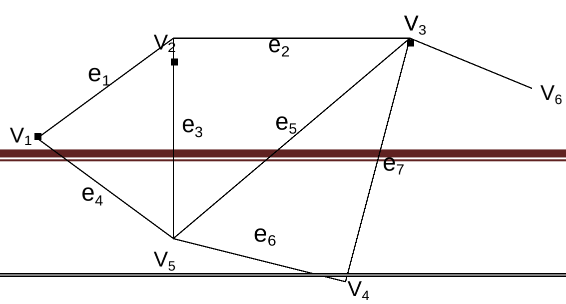


Fig.1.35

In this graph,

i) The sequence v_1e_1 v_2c_2 v_3,e_8v_6 is a walk of length 3 (because this walk contains 3 edges; e_1,e_2,e_8). In this walk, no vertex and no edge is repeated.

ii) The sequence v_1,e_4 v_5e_3 $v_2c_2v_3e_5$ $v_5e_6v_4$ is a walk of length 5. In this walk, the vertex v5 is repeated; but no edge is repeated.

iii) The sequence $v_1e_1v_2e_3v_5e_3v_2e_2v_3$ is a walk of length 4. In this walk, the edge e_3 is repeated and the vertex V_2 is repeated

A walk that begins and ends at the same vertex is called a **closed walk**. In other words, a closed walk is a walk in which the terminal vertices are coincident.

A walk which is not closed is called an **open walk**. In other words, an open walk is a walk that begins and ends at two different vertices.

For Example, in the graph shown in figure (1.35) $v_1e_1v_2c_3v_5e_4v_1$ is a **closed walk** and $v_1e_1v_2e_2v_3e_5v_5$ is our **open walk**.

TRAIL AND CIRCUIT:

In a walk, vertices and /or edges may appear more than once, if in an open walk no edge appears more than once, than the walk is called **a trail**. A closed walk in which no edge appears more than once is called a **circuit**.

For example: In fig (1.35), the open walk $v_1e_1v_2e_3v_5e_3v_2e_2v_3$ (shown separately in figure 1.36(a) is not a trail (because, in this walk, the edge e_3 is repeated) where as

Fig. 1.36 (a) :Not a trail

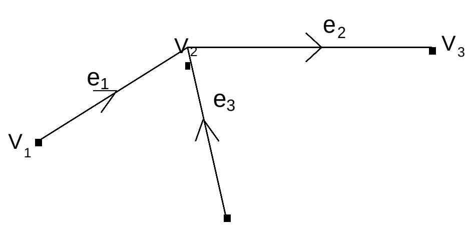
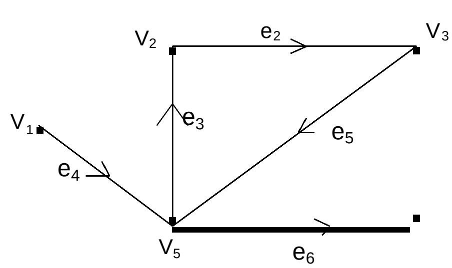


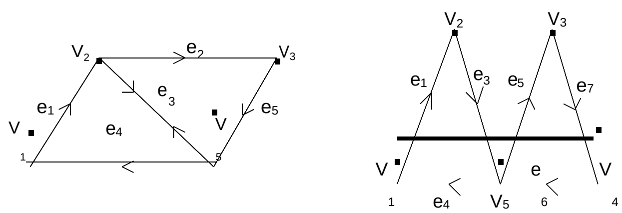
Fig. 1.36 (b): trail



The open walk $v_1e_4v_5e_3v_2v_2v_3e_5v_5e_6v_4$ (shown separately in fig 1.36(b) is trail.

Also, in the same fig (ie., in fig1.35), the closed walk $v_1\ e_1v_2\ e_3\ v_5\ e_3\ v_2\ e_2\ v_3\ e_5\ v_5\ e_4\ v_1$ (shown separately in fig 1.37(a) is not a circuit (because e_3 is repeated) where as the closed walk $v_1e_1v_2e_3v_5e_5v_3e_7v_4e_6v_5e_4v_1$ (shown separately in fig1.37(b)) is a circuit.

Fig. 1.37(a) Fig. 1.37(b)



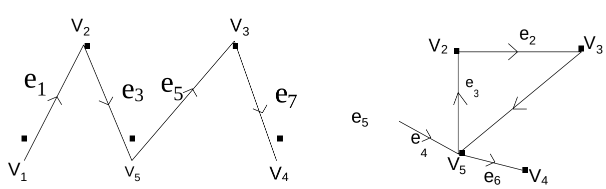
PATH AND CYCLE:

(a) : Not a circuit (b) : Circuit

A trail in which no vertex appears more than once is called a **path**.

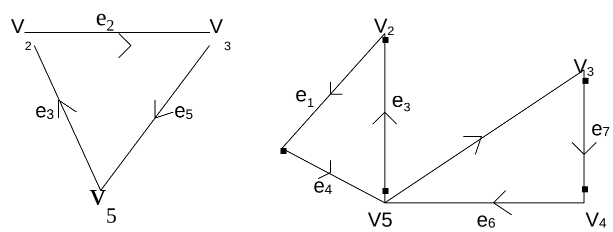
A Circuit in which the terminal vertex does not appear as an internal vertex (also) and no internal vertex is repeated is called a '**cycle**'.

Fig. 1.38



(a) : Path (a) : Not a path

Fig. 1.39



(a) : Cycle **(b) : Not a Cycle**

For example, in figure (1.35), the trail $v_1e_1e_3v_5e_5v_3e_7v_4$ (shown separately in fig 1.38(a)) is a path whole as the trail $v_1e_4v_5e_3v_2e_2e_5v_5e_6v_4$ (shown separately in fig 1.38(b) is not a path (because in this trail, v5 appears twice).

Also, in the same fig, the circuit $v_2e_2v_3e_5v_5e_3v_2$ (shown separately in fig 1.39(a)) is a cycle where as the circuit $v_2e_1v_1e_4v_5e_5v_3e_7v_4e_6v_5e_3v_2$ (shown separately in fig 1.39(b) is not a cycle (because, in this circuit, v5 appears twice)

The following facts are to be emphasized.

1. A walk can be open or closed. In a walk (closed or open), a vertex and / or an edge can appear more than once.
2. A trail is an open walk in which a vertex can appear more than once but an edge cannot appear more than once.
3. A circuit is a closed walk in which a vertex can appear more than once but an edge cannot appear more than once.
4. A path is an open walk in which neither a vertex nor an edge can appear more than once. Every path is a trail; but a trail need not be a path.
5. A cycle is a closed walk in which neither a vertex nor an edge can appear ore than once.

Every cycle is a circuit; **but, a circuit need not be a cycle.**

Example:

For the graph shown in figure1.40 indicate the nature of the following walks.

$v_1e_1v_2e_2v_3e_2v_2$

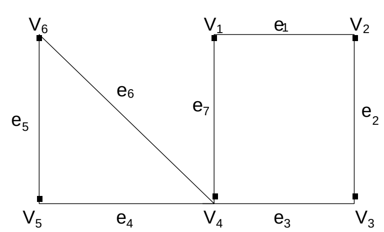
$v_4e_7v_1e_1v_2e_2v_3e_3v_4e_4v_5$

$v_1e_1v_2e_2v_3e_3v_4e_4v_5$

$v_1e_1v_2e_2v_3e_3v_4e_7v_1$

$v_6e_5v_5e_4v_4e_3v_3e_2v_2e_1v_1e_7v_4e_6v_6$

Fig. 1.40



Solution:

1. Open walk which is not a trail the edge e_2 is repeated.
2. Trail which is not a path (the vertex v_4 is repeated)
3. Trail which is a path
4. Closed walk which is a cycle.
5. Closed walk which is a circuit but not a cycle (the vertex v_4 is repeated)

EULER CIRCUITS AND EULER TRAILS.

Consider a connected graph G. If there is a circuit in G that contains all the edges of G. Than that circuit is called an **Euler circuit** (or Eulerian line, or Euler tour) in G. If there is a trail in G that contains all the edges of G, than that trail is called an **Euler trail**.

Recall that in a trail and a circuit no edge can appear more than once but a vertex can appear more than once. This property is carried to Euler trails and Euler Circuits also.

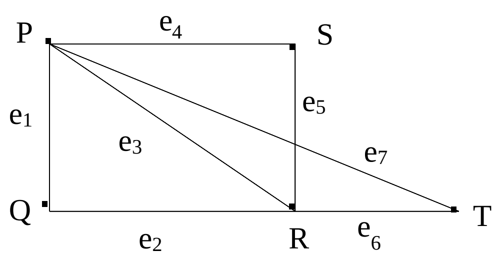
Since Euler circuits and Euler trails include all edge, then automatically should include all vertices as well.

A connected graph that contains an Euler circuit is called a **Semi Euler graph** (or a Semi Eulerian graph).

For Example, in the graph shown in figure 1.41 closed walk.

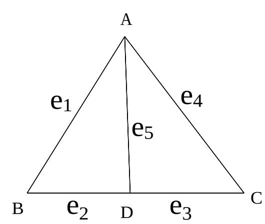
$Pe_1Qe_2Re_3Pe_4Se_5Re_6Te_7P$ is an Euler circuit. Therefore, this graph is a an Euler graph.

Fig 1.41



Consider the graph shown in fig.1.41. We observe that, in this graph, every sequence of edges which starts and ends with the same vertex and which includes all edges will contain at least one repeated edge. Thus, the graph has no Euler circuits. Hence this graph is not an Euler graph.

Fig. 1.42



It may be seen that the trail $Ae_1Be_2De_3Ce_4Ae_5D$ in the graph in fig 1.42 is an Euler trail. This graph therefore a Semi – Euler Graph.

Example 1: Show that the following graph contains an Euler Circuits

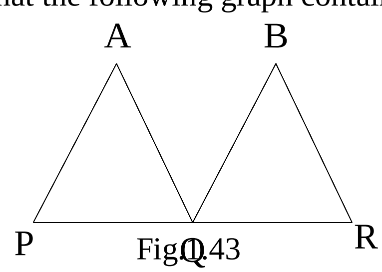
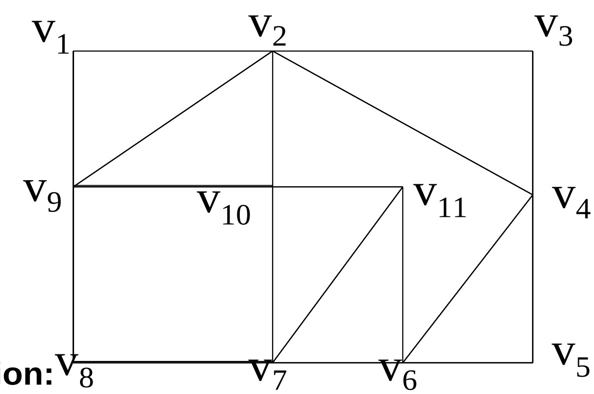


Fig1.43

Solution: The graph containas an Euler Circuit PAQBRQP

Example 2: find an Euler circuit in the graph shown below.

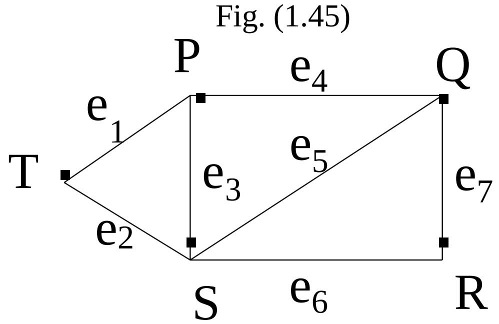
Fig.1.44



Solution:
$V_1V_2V_9V_{10}V_2V_{11}V_7V_{10}V_{11}V_6V_4V_2V_3V_4V_5V_6V_7V_8V_9V_1$

Example 3: show that the following graph contains an Euler trail.

Fig. (1.45)



Solution: the graph contains $Pe_1Te_2Se_3Pe_4Qe_5Se_6Re_7Q$as an **Euler trail**.

ISOMORPHISM :

Consider two graphs G = (V, E) and G' = (V', E') suppose their exists a function f : V → V' such that (i) f is a none to one correspondence and(ii) for all vertices A, B of G {A, B} is an edge of G if and only if { f(A), f(B)} is an edge of G', then f is called as **isomorphism** between G and G', and we say that G and G' are **isomorphic graphs**.

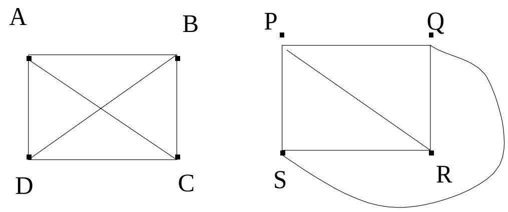
In other words, two graphs G and G' are said to be isomorphic (to each other) if there is a one to one correspondence between their vertices and between their edges such that the adjacency of vertices is preserved such graphs will have the same structures, differing only in the way their vertices and edges are labelled or only in the way they are represented geometrically for any purpose, we regard them as essentially the same graphs.

When G and G' are isomorphic we write $G \approxeq G'$

Where a vertex A of G corresponds to the vertex A' = f(A) of G' under a one to one correspondence f : G $\rightarrow$ G', we write A $\leftrightarrow$ A' Similarly , we write {A, B} $\leftrightarrow$ {A', B'} to mean that the edge AB of G and the edge A'B' of G' correspond to each other, under f.

For example, look at the graphs shown in fig1.46

Fig. 1.46



Consider the following one to one correspondence between between the vertices of these two graphs.

A$\leftrightarrow$ P, B $\leftrightarrow$Q, C $\leftrightarrow$ R, D $\leftrightarrow$ S

Under this correspondence, the edges in two graphs correspond with each other as indicated below:

{A, B} $\leftrightarrow$ {P, Q}, {A, C} $\leftrightarrow$ {P, R}, {A, D} $\leftrightarrow$ {P, S}

{B, C} $\leftrightarrow$ {Q, R}, {B, D} $\leftrightarrow$ {Q, S}, {C, D} $\leftrightarrow$ {R, S},

We check that the above indicated one to cone correspondence between the

Vertices / edges of the two graphs. Preserves the adjacency of the vertices. The existence of this correspondence proves that the two graphs are isomorphic (note that both the graphs represent the complete graph K_4).

Next, consider the graphs shown in figures 1.47 (a) and 1.47(b)

Fig. 1.47

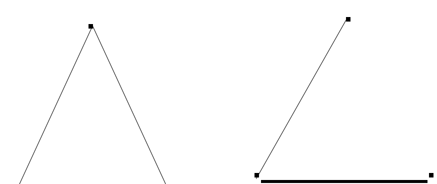


Fig. (a) Fig. (b)

We observe that the two graphs have the same mumber of vertices but different number of edges. Therefore, although there can exist one-to-one correspondence between the vertices, there cannot be a one-to-one correspondence between the edges. The two graphs are therefore not isomorphic.

From the definition of isomorphism of graphs, it follows that if two graphs are isomorphic, then they must have

1. The same number of vertices.

2. The same number of edges.

3. An equal number of vertices with a given degree.

These conditions are necessary but not sufficient. This means that two graphs for which these conditions hold need not be isomorphic.
In particular, two graphs of the same order and the same size need not be isomorphic. To see this, consider the graphs shown in figures 1.48(a) and (b).

Fig.1.48(a)

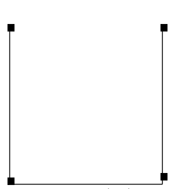


Fig. (a)

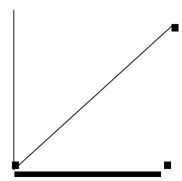


Fig. (b)

We note that both graphs are of order 4 and size 3. But the two graphs are not isomorphic. Observe that there are two pendant vertices in the first graph where as there are three pendant vertices in the second graph. As such, under any one-to-one correspondence between the vertices and the edges of the two graphs, the adjacency of vertices is not preserved

Example 1:
Prove that the two graphs shown below are isomorphic.

Fig.1.49

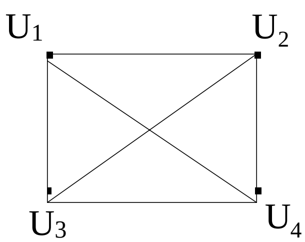


Fig. (a)

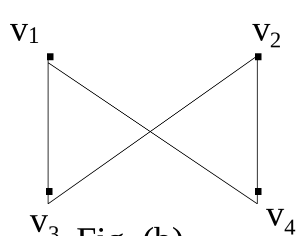


Fig. (b)

Solution: We first observe that both graphs have four vertices and four edges. Consider the following one – to- one correspondence between the vertices of the graphs.
$u_1 \leftrightarrow v_1$, $u_2 \leftrightarrow v_4$, $u_3 \leftrightarrow v_3$, $u_4 \leftrightarrow v_2$.
This correspondence give the following correspondence between the edges.
$\{u_1,u_2\} \leftrightarrow \{v_1,v_4\}$, $\{u_1,u_3\} \leftrightarrow \{v_1,v_3\}$
$\{u_2,u_4\} \leftrightarrow \{v_4,v_2\}$, $\{u_3, u_4\} \leftrightarrow \{v_3, v_2\}$.

These represent one-to-one correspondence between the edges of the two graphs under which the adjacent vartices in the first graph correspond to adjacent vertices in the second graph and vice-versa.

Example 2: Show that the following graphs are not isomorphic.

Fig. 1.50

Fig. (a)

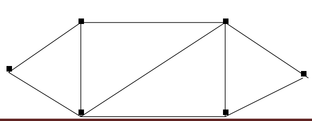
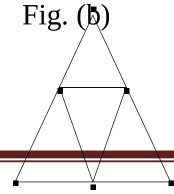


Fig. (b)



Solution: We note that each of the two graphs has 6 vertices and nine edges. But, the first graph has 2 vertices of degree 4 where as the second graph has 3 vertices of degree 4. Therefore, there cannot be anyone-to-one correspondence between the vertices and between the edges of the two graphs which preserves the adjacency of vertices. As such, the two graphs are not isomorphic.

TREES

Graphs

- Graph consists of two sets: set V of vertices and set E of edges.
- Terminology: endpoints of the edge, loop edges, parallel edges, adjacent vertices, isolated vertex, subgraph, bridge edge
- Directed graph (digraph) has each edge as an ordered pair of vertices

Special Graphs

- Simple graph is a graph without loop or parallel edges. A complete graph of n vertices K_n is a simple graph which has an edge between each pair of vertices. A complete bipartite graph of (n, m) vertices $K_{n,m}$ is a simple graph consisting of vertices, v_1, v_2, …, v_m and w_1, w_2, …, w_n with the following properties:
  - There is an edge from each vertex v_i to each vertex w_j
  - There is no edge from any vertex v_i to any vertex v_j
  - There is no edge from any vertex w_i to any vertex w_j

The Concept of Degree

- The degree of a vertex deg(v) is a number of edges that have vertex v as an endpoint. Loop edge gives vertex a degree of 2. In any graph the sum of degrees of all vertices equals twice the number of edges. The total degree of a graph is even. In any graph there are even number of vertices of odd degree

Paths and Circuits

- A walk in a graph is an alternating sequence of adjacent vertices and edges. A path is a walk that does not contain a repeated edge. Simple path is a path that does not contain a repeated vertex. A closed walk is a walk that starts and ends at the same vertex. A circuit is a closed walk that does not contain a repeated edge. A simple circuit is a circuit which does not have a repeated vertex except for the first and last

Connectedness

- Two vertices of a graph are connected when there is a walk between two of them. The graph is called connected when any pair of its vertices is connected. If graph is connected, then any two vertices can be connected by a simple path. If two vertices are part of a circuit and one edge is removed from the circuit then there still exists a path between these two vertices. Graph H is called a connected component of graph G when H is a subgraph of G, H is connected and H is not a subgraph of any bigger connected graph. Any graph is a union of connected components

Euler Circuit

- Euler circuit is a circuit that contains every vertex and every edge of a graph. Every edge is traversed exactly once. If a graph has Euler circuit then every vertex has even degree. If some vertex of a graph has odd degree then the graph does not have an Euler circuit. If every vertex of a graph has even degree and the graph is connected then the graph has an Euler circuit. A Euler path is a path between two vertices that contains all vertices and traverces all edge exactly ones. There is an Euler path between two vertices v and w iff vertices v and w have odd degrees and all other vertices have even degrees

Hamiltonian Circuit

Hamiltonian circuit is a simple circuit that contains all vertices of the graph (and each exactly once). Example: Traveling salesperson problem

Trees

- Connected graph without circuits is called a tree. Graph is called a forest when it does not have circuits. A vertex of degree 1 is called a terminal vertex or a leaf, the other vertices are called internal nodes. Examples: Decision tree, Syntactic derivation tree.
- Any tree with more than one vertex has at least one vertex of degree 1. Any tree with n vertices has n – 1 edges. If a connected graph with n vertices has n – 1 edges, then it is a tree

Rooted Trees

- Rooted tree is a tree in which one vertex is distinguished and called a root. Level of a vertex is the number of edges between the vertex and the root. The height of a rooted tree is the maximum level of any vertex. Children, siblings and parent vertices in a rooted tree. Ancestor, descendant relationship between vertices

Binary Trees

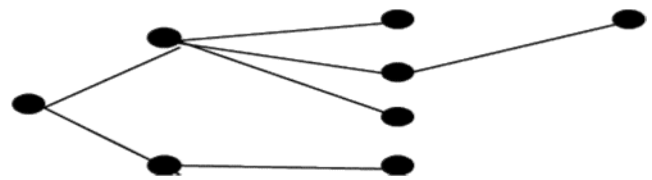
- Binary tree is a rooted tree where each internal vertex has at most two children: left and right. Left and right subtrees.
- Full binary tree: Representation of algebraic expressions
- If T is a full binary tree with k internal vertices then T has a total of 2k + 1 vertices and k + 1 of them are leaves. Any binary tree with t leaves and height h satisfies the following inequality: $t \leq 2^h$

Spanning Trees

- A subgraph T of a graph G is called a spanning tree when T is a tree and contains all vertices of G. Every connected graph has a spanning tree. Any two spanning trees have the same number of edges. A weighted graph is a graph in which each edge has an associated real number weight. A minimal spanning tree (MST) is a spanning tree with the least total weight of its edges.

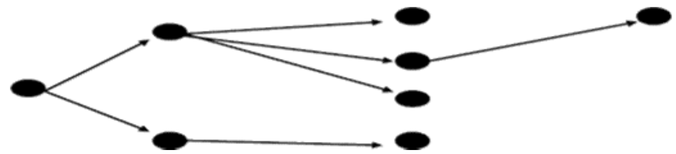
Trees: Definition & Applications

A tree is a connected graph with no cycles. A forest is a graph whose components are trees. An example appears below. Trees come up in many contexts: tournament brackets, family trees, organizational charts, and decision trees, being a few examples.



Directed Trees

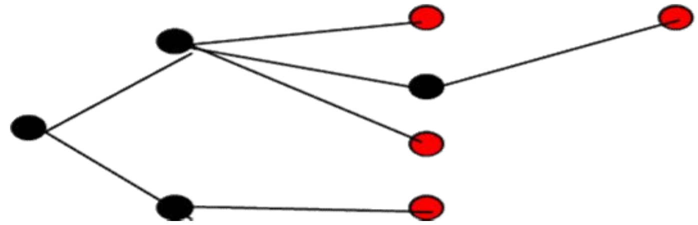
A directed tree is a digraph whose underlying graph is a tree and which has no loops and no pairs of vertices joined in both directions. These last two conditions mean that if we interpret a directed tree as a relation, it is irreflexive and asymmetric. Here is an example.



Theorem: A tree T(V,E) with finite vertex set and at least one edge has at least two leaves (a leaf is a vertex with degree one). Proof: Fix a vertex a that is the endpoint of some edge. Move from a to the adjacent vertex along the edge. If that vertex has no adjacent vertices then it has degree one, so stop. If not, move along another edge to another vertex. Continue building a path in this fashion until you reach a vertex with no adjacent vertices besides the one you just came from. This is sure to happen because V is finite and you never use the same vertex twice in the path (since T is a tree). This produces one leaf. Now return to a. If it is a leaf, then you are done. If not, move along a different edge than the one at the first step above. Continue extending the path in that direction until you reach a leaf (which is sure to happen by the argument above).

Trees: Leaves & Internal Vertices

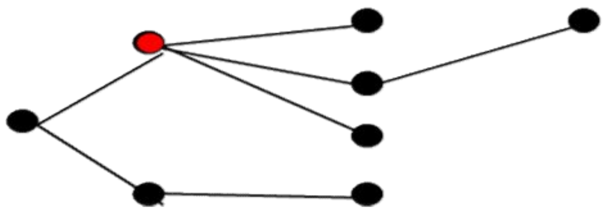
In the following tree the red vertices are leaves. We now know every finite tree with an edge has a least two leaves. The other vertices are internal vertices.

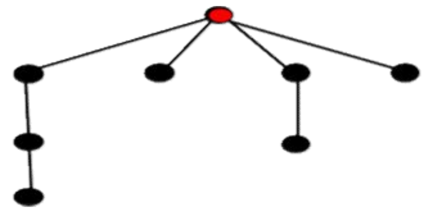


- Theorem: Given vertices a and b in a tree T(V,E), there is a unique simple path from a to b. Proof: Trees are connected, so there is a simple path from a to b. The book gives a nice example of using the contrapositive to prove the rest of the theorem.
- Theorem: Given a graph G(V,E) such that every pair of vertices is joined by a unique simple path, then G is a tree. This is the converse of Theorem 6.37. Proof: Since a simple path joins every pair of points, the graph is connected. Now suppose G has a cycle abc…a. Then ba and bc…a are distinct simple paths from b to a. This contradicts uniqueness of simple paths, so G cannot possess such a cycle. This makes G a tree.

Rooted Trees

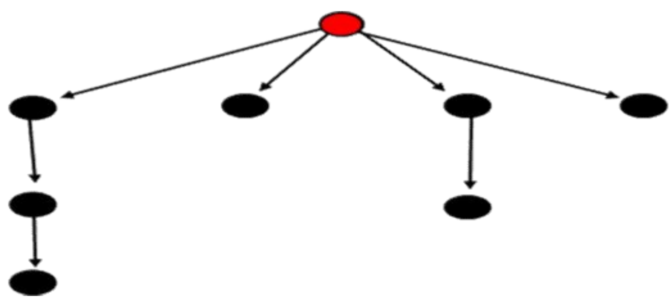
Sometimes it is useful to distinguish one vertex of a tree and call it the root of the tree. For instance we might, for whatever reasons, take the tree above and declare the red vertex to be its root. In that case we often redraw the tree to let it all "hang down" from the root (or invert this picture so that it all "grows up" from the root, which suits the metaphor better)





Rooted Directed Trees

It is sometimes useful to turn a rooted tree into a rooted directed tree T′ by directing every edge away from the root.



Rooted trees and their derived rooted directed trees have some useful terminology, much of which is suggested by family trees. The level of a vertex is the length of the path from it to the root. The height of the tree is the length of the longest path from a leaf to the root. If there is a directed edge in T′ from a to b, then a is the parent of b and b is a child of a. If there are directed edges in T′ from a to b and c, then b and c are siblings. If there is a directed path from a to b, then a is an ancestor of b and b is a descendant of a.

Binary & m-ary Trees

We describe a directed tree as binary if no vertex has outdegree over 2. It is more common to call a tree binary if no vertex has degree over 3. (In general a tree is m-ary if no vertex has degree over m+1. Our book calls a directed tree m-ary if no vertex has outdegree over m.) The directed rooted tree above is 4-ary (I think the word is quaternary) since it has a vertex with outdegree 4. In a rooted binary tree (hanging down or growing up) one can describe each child vertex as the left child or right child of its parent.

Trees: Edges in a Tree

Theorem: A tree on n vertices has n–1 edges. Proof: Let T be a tree with n vertices. Make it rooted. Then every edge establishes a parent-child relationship between two vertices. Every child has exactly one parent, and every vertex except the root is a child. Therefore there is exactly one edge for each vertex but one. This means there are n–1 edges.

Theorem: If G(V,E) is a connected graph with n vertices and n–1 edges is a tree.
Proof: Suppose G is as in the statement of the theorem, and suppose G has a cycle. Then we can remove an edge from the cycle without disconnecting G (see the next slide for why). If this makes G a tree, then stop. If not, there is still a cycle, so we can remove another edge without disconnecting G. Continue the process until the remaining graph is a tree. It still has n vertices, so it has n–1 edges by a prior theorem. This is a contradiction since G had n–1 vertices to start with. Therefore G has no cycle and is thus a tree.

(Why can we remove an edge from a cycle without disconnecting the graph? Let a and b be vertices. There is a simple path from a to b. If the path involves no edges in the cycle, then the path from a to be is unchanged. If it involves edges in the cycle, let x and y be the first and last vertices in the cycle that are part of the path from a to b. So there is a path from a to x and a path from y to b. Since x and y are part of a cycle, there are at least simple two paths from x to y. If we remove an edge from the cycle, at least one of the paths still remains. Thus there is still a simple path from a to b.)

Important Concepts, Formulas, and t heorems

1. Graph. A graph consists of a set of vertices and a set of edges with the property that each edgehas two (not necessarily different) vertices associated with it and called its endpoints.

2. Edge; Adjacent. We say an edge in a graph joins its endpoints, and we say two endpoints are adjacent if they are joined by an edge.

3. Incident. When a vertex is an endpoint of an edge, we say the edge and the vertex are incident.

4. Drawing of a Graph. To draw a graph, we draw a point in the plane for each vertex, and then for each edge we draw a (possibly curved) line between the points that correspond to the endpoints of the edge. Lines that correspond to edges may only touch the vertices that are their endpoints.

5. Simple Graph. A simple graph is one that has at most one edge joining each pair of distinct vertices, and no edges joining a vertex to itself.

6. Length, Distance. The length of a path is the number of edges. The distance between two vertices in a graph is the length of a shortest path between them.

7. Loop; Multiple Edges. An edge that joins a vertex to itself is called a loop and we say we have multiple edges between vertices x and y if there is more than one edge joining x and y.

8. Notation for a Graph. We use the phrase “Let G = (V, E)” as a shorthand for “Let G stand for a graph with vertex set V and edge set E.”

9. Notation for Edges. In a simple graph we use the notation {x, y} for an edge from x to y. In any graph, when we want to use a letter to denote an edge we use a Greek letter like 6 so that we can save e to stand for the number of edges of the graph.

10. Complete Graph on fl vertices. A complete graph on fl vertices is a graph with fl vertices that has an edge between each two of the vertices. We use K_m to stand for a complete graph on fl vertices.

11. Path. We call an alternating sequence of vertices and edges in a graph a path if it starts and ends with a vertex, and each edge joins the vertex before it in the sequence to the vertex after it in the sequence.

12. Simple Path. A path is called a simple path if it has no repeated vertices or edges.

13. Degree of a Vertex. The degree of a vertex in a graph is the number of times it is incident with edges of the graph; that is, the degree of a vertex x is the number of edges from x to other vertices plus twice the number of loops at vertex x.

14. Sum of Degrees of Vertices. The sum of the degrees of the vertices in a graph with a finite number of edges is twice the number of edges.

15. Connected. A graph is called connected if there is a path between each two vertices of the graph. We say two vertices are connected if there is a path between them, so a graph is connected if each two of its vertices are connected. The relationship of being connected is an equivalence relation on the vertices of a graph.

16.16. Connected Component. If C is a subset of the vertex set of a graph, we use E(C) to stand for the set of all edges both of whose endpoints are in C. The graph consisting of an equivalence class C of the connectivity relation together with the edges E(C) is called a connected component of our original graph.

17.17. Closed Path. A path that starts and ends at the same vertex is called a closed path.

18.18. Cycle. A closed path with at least one edge is called a cycle if, except for the last vertex, all of its vertices are different.

19. Tree. A connected graph with no cycles is called a tree.

Important Properties of Trees.

0

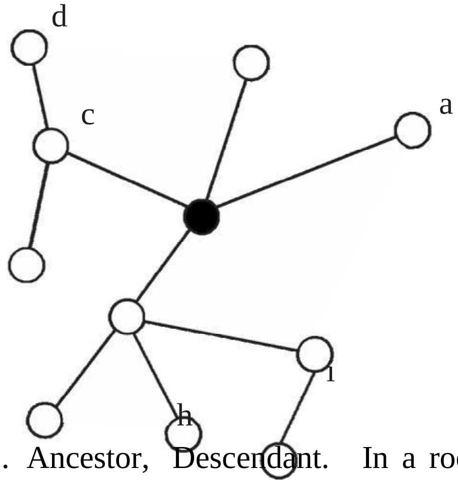
.

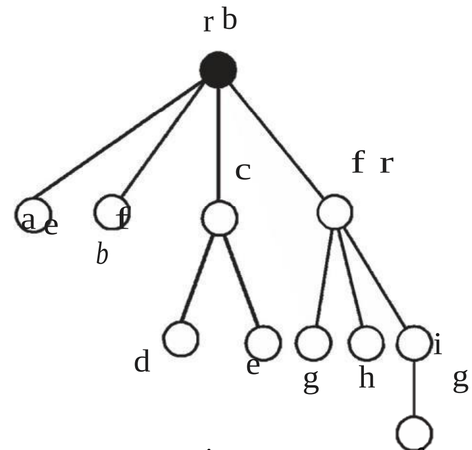
(a) There is a unique path between each two vertices in a tree. (b) A tree on V vertices has V — 1 edges.

(c) Every finite tree with at least two vertices has a vertex of degree one.

Rooted trees

A rooted tree consists of a tree with a selected vertex, called a root, in the tree.





1. Ancestor, Descendant. In a rooted tree with root r, a vertex x is an ancestor of a vertex y, and vertex y is a descendant of vertex x if x and y are different and x is on the unique path from the root to y.

2. Parent, Child. In a rooted tree with root r, vertex x is a parent of vertex y and y is a child of vertex x in if x is the unique vertex adjacent to y on the unique path from r to y.

3. Leaf (External) Vertex. A vertex with no children in a rooted tree is called a leaf vertex or an external vertex.

4. Internal Vertex. A vertex of a rooted tree that is not a leaf vertex is called an internal vertex.

5. Binary Tree. We recursively describe a binary tree as

- an empty tree (a tree with no vertices), or
- a structure T consisting of a root vertex, a binary tree called the left subtree of the root and

a binary tree called the right subtree of the root. If the left or right subtree is nonempty, it root node is joined by an edge to the root of T.

1. Full Binary Tree. A binary tree is a full binary tree if each vertex has either two nonempty children or two empty children.

2. Recursive Definition of a Rooted Tree. The recursive definition of a rooted tree states that it is either a single vertex, called a root, or a graph consisting of a vertex called a root and a set of disjoint rooted trees, each of which has its root attached by an edge to the original root.