UNIT 2 CONNECTEDNESS

Structure **Page Nos.**

2.0 INTRODUCTION

In the last unit you saw that graphs are often used to represent (that is, model) communication or transportation networks and several other systems such as representation of a molecule in a chemical compound. In a transportation network, it is necessary to know which destinations are connected by a direct route. For example, if air travel is abolished then the people without any seaport cannot go to any other country unless their neighbours provide the initial road passage through their territory. When we use a graph to model this situation, we need to see which vertices are connected. We also need to ensure that there is an edge between any two vertices. Such graphs are called connected graphs. In Sec.2.2 we will define connected graphs and we will show that any graph can be partitioned into connected graphs.

In Sec.2.3, we will familiarise you with a type of graph which is useful in electronics and other areas. These graphs are called bipartite graphs. Such graphs are also very useful in studying neural networks.

In Sec. 2.4 we have considered another type of graph, a type, which also represents the chemical compounds butane and isobutane. We call such graphs trees. Here we will show that a tree has got several interesting properties, which are used in studying some real-life situations and various chemical compounds.

2.1 OBJECTIVES

After studying this unit, you should be able to

- distinguish between walks, paths, circuits and cycles in a graph;
- identify the components of various graphs;
- define, and recognize, bipartite graphs, and trees.

2.2 CONNECTED GRAPHS

From Unit 1, you know that graphs model different real-life situations, especially situations involving routes — the vertices represent towns or junctions and each edge represents a road or some other form of communication link. This kind of a picture is very helpful in understanding connected graphs that we introduce in this section. To understand such graphs we need some definitions which describe ways of “going from one vertex to another". We shall first give these definitions in the following subsection.

2.2.1 Paths, Circuits and Cycles

Consider the graph in Fig.1. Imagine yourself walking along its edges, going from vertex to vertex.

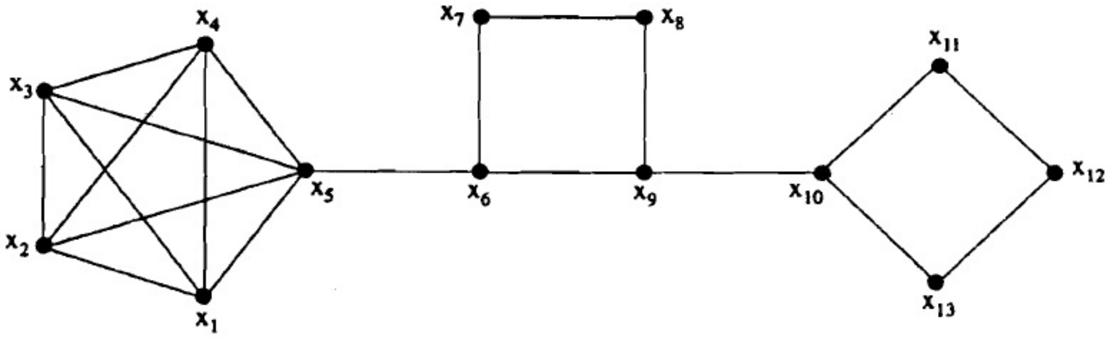


Fig.1

Suppose we want to start at the vertex x_1 and reach the vertex x_{12}. Is this possible? One possible way is to start from vertex x_1, walk along the edge x_1x_2, reach x_2, walk along the edge x_2x_3, reach x_3, walk along x_3x_4, reach x_4, and continue this till we reach x_{12}. Suppose we denote the edge joining x_{i-1} and x_i as $x_{i-1}x_i$. Then we can describe this 'walk' in an alternating sequence of vertices and edges as x_1, x_1x_2, x_2, x_2x_3, x_3, x_3x_4, x_4, x_4x_5, x_5, x_5x_6, x_6, x_6x_9, x_9, x_9x_{10}, x_{10}, $x_{10}x_{11}$, x_{11}, $x_{11}x_{10}$, x_{10}, $x_{10}x_{13}$, x_{13}, $x_{13}x_{12}$, x_{12}. This is by no means the shortest way to reach x_{12} from x_1. We could have gone from x_1 to x_5 directly. Moreover, we passed through the vertex x_{10} twice. This is not necessary. So the walk above can be described as a leisurely walk. If we have more time at our disposal, we can trace and retrace more edges. For example, we could have gone from x_6 to x_9, and again back to x_6.

So what are we doing when choosing a walk? We are, in fact, choosing a sequence whose elements are vertices and edges, alternately. Let us formally define a walk.

Definition: A **walk** in a graph G is a finite sequence $W = \{v_0, e_1, v_1, e_2, \ldots, e_k, v_k\}$, where $v_0, v_1, \ldots, v_k$ are vertices of G and $e_1, e_2, \ldots, e_k$ are edges joining the vertices v_{i-1} and v_i, $1 \le i \le k$. (Note that all the v_is or e_is may not be distinct. There may be repetition.)

In this case we say that **W is a walk from v_0 to v_k**, or **W is a v_0-v_k walk**, or **W is a walk joining v_0 and v_k**. The vertex v_0 is called the **initial vertex** and the vertex v_k is called the **end vertex** of the walk W. The **number of edges contained in a walk,** i.e., k, is called **the length** of the walk **W**, and is denoted by ℓ **(W)**. Since the vertices as well as the edges can be repeated, the length can very well be greater than the number of edges of the graph G.

Note: As you have seen, in a walk the vertices as well as edges can be repeated. So we cannot view this as a subgraph unless all the vertices as well as the edges in the walk are distinct.

Let's consider an example.

Example1 : Consider the graph on 5 vertices and 7 edges given in Fig.2. Find x_1-x_5 walks of length 8 and length 4, respectively.

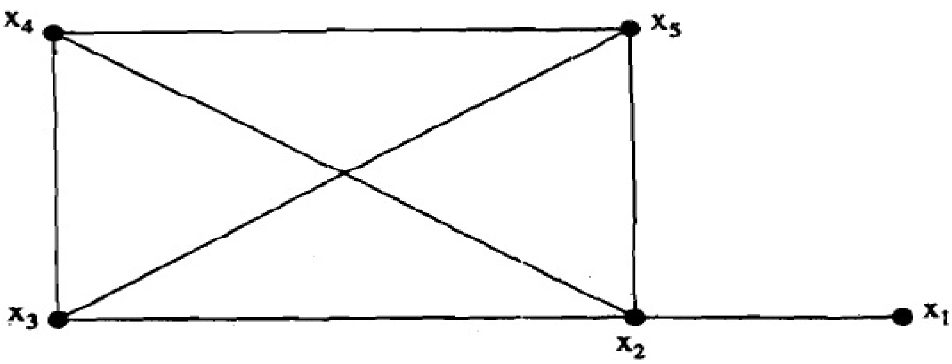


Fig. 2

Solution: Consider the walk
W = {x_1, x_1 x_2, x_2, x_2 x_3, x_3, x_3 x_4, x_4, x_4 x_2, x_2, x_2 x_5, x_5, x_5 x_3, x_3, x_3 x_4, x_4, x_4 x_5, x_5}.
Then W is an x_1-x_5 walk of length 8.

Again, W′ = { x_1, x_1x_2, x_2, x_2x_4, x_4, x_4x_3, x_3, x_3x_5, x_5} is an x_1-x_5 walk of length 4.

***

Why don't you try an exercise now?

E1) For the graph given in Fig.3, find a u-v walk of length 7.

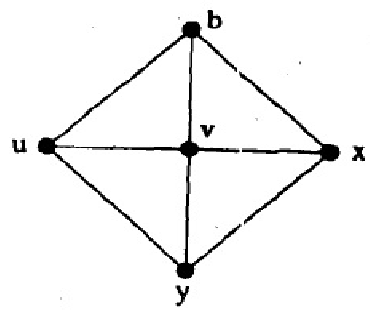


Fig.3

Since we are considering only simple graphs, we often write a walk W as {v_0, v_1, … , v_k}. While doing so, we assume that two consecutive vertices in the walk are joined by an edge in the graph, and that edge is included in the walk. For example, the x_1-x_{12} walk corresponding to Fig.1 that we discussed can be written as
W = {x_1, x_2, x_3, x_4, x_5, x_6, x_9, x_{10}, x_{11}, x_{10}, x_{13}, x_{12}}.

Let us now consider some particular kinds of walks, that we use in studying computer science.

Definitions :

1) A walk W is called a **path** if all its vertices are distinct, and hence, all its edges are distinct.

2) A u-v walk is **closed** if u = v, and **open** if u $\neq$ v.

3) A walk in which all the edges are distinct and the only repeated vertex is the first vertex, this being the same as the last vertex, is called a **cycle**. (Remember, we had introduced you to cycles in Unit 1.)

Let us consider an example.

Example 2 : In the graph in Fig.4, find the following:

i) a closed walk which is not a cycle;
ii) a walk which is not a path;
iii) a cycle.

Solution: i) There are several closed walks in it which are not cycles. For instance,W = {x_5, x_6, x_7, x_8, x_5, x_{11}, x_{12}, x_{13}, x_{14}, x_{15}, x_{11}, x_5} is a closed walk. Since the edge x_5 x_{11} is repeated, it is not a cycle.

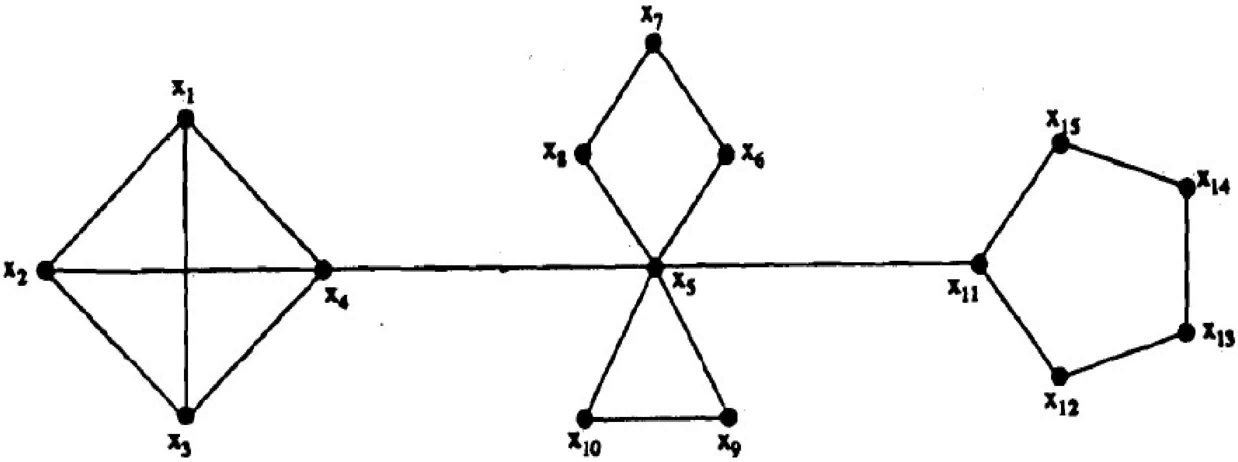


Fig.4

ii) $W_0 = \{x_5, x_6, x_7, x_8, x_5, x_9, x_{10}, x_5\}$ is a walk. Here the vertex x_5 is repeated three times. Thus, this is not a path.

iii) $W_1 = \{x_5, x_6, x_7, x_8, x_5\}$ is a cycle. So is $W_2 = \{x_{11}, x_{12}, x_{13}, x_{14}, x_{15\,2}{}^{x}{}_{11}\}$.

***

Try these exercises now.

E2) If all edges are distinct, then all vertices are distinct. True or false ? Why ?

E3) Let G = (V, E) be a graph, where V = {t, u, v, w, x, y, z} and E = {tu, tv, tw, ux, vw, vy, uz, wx, wz, xy, xz}. In G, find
- i) a u-v walk that is not a path,
- ii) a (u-u) walk that is not a cycle,
- iii) a (u-u) cycle of minimum length.

E4) Let G be a graph such that $\delta(G) \geq k$. Use the principle of induction to show that G has a path of length k starting at any given vertex.
(Recall that $\delta(G) = \min\{d_G(x) : x \in V(G)\}$.)

Now, we know that a walk need not be a path. However, we shall now prove that in any u-v walk, we can always find a path from u to v.

Theorem 1: If W is a u-v walk joining two distinct vertices u and v, then there is a path joining u and v contained in the walk.

Proof : We will prove this using the principle of mathematical induction (see Unit 2, Block 1, MCS-013) on the length of the walk.

Let P (k) denote the statement 'If W is a u-v walk of length k, then there exists a path joining u and v contained in W.'

If k = 1, then P (1) is true since every walk of length 1 is a path.

Now, let us assume that the statement P (k–1) is true. In other words, we assume that given any x-y walk of length $\leq k-1$, there exists a path joining x and y contained in the walk. Then we want to show that the statement P (k) is true.

So, consider the u-v walk $W = \{u = u_0, e_1, u_1, \ldots, e_k, u_k = v\}$ of length k.
If W is already a path, we are done. Otherwise, there is at least one vertex which is repeated. Suppose j is the smallest integer such that the vertex u_j is repeated. Then

there is an integer $t > j$ such that $u_j = u_t$. Now consider the walk W_1 obtained by removing the part $\{e_{j+1}, \ldots, e_t\}$ in W, that is,
$W_1 = \{u = u_0, e_1, \ldots, u_j = u_t, e_{t+1}, \ldots e_k, u_k = v\}$. Clearly, W_1 is a u-v walk contained in the walk W, and its length is $k–t+j<k$, since $j < t$. Hence, by induction, we can get a path P joining u and v contained in W_1. Since P is contained in W_1 and W_1 is contained in W, the path P is contained in the walk W.
So, the result is true for any walk of length k, i.e., P(k) is true.

Therefore, by induction, P(n) is true for all n. Hence the result.

Note that the theorem above does not say that there is a walk joining any two vertices in a graph. In fact, in many practical situations it is very important to know which vertices in a graph can be joined by a walk, **and hence** by a path. For instance, in the graph in Fig.5, there is no a-x walk. Hence there is no path from a to x.

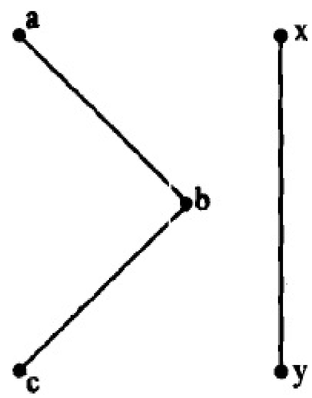


Fig.5

Try an exercise now.

E5) Consider the walk $\{x_3, x_5, x_2, x_4, x_3, x_2, x_1\}$ in the graph given in Fig.2. Obtain two distinct x_3-x_1 paths contained in this walk.

While studying networks, we often need to know whether two vertices in a graph are joined by a walk or not. This leads us to the definition of a connected graph, which we will introduce now.

2.2.2 Components

As you have noticed, almost all the graphs we have discussed so far have been 'in one piece'. Some, like the one in Fig.5, are not. We can formalise this difference by introducing the concept of connectedness.

Definition: A graph G = (V, E) is called **connected** if for any two vertices $u, v \in V$, there exists a u-v walk in G. If G is not connected, then it is called **disconnected**.

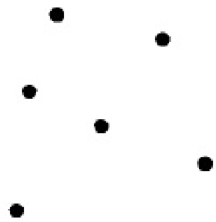


Fig.6 : A null graph

This means that in a connected graph any two distinct vertices are joined by a path (by Theorem 1). For instance, K_n is connected $\forall n \geq 1$. However, a **null graph**, i.e., a graph whose edge set is empty, is totally disconnected (see Fig.6).

Here are some related exercises for you.

E6) Can a graph with one vertex be connected? Give reasons for your answer.

E7) Which of the graphs given in Fig.7 are connected?

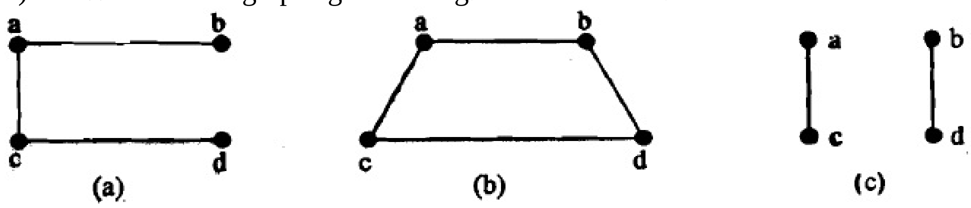


Fig.7

E8) "If a graph G is connected, then all its subgraphs are connected." Prove or disprove this statement.

While solving E8, you would have realised that subgraphs of connected graphs need not be connected. Similarly, some subgraphs of disconnected graphs are connected. Let us discuss such subgraphs now.

Definition: Let G = (V, E) be a graph. A subgraph H of G is called a **component** of G if H is connected and it is not a subgraph of any other connected subgraph of G. Thus, a component of G is, in a sense, a 'maximal' connected subgraph of G.

The number of components of G is denoted by **c(G)**. For instance, the graph in Fig.5 has two components, the graph in Fig.6 has 6 components, and the graph in Fig.1 has one component.

Let us consider another example.

Example 3: Find all the components of the graph G given in Fig.8.

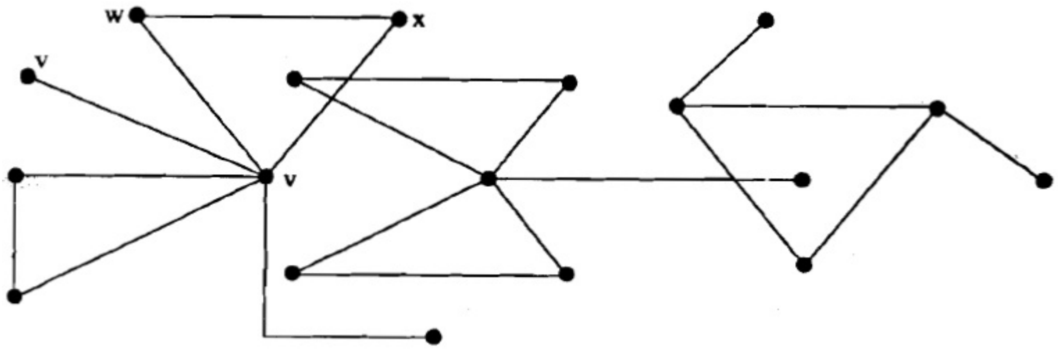


Fig.8

Solution: G has three components, given in Figures 9 (a), (b) and (c).

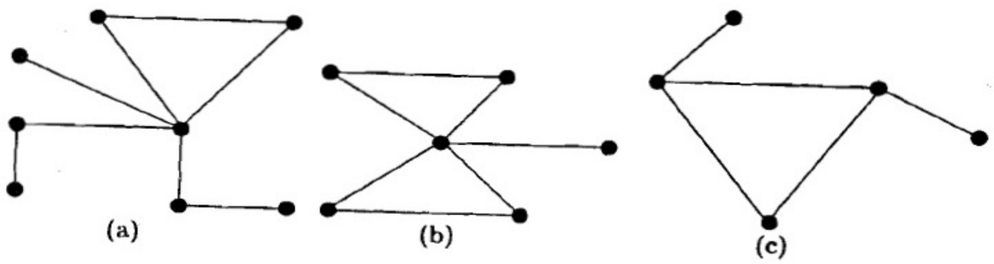


Fig.9

* * *

You can now try this exercise.

E9) Consider the graph G given by Fig. 10. Find
   i) all the connected subgraphs of G;
   ii) all the components of G. Are they disjoint? Give reasons for your answer.

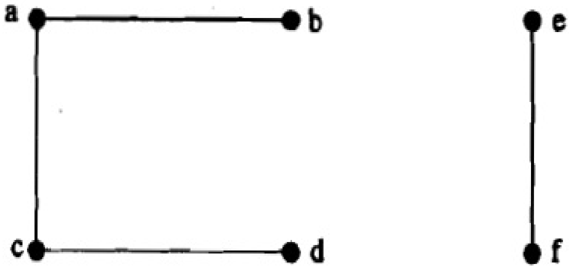


Fig.10

Consider all the graphs you have seen so far in this unit. In each case, it can be written as a disjoint union of its components. This phenomenon generalizes to any graph, as you will see in the following theorem, which we shall only state.

Theorem 2: Every graph can be partitioned into its components.

You may ask how this result can be of help. Knowing the number of components, for instance, helps us to find a bound for the number of edges of a graph. Here is a result about this, again without proof.

Theorem 3: If G is a graph with n vertices and k components, then G can have at least n–k edges, and at most $\frac{1}{2}(n-k)(n-k+1)$ edges.

A very useful result that immediately follows from this is :

Corollary 1: If G is a connected (n, m)-graph, then $n-1 \le m \le \frac{1}{2}n(n-1)$.

Try some exercises now.

E10) Give an example of a graph with
i) 4 components, each of which is complete;
ii) 3 components, where no two components are isomorphic.

E11) Can a graph have more components than vertices ? Give reasons for your answer.

Another approach used in the study of connected graphs is to ask 'how connected' a connected graph is. One possible interpretation of this question is to ask how many edges or vertices must be removed from the graph in order to disconnect it. We shall discuss this in the next subsection.

2.2.3 Connectivity

Let us now consider a graph showing an electric circuit (see Fig.11). This graph is connected. Suppose we cut the wire connecting d and e in the electric circuit. This means that in the graph showing the circuit, we are actually removing the edge de. When we cut the wire, the circuit becomes disconnected. Correspondingly, the removal of the edge de in the graph makes the graph disconnected.

Removing the edge de does not mean that we remove the vertices d and e.

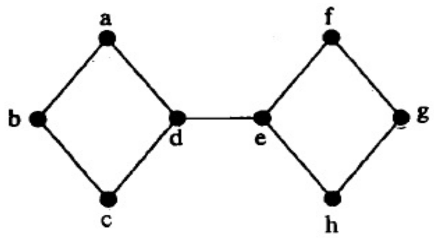


Fig.11

We just saw a situation in which the removal of one edge disconnects the graph. This lead us to the following definition.

Definition: An edge e of a connected graph G is called a **bridge** in G if the removal of e disconnects G. When we remove an edge e from the graph G, we denote the resulting graph by **G–e**.

Not every edge is a bridge. For instance, if we remove the edge ab in Fig. 11, the resulting graph is not disconnected. This also holds for the graph given in Fig. 12, which represents the roads connecting the main towns in a state.

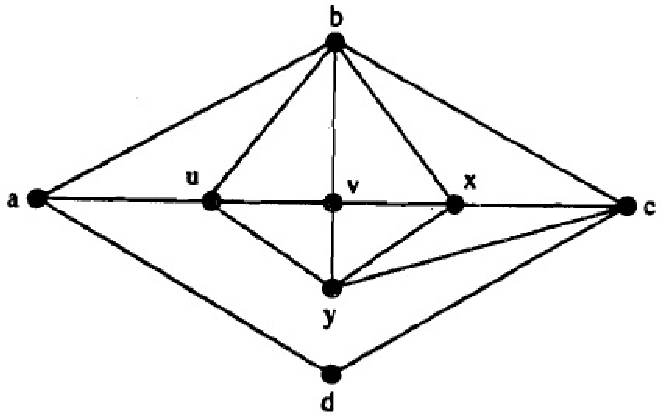


Fig.12

In this case no edge is a bridge since there always exist alternative connections.

Here are some related exercises for you.

E12) Find the bridges in each of the graphs in Fig.13.

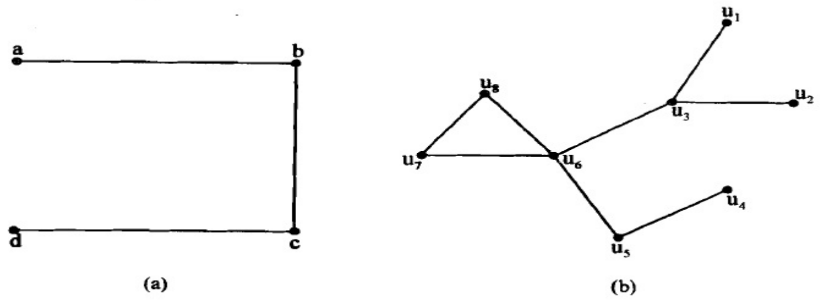


Fig.13

E.13) How many bridges do C_n and K_n have, where $n \geq 3$?

Let us consider the graph given in Fig.11 again. This graph is connected. Here, if we remove the edge de, then the resulting graph gets disconnected, the number of components becoming 2. On the other hand, if we remove the edge dc, then the graph does not get disconnected. Note that the edge dc belongs to the cycle {a, b, c, d, a}, but the edge de does not belong to any such cycle. The cycle seems to provide an alternative connection between the vertices c and d.

In fact, it follows from the definition of a bridge that **an edge e of a graph G is a bridge if and only if e does not belong to any cycle of G**.

While doing E13, you have obtained many graphs which do not have a bridge. To disconnect such a graph we need to remove more than one edge. Therefore, given a graph, it is natural to ask how many edges need to be removed before it gets disconnected. This leads us to the following definition.

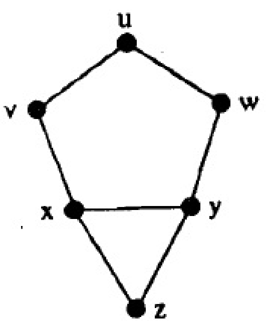


Fig.14

Definition: The **edge-connectivity, λ (G),** of a connected graph G is the least number of edges that need to be removed for G to become disconnected.

For example, the edge-connectivity of any graph with a bridge is 1. Let us consider another kind of example.

Example 4: Find the edge-connectivity of the graph G given in Fig.14.

Solution: First note that this graph does not have any bridges. Therefore its edge-connectivity is more than 1. Now, if we remove the edges xz, zy, then the graph gets disconnected. Similarly, there are other sets of two edges, namely, {xv, vu}and {uw, wy}, the removal of which disconnects G. Therefore, the edge connectivity of G is 2.

***

Note : In the **context of computer networks**, the edge-connectivity of a graph representing such a network gives the number of link failures that can be tolerated before the network becomes disconnected.

Why don't you try some exercises now?

E14) Find the edge-connectivity of C_n and K_n for $n \geq 3$.

E15) Find λ (G), where G is the Petersen graph.

Let us now look at a particular type of set of edges of a connected graph.

Definition : A **cutset** of a connected graph G is a set S of edges with the following properties:

i) the removal of all the edges in S disconnects G;
ii) the removal of any proper subset of S will not disconnect G.

For example, consider the graph given in Fig.15

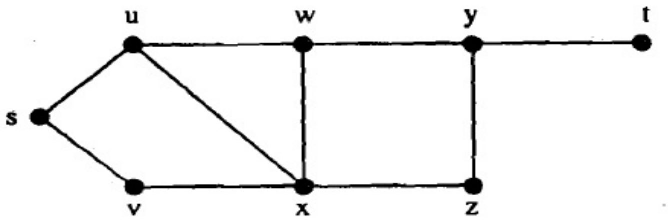


Fig.15

The set {uw, ux, vx} and {uw, wx, xz} are cutsets for this graph. However, the set {uw, wx, xz, yz} is not a cutset since this set has a subset {uw, wx, xz}, the removal of which disconnects G.

Note : 1) **Two cutsets of a graph need not have the same number of edges**. For example, the sets {uw, ux, vx} and {wy, xz} are both cutsets of the graph in Fig.15.

2) **The edge-connectivity of a graph G is the size of the smallest cutset of G**.

Try this exercise now.

E16) Which of the following sets of edges are cutsets of the graph given in Fig.16, and what is the edge-connectivity of the graph ?
i) {su, sv}, ii) {uv, wx, yz}, iii) {ux, vx, wx, yz},
iv) {yt}, v) {wx, xz, yz}, vi) {uw, wx, wy}

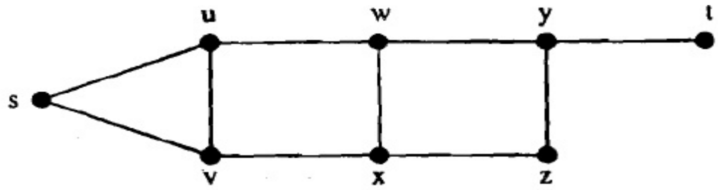


Fig.16

We can also think of connectivity in terms of the minimum number of vertices which need to be removed in order to disconnect a graph. Remember that, when we remove an edge, we do not remove its end vertices. However, **when we remove a**

vertex, then any edge incident with that vertex also gets removed.

Analogous to the notion of a bridge, we define a cut-vertex.

Definition : A **cut-vertex** of a connected graph G is a vertex v of G such that G–v is disconnected.

For instance, in Fig.11, both d and e are cut-vertices.

Now we can define vertex-connectivity and a vertex-cutset on similar lines as we have done for edges. Why don't you try it for yourself (see E 17)?

E17) Define vertex-connectivity and vertex-cutset of a graph.

E18) Find the vertex-connectivity and a vertex-cutset for the graph given in Fig.16.

Once again, **if a graph represents a computer network, then its vertex connectivity gives the number of node failures that the network can tolerate**.

We shall now introduce you to another type of graph which underlies many computer, and other, applications.

2.3 BIPARTITE GRAPHS

In this section we shall define bipartite graphs and explain their importance through various problems. Let us first start with the following problem.

Four persons x_1, x_2, x_3 and x_4 are available to fill five jobs y_1, y_2, y_3, y_4 and y_5. x_1 is qualified for the jobs y_1 and y_2; x_2 is qualified for the jobs y_1 and y_3; x_3 is qualified for the job y_4; and x_4 is qualified for the jobs y_2, y_3 and y_5. The assignment problem is concerned with the following questions:

i) *Can each person be assigned to a single job for which she is qualified?*
ii) *If so, how should the assignment be made?*
iii) *If assigning to a single job is not possible, at most how many jobs should be assigned to each person ?*

The problem of the kind stated above is known as an assignment problem. To solve this problem it is convenient to consider the following graph-theoretic model of the situation (see Fig.17).

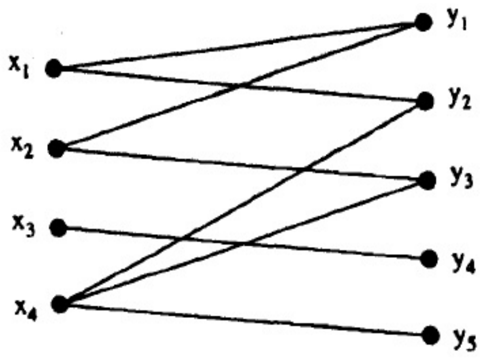


Fig.17

The graph G, representing the problem, has an edge joining x_i and y_j if x_i is qualified for the job y_j. Then the problem of assigning people to jobs for which they are qualified is equivalent to the problem of selecting a subset of the set of edges such that each x will be connected to exactly one y by one of these edges.

Now, if you look at the graph given in Fig.17, you will see that the set of its vertices can be divided into two disjoint subsets such that no two vertices in a subset are adjacent. Let us formally define such graphs.

Definition: A graph G is said to be **bipartite** if V (G) = X $\cup$ Y, where X and Y are non-empty sets such that X $\cap$ Y = ϕ and every edge in E (G) has one end vertex in the set X and the other end vertex in the set Y. The sets X and Y form a **partition** of the set V (G), and we often say that X $\cup$ Y is a **bipartition** of the graph G. We also denote such a graph by **G(X,Y)**.

An alternative way of thinking of a bipartite graph is in terms of colouring its vertices with two colours, say red and blue — a graph is bipartite if we can colour each vertex red or blue in such a way that every edge has a red end and a blue end.

Bipartite graphs are useful in studying various real-life problems, including neural networks. One model that emulates the essential working of the network using graph theory is given in Fig.18. As you can see, this is a bipartite graph, so that the properties of bipartite graphs are useful for studying this model.

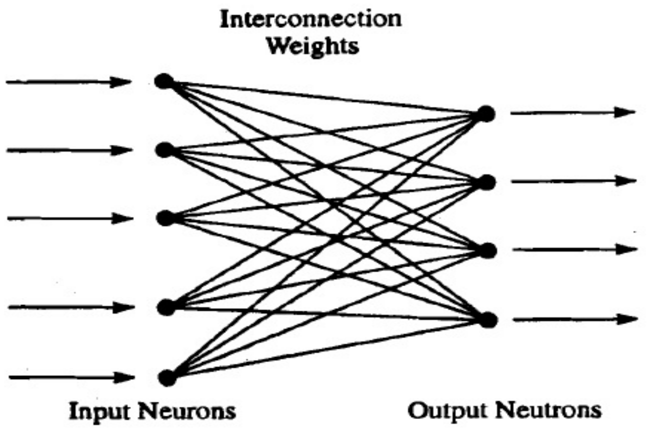


Fig.18

Let us consider some other examples of bipartite graphs.

Example 5 : Show that C_6 is bipartite and K_3 is not bipartite.

Solution : In Fig.19 we show C_6. Since V(C_6) can be partitioned into {a, c, e} and {b, d, f}, C_6 is bipartite.

In K_3 each vertex is adjacent to every other vertex. Therefore, no bipartition is possible.

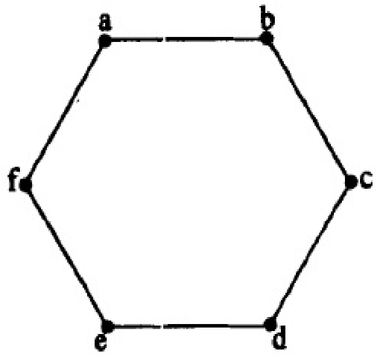
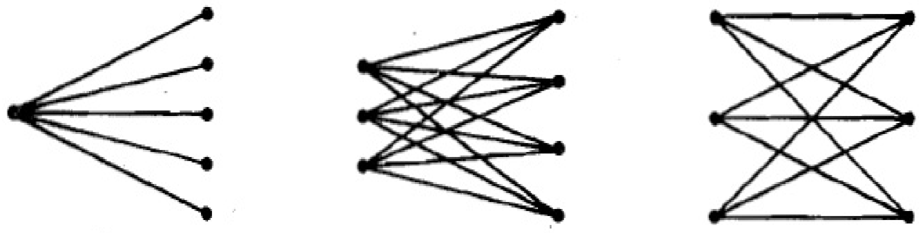


Fig.19 : C_6 is bipartite

***

Note that in a bipartite graph G(X,Y), it is not necessary that each vertex of X is joined to each vertex of Y. For instance, in C_6 in Fig.19, a is not joined to d. This leads us to the following definition.

Definition : A **complete bipartite graph** is a bipartite graph G(X,Y) in which each $x \in X$ is joined to every $y \in Y$, i.e., G is also a complete graph. If $|X| = r$ and $|Y| = s$, we denote G(X,Y) by $\mathbf{K_{r,s}}$. In Fig.20 we have shown a few of these graphs.



$K_{1,5}$ $K_{3,4}$ $K_{3,3}$

Fig 20 : Some complete bipartite graphs

Now, given a bipartite graph, you may wonder if the bipartition is unique. The following example will give you an answer to this question.

Example 5: Find two different bipartitions of the graph given in Fig.21.

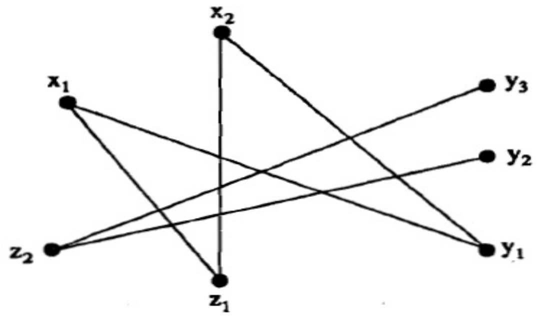


Fig.21

Solution: The vertex set is $\{x_1, x_2, y_1, y_2, y_3, z_1, z_2\}$.
One way of partitioning this can be by taking $X = \{x_1, x_2, z_2\}$, $Y = \{z_1, y_1, y_2, y_3\}$.
Another way can be $X_1 = \{x_1, x_2, y_3\}$, $Y_1 = \{z_2, z_1, y_1, y_2\}$.
Both these partitions make G bipartite.

***

Note that the graph in Fig.21 is not connected. Had it been connected it would not have been possible to find more than one bipartition of G. In fact, we have the following theorem.

Theorem 4 : A connected bipartite graph has a unique bipartition.

We shall now state a theorem, which gives a characterisation for bipartite graphs.

Theorem 5: A graph G is bipartite if and only if G does not contain any cycle of odd length as a subgraph.

This result is very useful. For instance, using it we know that **C_n is not bipartite whenever n is odd**.

You can try some exercises now.

E19) Check whether the hypercube Q_3 and the star graphs are bipartite.

E20) For which values of m and n is $K_{m,n}$ regular ?

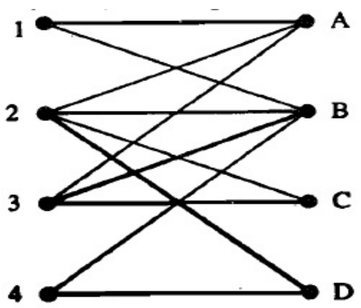
E21) i) Is the subgraph of a bipartite graph bipartite ?
ii) Is the complement of a bipartite graph bipartite ?
Give reasons for your answers.

E22) Show that if $G_1, \ldots, G_n$ are bipartite, then $\bigcup_{i=1}^{n} G_i$ is bipartite.

Let us now go back to the assignment problem. In that problem we are interested in finding those special subgraphs of the associated bipartite graph which give a solution to the problem. We have defined such subgraphs below.

Definition: A **matching** in a bipartite graph G is a set of edges such that no two edges have a common end vertex. In other words, a matching in G (X,Y) defines a one-to-one correspondence between the vertices in a subset of X and the vertices in a subset of Y.

For example, Fig.22 shows a bipartite graph and one of its matchings. Can you find any other matching? We leave this as an exercise for you to check (see E 23).



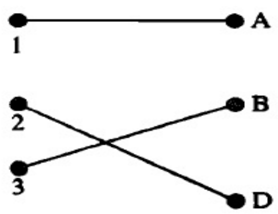


Fig.22 : a) G is bipartite; b) A matching in G

Related to the concept of matching, we have another concept.

Definition: A matching of X into Y is called a **complete matching** of X and Y if there is an edge incident with every vertex in X. In other words, a matching is complete if a one-to-one correspondence is defined between all the vertices in X and the vertices in a subset of Y.

Is the matching given in Fig.22 (b) a complete matching? No, because in this matching, the vertex 4 is not included.

In graph-theoretic terminology, the **assignment problem can be stated in the following way**: if G = G (X, Y) is a bipartite graph, when does there exist a complete matching from X to Y in G? So, for a given bipartite graph, we want to know whether there is a complete matching of the set of vertices in X into the set of vertices in Y. The following theorem gives a **necessary and sufficient condition** for the existence of such a matching. As before we shall only state the theorem, omitting the proof.

Theorem 6: Let G = G (X, Y) be a bipartite graph. A complete matching of X into Y exists in G **if and only if** $|A| \le |R(A)|$ for every subset A of X, where $|A|$ denotes the number of elements in A (also called cardinality of A) and R(A) denotes the set of vertices in Y that are adjacent to the vertices in A.

Let us apply this theorem to the assignment problem in the following example.

Example 6 : Verify the conditions of Theorem 6 for the assignment problem given at the beginning of this section (see Fig.17).

Solution : To check the theorem we have to consider all subsets of the vertex set X = $\{x_1,x_2,x_3,x_4\}$, their cardinality, the corresponding sets R(A), and their cardinality. Table 1 gives a list of all the possibilities.

Table 1

A	$\lvert A \rvert$	R(A)	$\lvert R(A) \rvert$
ϕ	0	ϕ	0
$\{x_1\}$	1	$\{y_1,y_2\}$	2
$\{x_2\}$	1	$\{y_2,y_3\}$	2
$\{x_3\}$	1	$\{y_4\}$	1
$\{x_4\}$	1	$\{y_2,y_3,y_5\}$	3
$\{x_1,x_2\}$	2	$\{y_1,y_2,y_3\}$	2
$\{x_2,x_3\}$	2	$\{y_1,y_3,y_4\}$	2
$\{x_3,x_4\}$	2	$\{y_2,y_3,y_4,y_5\}$	4
$\{x_1,x_4\}$	2	$\{y_1,y_2,y_3,y_5\}$	4
$\{x_2,x_4\}$	2	$\{y_1,y_2,y_3,y_5\}$	4
$\{x_1,x_3\}$	2	$\{y_1,y_2,y_4\}$	3
$\{x_1,x_2,x_3\}$	3	$\{y_1,y_2,y_3,y_4\}$	4
$\{x_2,x_3,x_4\}$	3	$\{y_1,y_2,y_3,y_4,y_5\}$	5
$\{x_1,x_3,x_4\}$	3	$\{y_1,y_2,y_3,y_4\}$	4
$\{x_1,x_2,x_4\}$	3	$\{y_1,y_2,y_3,y_5\}$	4
$\{x_1,x_2,x_3,x_4\}$	4	$\{y_1,y_2,y_3,y_4,y_5\}$	5

It shows that the condition $\lvert A \rvert \le \lvert R(A) \rvert$ is satisfied for all subsets A of X. Hence the condition of Theorem 6 is satisfied. This shows that there exists a complete matching from X into Y for the assignment problem. Therefore, the assignment problem is solved.

***

You can now try an exercise.

E 23) For the bipartite graph given in Fig.22, find a matching, apart from the given one. Does the graph have a complete matching ? Give reasons for your answer.

Let us now see another type of graph which has come into prominence because of its applications to electrical networks.

2.4 TREES

We are all familiar with the idea of a family tree. The concept of a tree in graph theory first arose in connection with work of a mathematician G. Kirchoff on electric networks in the 1840s, and with the work of another mathematician Cayley on the enumeration of chemical molecules in the 1870s. More recently, trees are used in many areas, ranging from linguistics to computing. For instance, trees are used to study the following problems :

- How should items in a list be stored so that an item can be easily located ?
- How should a set of characters be efficiently coded by bit strings ?

So, let us begin our study of trees by considering the following graphs.

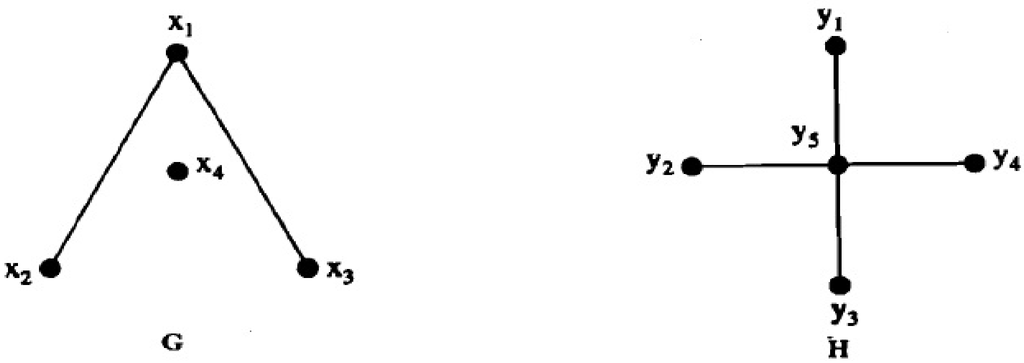


Fig.23

Can you find any difference in their structures ? You might have noticed that G is disconnected, and has no cycles. On the other hand, H is connected and has no cycles. From the following definition you will see that H is an example of a tree.

Definition : A **tree** is a connected graph with no cycles. A **forest** is a graph, each of whose components is a tree.

Fig.24 shows a graph with four components, each of which is a tree. Hence, the graph is a forest.

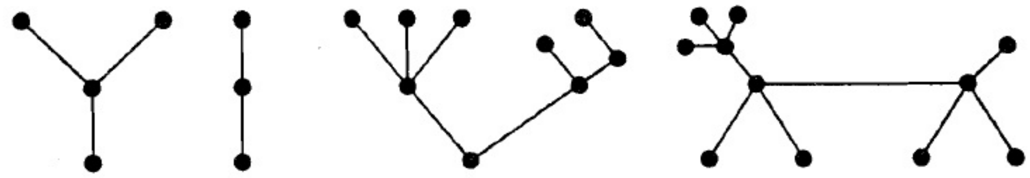


Fig.24

A tree has several defining properties which we shall list in the following theorem.

Theorem 7 : Let G be a graph with n vertices. Then the following statements are equivalent.

i) G is a tree.
ii) G has no cycles and has $(n - 1)$ edges.
iii) G is connected and has $(n - 1)$ edges.
iv) G is connected and every edge is a bridge.
v) Any two vertices of G are connected by exactly one path.

Proof : If $n = 1$, all the five results are trivial. We shall, therefore, assume that $n \geq 2$. Now, from your study of mathematical logic you know that if we prove (i)$\Rightarrow$(ii), (ii) $\Rightarrow$(iii), (iii) $\Rightarrow$(iv), (iv) $\Rightarrow$(v) and (v) $\Rightarrow$(i), then all the statements will be proved to be equivalent. So, let us prove the implications one by one.

(i) $\Rightarrow$(ii) : By definition, G does not have any cycles. We shall show that G has $(n - 1)$ edges, by induction.
If $n = 2$, then the number of edges is 1. Therefore, the result is true for $n = 2$.
So, now let us assume that every tree on p vertices has $(p - 1)$ edges for any positive integer p such that $2 \leq p < n$. Then we have to show that every tree on n vertices has $(n - 1)$ edges.
Now suppose we remove any edge. Since G has no cycles, the removal of any edge disconnects G into two graphs G_1 and G_2, such that G_1 and G_2 are connected and have no cycles. Therefore, G_1 and G_2 are trees and each has less than n vertices.
Let n_1 and n_2 be the vertices in G_1 and G_2. Then $n_1 + n_2 = n$.
Since n_1 and n_2 are less than n, by our induction assumption, the number of edges in G_1 and G_2 are $n_1 - 1$ and $n_2 - 1$, respectively. Therefore, the total number of edges in both the graphs is $n_1 + n_2 - 2 = n - 2$. These edges, together with the edge which is removed, will give the total number of edges in the original graph. Therefore, the total number of edges in G is $n - 1$.

Thus, we have shown that every tree on n vertices has $n - 1$ edges. By induction, this is true for all n.

(ii) $\Rightarrow$(iii) : Suppose that G is disconnected. Let $c(G) = t > 1$. Let $G_1, G_2, \ldots, G_t$ be the components of G such that the number of vertices in each G_i is p_i for $i = 1,2,\ldots,t$, and the number of edges in each G_i is q_i, for $i = 1,2,\ldots,t$. Then
$p = p_1 + p_2 + \ldots + p_t$, $q = q_1 + \ldots + q_t$.

Now, since every G_i is connected and without cycles, G_i is a tree for $i = 1,2,\ldots,t$. Therefore, by what we have shown while proving (i) $\Rightarrow$(ii), $q_i = p_i - 1 \le i \le t$. Then $p - 1 = q = q_1 + \ldots + q_t = p - t$.
That is, $t = 1$, which contradicts our assumption that $t > 1$. Therefore, G is connected.

(iii) $\Rightarrow$(iv) : Suppose there is an edge which is not a bridge. Then the removal of that edge will result in a graph with n vertices and (n–2) edges. This is not possible when G is connected, by Corollary 1 to Theorem 3. Therefore, every edge is a bridge.

(iv) $\Rightarrow$(v) : Since T is connected, each pair of vertices is connected by at least one path. If a given pair of vertices is connected by two paths, then they form a cycle, which contradicts the fact that every edge is a bridge. Therefore, there is a unique path joining any two vertices.

(v) $\Rightarrow$(i) : We are assuming that any two vertices are connected by a unique path. So, the graph G is connected. Now, suppose G contains a cycle $C = \{x_0,x_1,\ldots,x_n = x_0\}$. Then we can find two distinct paths $P_1 = \{x_0,x_1\}$ and $P_2 = \{x_0,x_{n-1},\ldots,x_2,x_1\}$ connecting the vertices x_0 and x_1, which contradicts our assumption. Therefore, G does not contain any cycle, and hence, is a tree.

Why don't you try some exercises now ?

E24) For which values of m and n is $K_{m,n}$ a tree ?

E25) Which of the following graphs are trees, and why ?

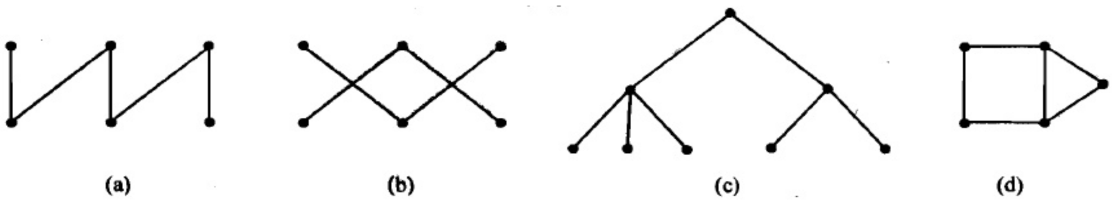


Fig.25

The theorem above tells us that a tree has got several nice properties which a general graph does not have. In fact, the importance of trees in graph theory is that every connected graph contains a tree which has all the vertices of the original graph, as you will now see.

Let us consider a connected graph G. Consider a cycle in it and remove one of its edges such that the resulting graph stays connected. We repeat this procedure with one of the remaining cycles, continuing until there are no cycles left. The graph which remains is a connected subgraph of G which does not have any cycle. Therefore, it is a tree. Note that this tree has all the vertices of G. Such a graph is called a spanning tree, as you will realize from the following definition.

Definition : A **spanning tree** for a graph G is a subgraph of G which contains all the vertices of G and is a tree.

This concept is useful for finding, for example, the minimum number of roads to be kept open to maintain connections in a given transport network.

Now, the question is whether every graph has a spanning tree. The following theorem, the proof of which is omitted, tells us about this.

Theorem 7 : A graph G is connected **if and only** if it has a spanning tree.

The theorem above tells us that in a graph with k components, each component will have a spanning tree. Because of this result and because of the special structure of trees, in trying to prove a general result in graph theory, it is sometimes convenient to try to prove the corresponding result for a tree. The general result would, then, follow.

Note : A **spanning tree is not unique**. For instance, Fig. 26 shows a connected graph G and two of its spanning trees, T_1 and T_2.

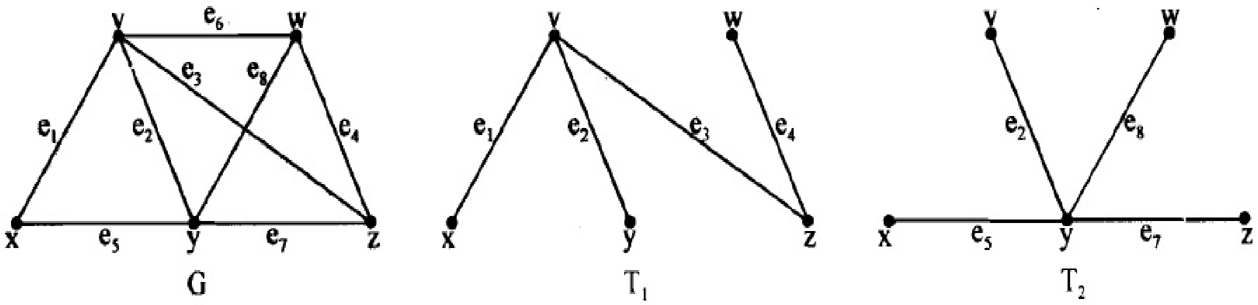


Fig.26

You can try some exercises now.

E26) Draw three spanning trees of the following graph.

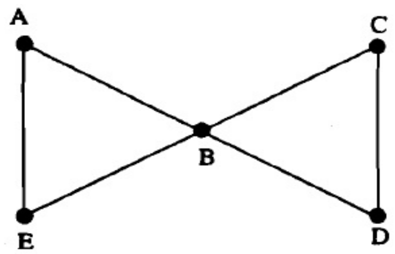


Fig.27

E27) Is a tree a bipartite graph ? Give reasons for your answer.

Spanning trees are important in data networking, particularly in multicasting over Internet Protocol (IP) networks. To send data from a source computer to multiple receiving computers, each of which is a subnetwork, data could be sent separately to each computer. This type of networking, called unicasting, is inefficient, since many copies of the same data are transmitted over the network. To make the transmission of data to multiple receiving computers more efficient, IP multicasting is used. With IP multicasting, a computer sends a single copy of data over the network, and as data reaches intermediate routers the data are forwarded to one or more other routers so that ultimately all receiving computers in their various subnetworks receive these data.

For data to reach receiving computers as quickly as possible, there should be no loops (which in graph theory terminology are circuits or cycles) in the path that data take through the network. That is, once data have reached a particular router, data should never return to this router. To avoid loops, the multicast routers use network algorithms to construct a spanning tree in the graph that has the multicast source, the routers, and the subnetworks containing receiving computers as vertices, with edges representing the links between computers and/or routers.

So far we have seen three types of graphs : connected graphs, bipartite graphs and trees. You will see several other types of graphs in the following units. Let us now summarise what we have covered in this unit.

2.5 SUMMARY

In this unit we have discussed the following points.

1) The definition and use of the terms 'walk', 'path' and 'cycle' in a graph. We have also proved and used the fact that in every u-v walk there is a u-v path.

2) Properties of connected graphs, how to find components of a graph, the effect of removal of a vertex or an edge on the number c(G) of the components of a graph G.

3) The application of the fact that if G is a (p,q)-graph with k components, then $p - k \le q \le \frac{1}{2}(p-k)(p-k+1)$.

4) The definition and properties of bipartite graphs, and a characterisation of such graphs in terms of not containing odd cycles.

5) What a matching is, and when a complete matching exists.

6) Explanation of trees and spanning trees, particularly the importance of such graphs among the class of all connected graphs.

7) The proof and application of the statement that a graph G with n vertices is a tree **iff** it has no cycles and has (n–1) edges **iff** it is connected and has (n–1) edges.

2.6 SOLUTIONS/ANSWERS

E1) {u, uv, ux, x, xy, y, yv, v, vb, b, bu, u, uv, v} is a walk of length 7. This is not the only one. Think of some others too.

E2) False. For instance, in a cycle all the edges are distinct, not all the vertices.

E3) i) It is easy to find examples if you draw the walk. One example is {u,x,w,z,y,x,w,v}. There are other examples.

ii) W = {u,v,y,z,w,x,z,u} is a walk in which the vertex z is repeated. Therefore, W is not a cycle.

iii) W_0 = {u,t,w,x,u} is a cycle such that all other cycles have length greater than $\ell(W_0)$.

E4) We use induction on k. If k = 1, then every vertex has at least one neighbour. Thus, there exists a path of length 1 starting at any vertex.
Now, by induction, assume that in every graph H with $\delta(H) \ge (k-1)$, there is a path of length (k–1) starting at any given vertex.

Let G be a path with $\delta(G) \ge k\,(>1)$. Let x_0 be any vertex in G. Choose any edge e_1 incident on x_0. Consider $G - e_1$. Removal of one edge reduces only the degree of its end vertices by one. Thus, $\delta(G - e_1) \ge (k-1)$. Thus, by induction, there is a path $\{x_1, e_2, \ldots, e_k, x_k\}$ of length (k–1) in G_1. Moreover, since the degree of x_{k-1} is at least k, we can choose x_k different from $x_0, x_1, \ldots, x_{k-2}$. Also $\{x_0, e_1, x_1, e_2, \ldots, x_k\}$ is a path of length k in G starting from x_0. Therefore, there exists a path of length k starting at any vertex in G.

Since this is true for all k, the result follows.

E5) $\{x_3, x_5, x_2, x_1\}$ and $\{x_3, x_2, x_1\}$.

E6) It is connected. Because if it is disconnected then there exists two distinct vertices which are not joined by a path, which is not possible since the graph does not have two distinct vertices.

E7) (a) and (b) are connected, (c) is disconnected.

E8) The statement is false. For example, consider the graph K_3 given in Fig.28(a). The subgraph of this graph obtained by deleting the edges v_1v_3 and v_2v_3, given in Fig.28(b), is not connected.

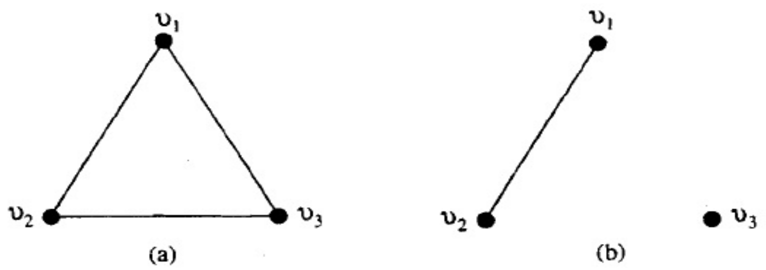


Fig.28

E9) i) The graphs having single vertices a,b,c,d,e,f, and the graphs having the following vertices and edges

i)	V = {a,b},	E = {ab}
ii)	V = {a,c},	E = {ac}
iii)	V = {c,d},	E = {cd}
iv)	V = {c,f},	E = {cf}
v)	V = {d,c,a},	E = {dc,ca}
vi)	V = {b,a,c},	E = {ba,ac}
vii)	V = {a,b,c,d},	E = {ab,ac,cd}

ii) Two components are the graph formed by the vertices a,b,c and d, and the graph formed by the vertices e and f. They are disjoint, for if they have a vertex in common the two component graphs would be connected.

E10) An example each is given below. You can think of many others.

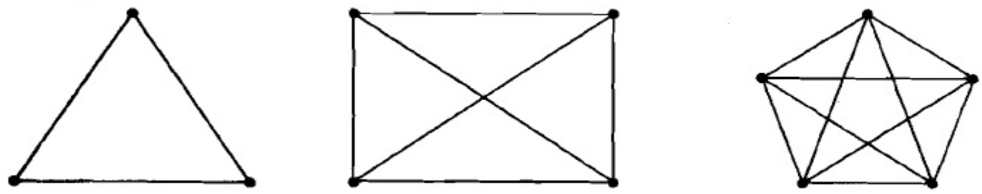


Fig.29 : $K_3 \cup K_4 \cup K_1 \cup K_5$

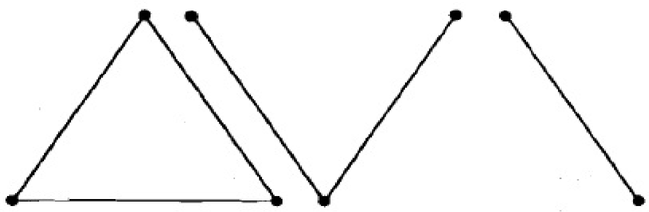


Fig.30

E11) No, since each component must contain at least one vertex.

E12) In (a) there is only one bridge given by bc. In (b) there are several bridges, e.g., u_1u_3, u_3u_6, u_5u_6.

E13) None.

E14) $\lambda(C_n) = 2, \lambda(K_n) = n-1$.

E15) $\lambda(G) = 3$.

E16) The sets given in (i), (iii), (iv) and (vi) are cutsets. The set given in (ii) is not a cutset, since its removal does not disconnect the graph; the set given in (v) is also not a cutset, since we can disconnect the graph by removing just xz and yz.

E17) The **vertex-connectivity** of a connected graph G is the smallest number of vertices whose removal disconnects G.

A **vertex-cutset** of a connected graph G is a set H of vertices with the following properties :

i) the removal of all vertices in H disconnects G;.
ii) the removal of any proper subset of H will not disconnect G.

E18) The vertex-connectivity is 1, and the vertex-cutset is {w}.

E19) Q_3 does not contain any odd cycle. Therefore, by Theorem 5, it is bipartite. The star network with n+1 vertices is $K_{1,n}$, and hence, is bipartite.

E20) Only for m=n, is $K_{m,n}$ regular.

E21) i) Yes. Let G be a bipartite graph, with a bipartition $X \cup Y$. Let H be a subgraph of G. If V(H) is disjoint from either X or Y, then $E(H) = \phi$. And then, any partition of V(H) into two subsets will serve as a bipartition. In the other situtation, $(V(H) \cap X) \cup (V(H) \cap Y)$ is a bipartition of V(H).

ii) No, e.g., the complement of G given in Fig.21 contains C_7. So, by Theorem 5, G is not bipartite.

E22) Let $G_i, 1 \le i \le n$, be bipartite graphs with the bipartitions $V(G_i) = X_i \cup Y_i$, respectively. Let $G = \bigcup_{i=1}^{n} G_i$. Then E(G) is the disjoint union $\bigcup_{i=1}^{n} E(G_i)$.
$V(G) = A \cup B$, where $A = \bigcup_{i=1}^{n} X_i$ and $B = \bigcup_{i=1}^{n} Y_i$, is a bipartition of V(G). This can be seen as follows :
Let e be an edge in E(G). Since E(G) is a disjoint union of $E(G_1),\ldots,E(G_n)$, the edge e belongs to only one of them. Without loss of generality, suppose $e \in E(G_r)$. Since G_r is bipartite with a bipartition $X_r \cup Y_r$, this means e has one end vertex in X_r and the other in Y_r, that is, e has one end vertex in A and the other in B. Thus, G is bipartite with a bipartition $A \cup B$.

E23) Fig.31 gives another matching, which is in fact a complete matching.

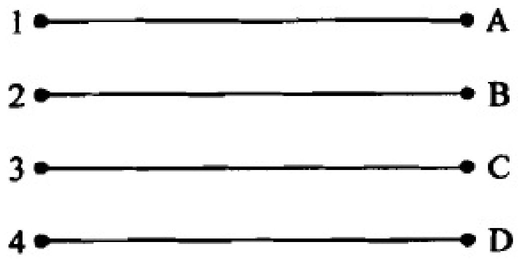


Fig.31

E24) For m=1, $n \geq 1$ or $n = 1$ and $m \geq 1$ only. For $m \geq 2, n \geq 2, K_{m,n}$ will contain a cycle.

E25) The ones in (a) and (c) are trees by Condition (iii), Theorem 7. The one in (b) is not, for the same reason. The one in (d) is not because it contains C_5.

E26)

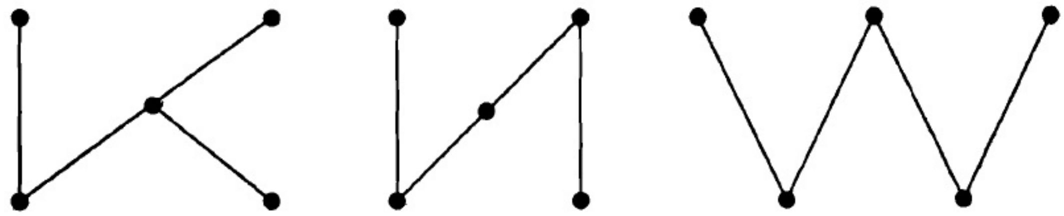


Fig.32

E27) Yes. Since a tree does not have any cycles, by Theorem 5 it is a bipartite graph.