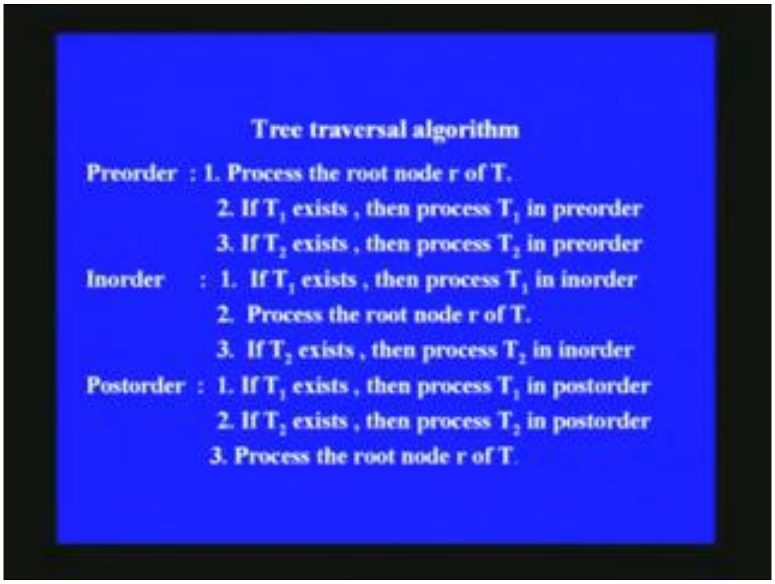
Discrete Mathematical Structures
Dr. Kamala Krithivasan
Department of Computer Science and Engineering
Indian Institute of Technology, Madras
Lecture # 17
Trees and Graphs

So we were considering traversal algorithms for binary trees. There are three traversal algorithms. In the preorder one you process the root vertex first then the left subtree then the right subtree. And in the inorder one the left subtree is traversed first in the inorder manner then the root then the right subtree is traversed in the inorder manner.

In the postorder traversal first the left subtree is traversed in the postorder manner then the right subtree is traverse in the postorder manner then the root. Now, given a preorder sequence of the vertices visited and the inorder sequence you can construct the tree in a unique manner. Similarly, given the inorder sequence and the postorder sequence you can construct the tree in a unique manner. But given preorder and postorder you cannot construct the tree in a unique manner.

(Refer Slide Time: 1.14)



So let us see how to construct the tree given a preorder and inorder sequence. Let us take this example, you have vertices a, b, c, d, e, f, g, h, i, j, k, i, j, k, l. Now, let us see how to construct the tree from this preorder and inorder sequences. The preorder sequence is given like that and the inorder sequence is given like that. Now what do you understand by this?
In the preorder sequence the root is visited first so a is the root of the tree so this is the root. So these vertices will constitute the left subtree and these nodes will constitute the right subtree. And here you can see that this is the different order but that constitutes the

left subtree. Now, if you look at this way this left subtree is again traversed in the preorder manner so b will be the root, so b is the root here. So if you look here b is the root, so d is the left subtree of that and h, k, e occur on the right subtree. So d will be here and h, k, e should occur on the right subtree.

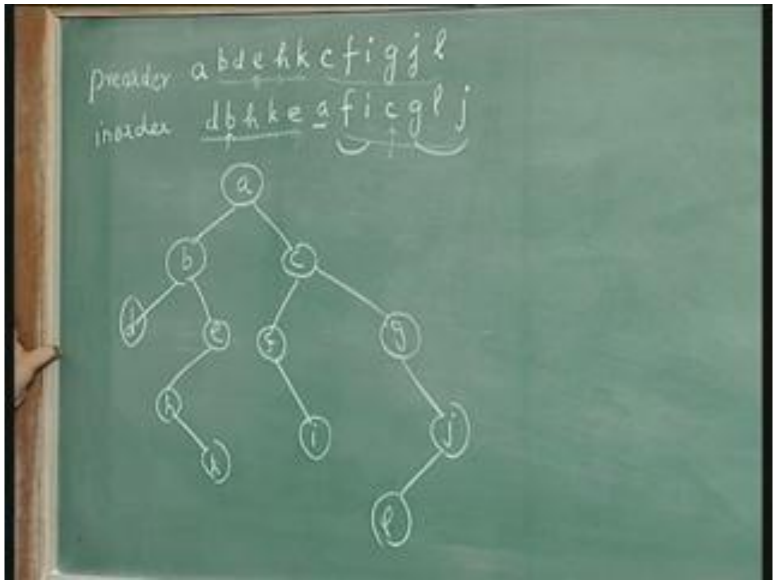
How do you arrange these vertices e, h, k? They are in the different order here in the preorder. Now again the root will come first in the preorder so e will be the root and h, k should occur on the left subtree or the right subtree. So here e is the root and h, k occur on the left subtree so both h and k should come in this side.

Now, again h should be the root in the preorder so h will come like this and then because k occurs on the right side k will occur like this. This is the way the left subtree is belt. Now look at the right subtree these are the 6 vertices occurring in the right subtree. Again in the preorder manner you can see that the first node is c so c is the root, here c is the root so f, i should occur on the left subtree of c and g, l, j should occur on the right subtree. Again, if you look at this one f comes first so f should be the root of the left subtree and here you see that i occurs on the right next to f so it should occur on the right subtree so it will be like this.

Now g, l, j should occur on the right subtree of this and because g occurs first it will be the root so g will be the root of this tree and l, j will occur on the left or the right subtree. But here again you find that both l and j occur on the right side next to g. So they will occur on the right subtree and j occurs first here so j will be the root and l occurs to the left of j so l will occur like this. So this tree can be built in a unique manner from the preorder and the inorder sequence of the vertices visited.

Now we have also seen that these trees binary trees can be use to represent arithmetic expressions and for evaluating arithmetic expressions.

(Refer Slide Time: 5.32)



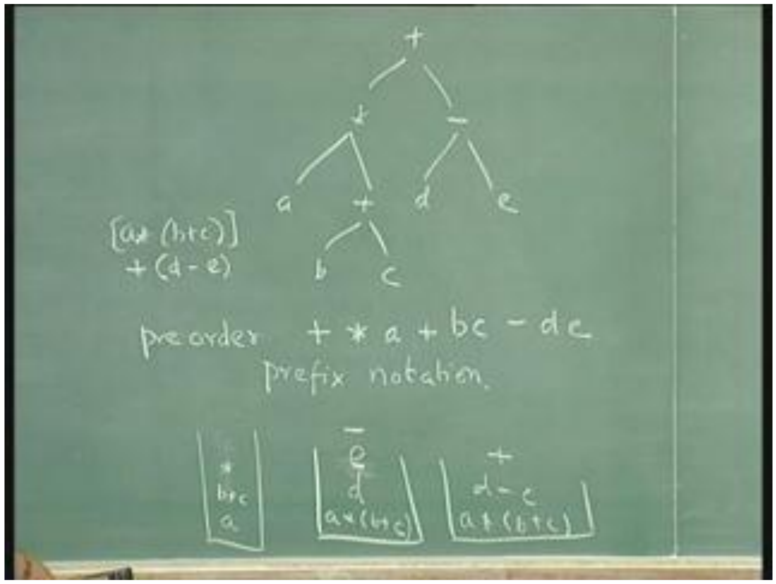
And if you traverse such a tree in a preorder or in a postorder manner what you get. So let us take this example, this is a binary tree where the internal nodes are operators and the leaf nodes are some operands. If you traverse this tree in a postorder manner what is the sequence of vertices visited and how do you write that sequence?

You see that in a postorder left subtree, right subtree, root that is the order in which you go so you will see that the order in which you traverse will be given by a b c plus star d e minus plus a then the right subtree is b c plus then star and then this right subtree is d e minus then the whole thing is plus. This is called a post fix notation for arithmetic expressions.

Now the expression will be evaluated like this; you will have a stack in which operands will be transferred one by one. So a will be transferred, b then c then when plus an operator is transformed it operates, it is a binary operator it operates on these two and the result of this b plus c is kept here. Then the next operator is transferred and it operates on these two so you get a in the stack you get a star b plus c. Then you transfer d to the stack transfer e then the operator comes minus it operates on these two and you get a star b plus c then you get d minus e this operator operated on this. When this plus is transferred it will operate on this and you will get the expression a plus a star b plus c this is one plus d minus e. This is what you get. This is the arithmetic expression and this is represented by this. Similarly, you can also have the preorder way of representing that. So in the preorder way if you represent visit the nodes it will be plus star a plus b c minus d e this is the order in which the nodes will be visited and this is called the prefix notation. These are the ways in which an arithmetic expression is represented internally in the computer by the compiler as an intermediate code then the machine code will be generated.

We have seen some facts about search trees and trees as data structures. In undirected graph also you can talk about trees.

(Refer Slide Time: 10.08)



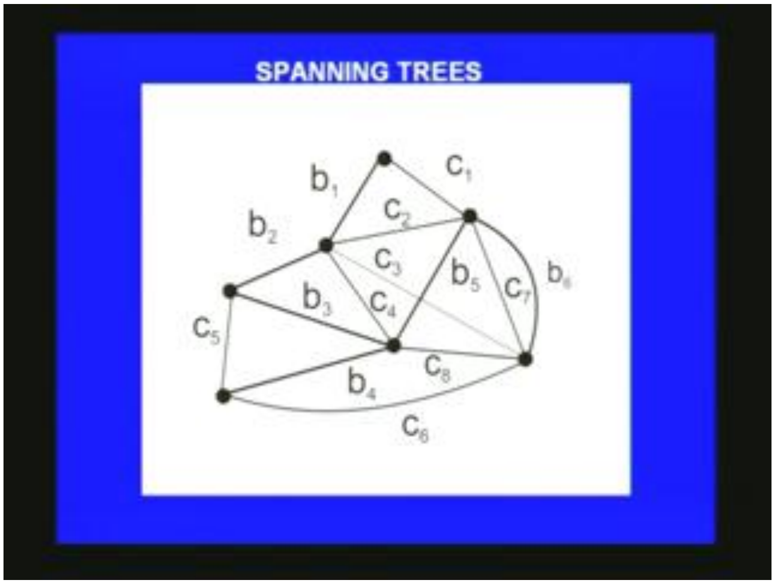
So now let us take some undirected graph and what do you mean by a tree. Take undirected graphs, a tree has no cycle, something like this is called a tree in an undirected graph. Look at this, there is no cycle in this graph such a thing is called a tree. A tree with n nodes has n minus 1 edges, this also you can very easily see. There are several properties about trees which can be considered from any book on graph theory. We will not go into the details of that. But for any graph also not necessarily a tree but any graph you can define what is known as a spanning tree. Let us see what a spanning tree is. A tree T is said to be a spanning tree of a connected graph if T is a sub graph of G and T contains all vertices of G.

(Refer Slide Time: 11.19)

These spanning trees are very very important in several practical applications like computer networks and so on. You would basically require as spanning tree and sometimes it is also very necessary to calculate the minimal spanning tree. In a moment let us see what is meant by that. Look at this graph, it is a graph and a subgraph of that some of the edges are labeled by b and some of the edges are labeled by cs. Look at this, how many vertices are there? There are seven vertices; 1 2 3 4 5 6 7. Look at the edges b_1 b_2 b_3 b_4 b_5 b_6. These six edges b_1 b_2 b_3 b_4 b_5 b_6 form a subgraph of this, a connected subgraph of this and it has a structure of a tree, there is no cycle in that.

Obviously you can see that the graph has seven vertices and there are six b edges they are called branch edges. The subgraph consisting of the b edges b_1 b_2 b_3 b_4 b_5 b_6 is called a spanning tree of this graph because it is a tree, it is a subgraph of g and also it contains all the vertices of g. The other vertices are c_1, c_2 etc, there are eight more edges.

(Refer Slide Time: 11.54)

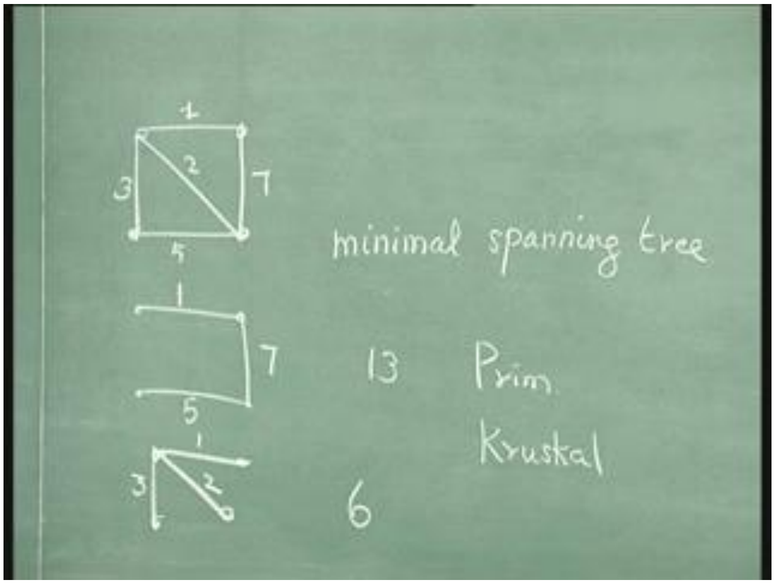


Now you can see that when you have a spanning tree consisting of the branches just add one c edge then you get b_1 b_3 b_5 c_1 that forms a cycle. Even if you add one more edge from the graph to the spanning tree you will get a cycle or a circuit and that is called a fundamental cycle, fundamental circuit or cycle. This is obtained by just adding one edge of the graph which is not a branch to the spanning tree. The spanning tree of a graph is not unique you can have several spanning trees for a graph.

For example, take this graph it has got four vertices, this can form a spanning tree, this can form a spanning tree and so on. There can be several spanning trees. Now sometimes it is of use and very interesting to find out what is a minimal spanning tree.

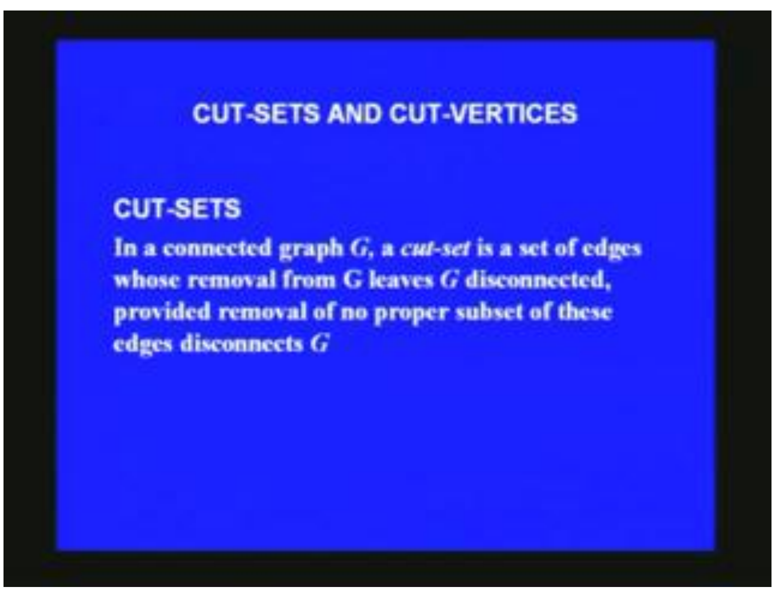
For example, if the edges have weight say 1, 2, 3, 5, 7 and something like that there are five edges in the graph and the spanning tree will have only three edges. If you choose this spanning tree the weight of the tree will be 1, 7, 5 which will add up to 13. If you choose this tree it is 1, 2 and 3 the weight will add up to 6. So this is the minimal spanning tree in this example. You are interested in finding the minimal spanning tree because if you want to have telephone connection and all that sometimes tariff is calculated based on the minimal spanning tree. There are several important and interesting algorithms for calculating the minimal spanning tree. One is Prim 's algorithm and another is Kruskal's algorithm. They are well known algorithms for finding so much about spanning trees.

(Refer Slide Time: 16.24)



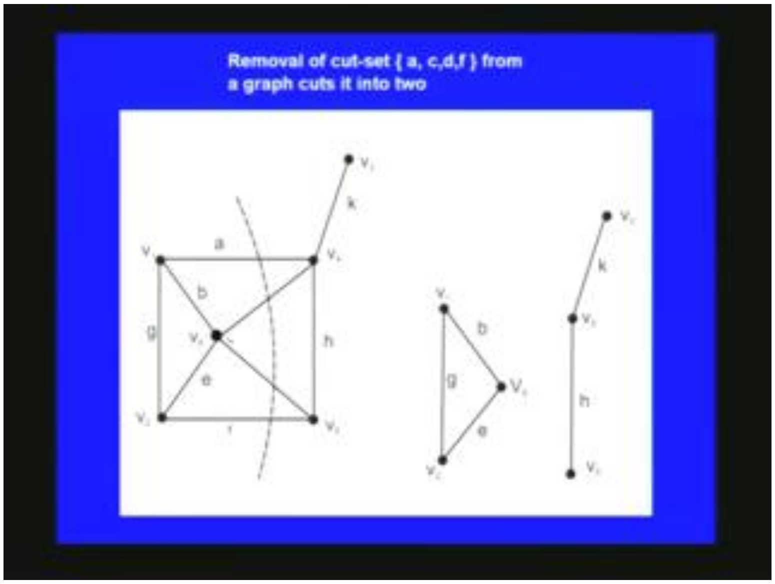
Next we shall see what are meant by cut-sets and cut vertices. In a connected graph G a cut-set is a set of edges whose removal from G leaves G disconnected provided removal of no proper subset of these edges disconnects G. This is the definition of a cut-set. I will read it again, in a connected graph G, a cut-set is a set of edges whose removal from G leaves G disconnected, provided removal of no proper subset of these edges disconnects G.

(Refer Slide Time: 16.36)



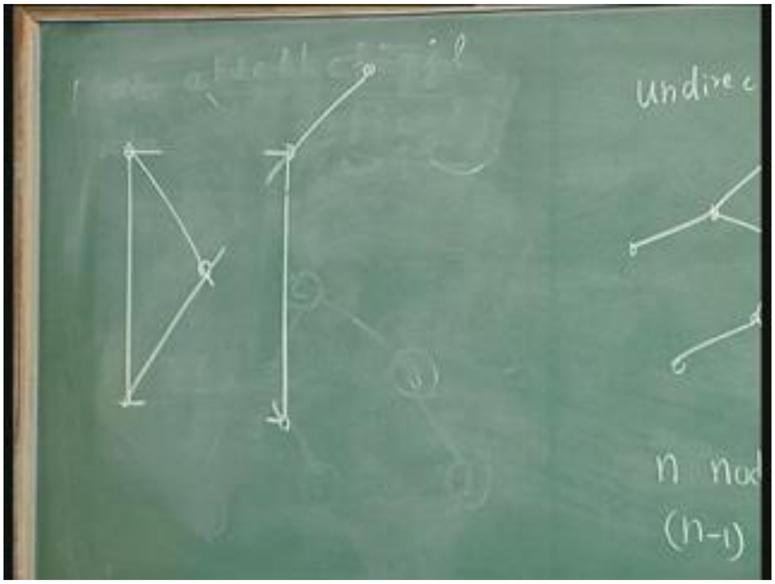
Now look at this graph, in this graph if you remove this edge, this edge, this edge and this edge these four edges if you remove you get two components which are not connected.

(Refer Slide Time: 17.21)



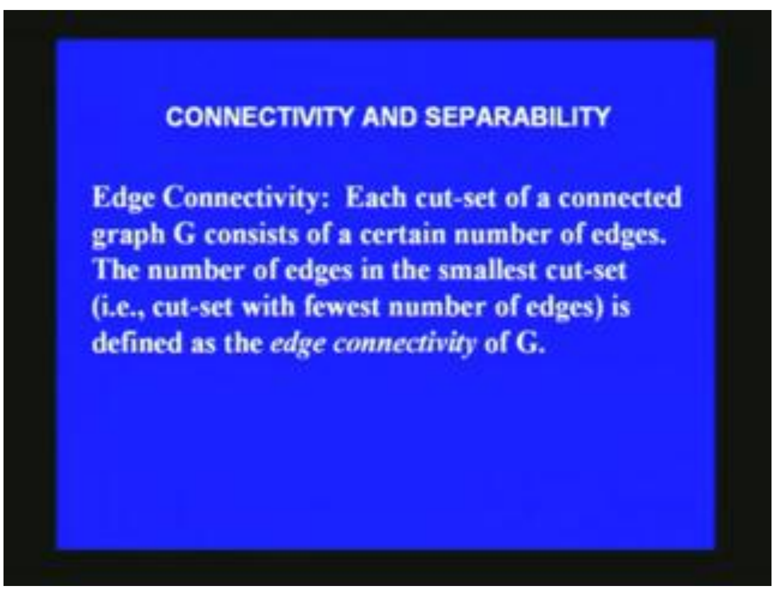
So let us consider this, I will draw it again on the board. Now these four edges if you remove mark the nodes, if you remove these four edges you get two disconnected components. But even if you leave one of them it will be connected. If you have even one edge present here it will be connected even if this is present it will be connected, if this is present it will be connected and if this is present it will be connected. So you have to remove all the four edges to get to disconnected components. If you remove any sub set of this any three or two or one of this still the graph will remain connected. So such a set is called a cut-set.

(Refer Slide Time: 18.32)



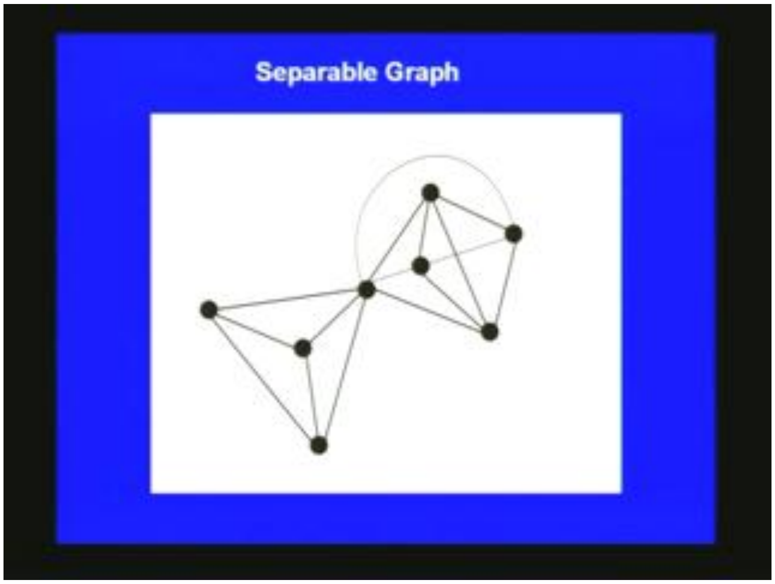
Now look at this graph again. If you remove these three edges then also the graph becomes disconnected like this but now the cut-set consists of these three edges. So you may have several cut-sets for a graph some of them have two edges, three edges like this. The cut-set having the minimal number of edges is the minimal cut-set and that is called the edge connectivity of the graph. Each cut-set of a connected graph G consists of a certain number of edges. The number of edges in the smallest cut-set that is the cut-set with fewest numbers of edges is defined as the edge connectivity of G. So the edge connectivity is defined as the size of the smallest cut-set.

(Refer Slide Time: 19.15)



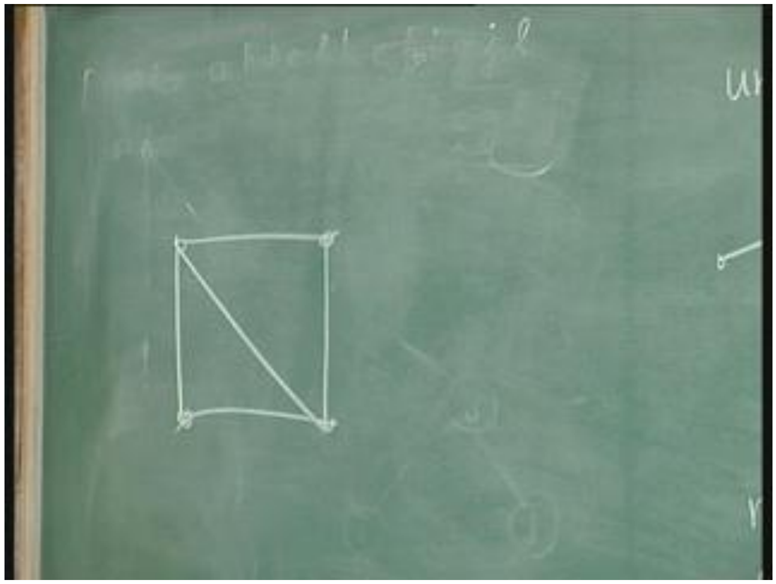
Look at this graph, in this graph if you remove these three edges you may have, this is a cut-set, remove these three edges you may get disconnected components. If you remove these four edges you may get two components and so on. But if you remove this vertex along with the edges connected to that you will get two disconnected components. So by removing one vertex you are able to get two disconnected components and that is called vertex connectivity. Sometimes by removing one vertex you may not be able to get disconnected components you may have to have several vertices removed to get disconnected components.

(Refer Slide Time: 20.28)



For example, if you take some thing like this, say if you remove any one vertex along with edges associated with that you will get only a connected component. But if you remove two vertices if you remove this vertex and this vertex you will get these two vertices they are disconnected components. So here the vertex connectivity is defined to be two. In this graph if you remove this vertex then it becomes disconnected, one connected component here, one connected component here and the disconnectivity is obtained by removing this vertex. So this graph has vertex connectivity one.

(Refer Slide Time: 22.03)



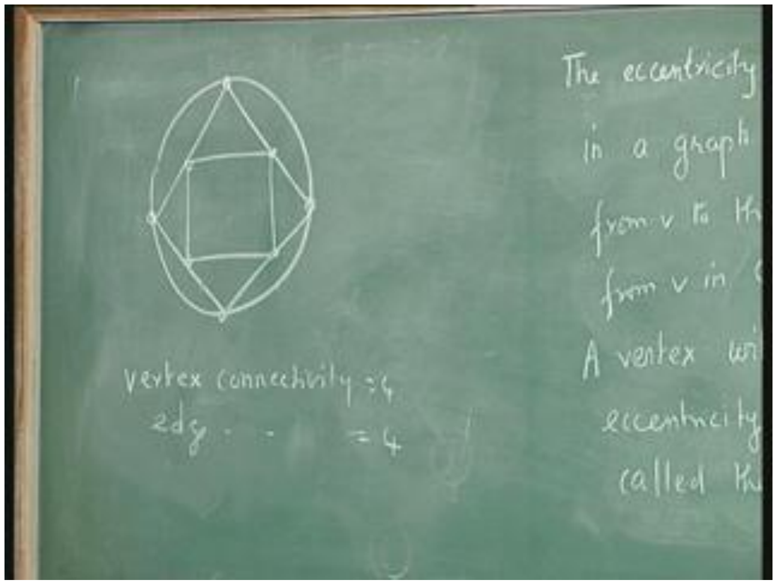
So let us define vertex connectivity in a proper manner. If you look at the graph we find that although removal of no single edge disconnects the graph. Here if you remove any one single edge the graph will not be disconnected. If you remove these three edges it may become disconnected. Here if you remove these four edges it may become disconnected and so on. But removing just one edge will not leave the graph disconnected. But if you remove this vertex it will become disconnected.

We find that although removal of no single edge disconnects the graph the removal of single vertex v does, it disconnects the graph. We define another analogous term called vertex connectivity. The vertex connectivity of a connected graph G is defined as the minimum number of vertices whose removal from G leaves the remaining graph disconnected. So vertex connectivity is defined as the minimum number of vertices whose removal from G leaves the graph disconnected. If the vertex connectivity is equal to 1 it is called a separable graph, the graph is called a separable graph. The graph which we have seen earlier is a separable graph.

We have seen that this separable graph has 8 nodes and how many edges are there? It is 3, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, so 16 edges are there. Now this also has similar structure eight nodes but you see that the vertex connectivity here is 4. If you remove any three of them still the graph will be connected.

Similarly, edge connectivity also here is 4. If you want to have connection between eight nodes which represents some stations or something like that you would rather like to have a structure like this than this because if some fault occurs here in this network if this node fails there will not be any connection from this portion to this portion. So you would like to ensure as much of connectivity as possible. If you take such a structure then even if one node fails or even if two nodes fail there will be some connection between the other nodes so you will not have any problems. So this sort of an idea, definition of vertex connectivity plays an important part in practical applications.

(Refer Slide Time: 26.20)

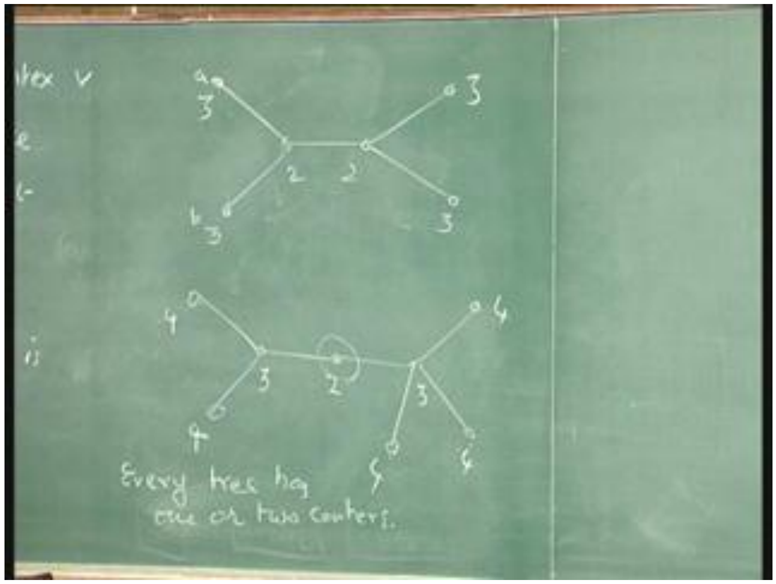


Now we have also seen about trees in undirective graph. Let us see one or two more definitions about trees. The eccentricity e (v) of a vertex v in a graph g is the distance from v to the vertex farthest from v in g. So, if u take a graph like this take a vertex the eccentricity of the vertex is the distance or length of the path simple path from this node to the node which is farthest away from that. And the vertex with minimum eccentricity in a graph is called the center of the graph g. Now, in the particular case of trees let us see what the eccentricities are and what are the sentences.

Look at this graph, this is a tree you know that this is a tree it has got 6 vertices. The distance between this node a and this node b is 2, length is 2. The distance between this node and this node is 3, the distance between this node and this node is 3, so the distance of the farthest node is 3 so the eccentricity of this vertex is 3.

Similarly, you can see that the eccentricity of this vertex is 3 while the eccentricity of this vertex is 2 because these two nodes are at distance 1 from this and this node is at distance 1 and these two nodes are at distance 2. So the eccentricity of this vertex is 2, the eccentricity of this vertex is also 2 in a similar manner and the eccentricity of these three vertices are 3 because the distance of the farthest point is 3. And similarly if you look at this graph the distance for this node is this, this or this and the length of the path or the distance is 4 so the eccentricity is 4 here, the eccentricity is 4 here in a similar manner, the eccentricities are 4 in these cases. The eccentricity will be 3 here because there is a node which is at distance 3 from here the eccentricity of this node is 2. The center of a graph is defined as the vertex with minimum eccentricity.

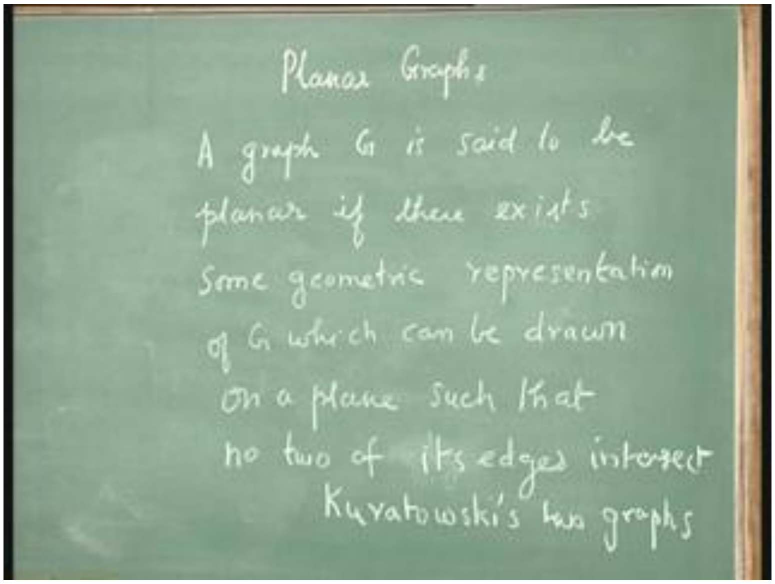
(Refer Slide Time: 31:03)



So if you look at this, this vertex has the minimum eccentricity therefore it is the center. Here these two vertices have minimum eccentricity so they are centers. And you can very easily prove that every tree has one or two centers. How can you do that?
You can just see that in this case if you remove the pendent vertices, pendent vertices will definitely have more eccentricity, you end of with an edge and both these vertices have eccentricity 2 and they form the centers.

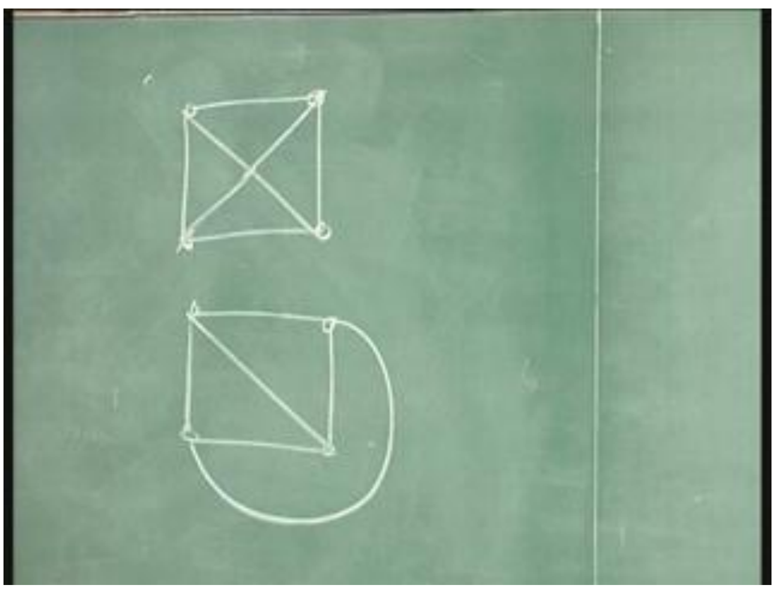
Now do the same thing here, remove the pendent vertices then you end up with a graph like this, these two are pendent vertices now remove them also then you end up with the single vertex and you end up with the single vertex that becomes the center. So either you have two centers or one center in a graph in a tree and this can be easily proved. So these are some facts about trees, cut-sets and cut vertices and so on. Now let us see what is mean by a planar graph. A planar graph is defined like this: A graph g is said to be planar if there exists some geometric representation of g which can be drawn on a plane such that no two of its edges intersect.

(Refer Slide Time: 31:15)



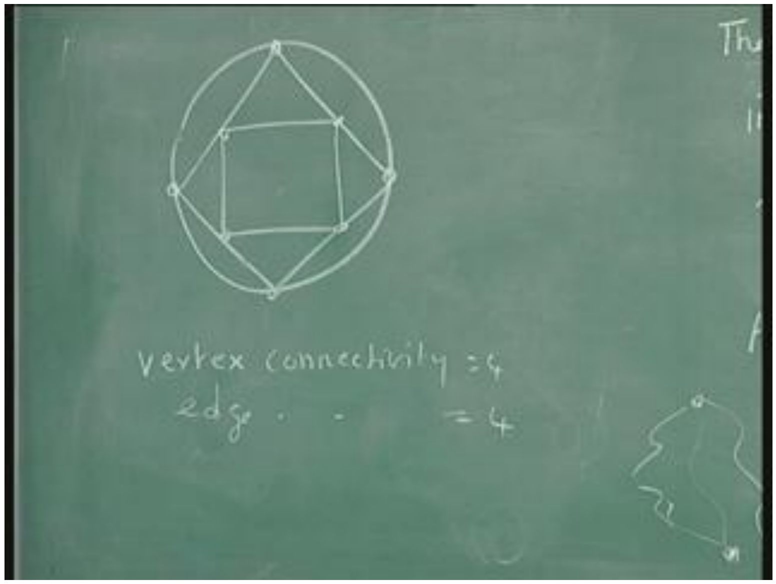
Take for example this graph. Consider this graph with four vertices, the edges intersect like this. But the same graph you can draw like this.

(Refer Slide Time: 32:10)



You can also draw in a different way. Instead of drawing this edge this way you can draw it this way in which case no two edges intersect. And if there is someway of representing the edges such that no two edges intersect, if there is someway of drawing the graph on a plane such that no two edges intersect it is called as a planar graph, this is the planar graph. Is every graph planar? Obviously no. There are examples of graphs which are not planar.

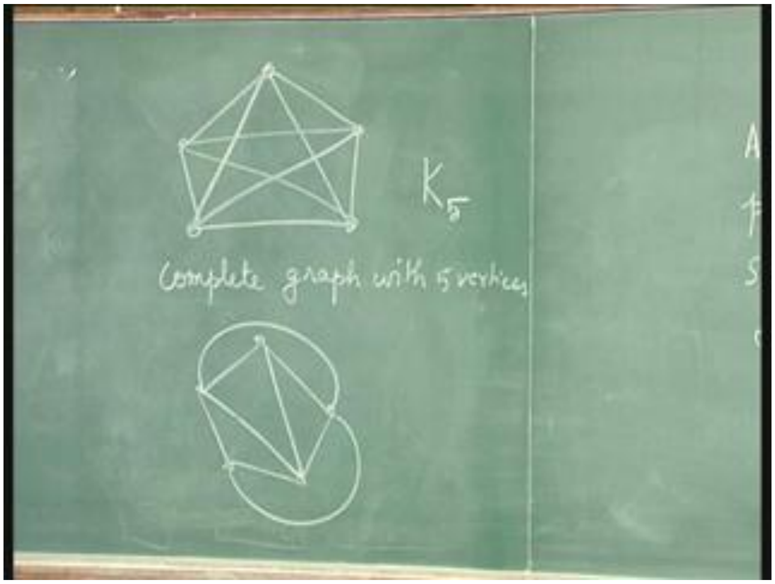
(Refer Slide Time: 32:22)



There are two graphs known as Kuratowski's two graphs which are known to be nonplanar, Kuratowski's two graphs. What are those graphs?
Look at this graph, this is a complete graph with 5 vertices, this is denoted as k_5. This is not a planar graph. Try to draw it in a planar manner, you have 5 vertices so you can connect like this, there are 10 edges. You can have one like this, one like this, no problem, 7 edges you can draw like this. The edge between this you can draw in this way, the edge between this you can draw like this, what is the other edge? The other edge is the edge between this and this.

I you try to draw it this way you have to cross this, if you try to draw it this way you have to cross this, if you try to cross this either this way also you have to cross this edge or you have to cross this edge either way you have to cross one edge. So you can draw 9 edges without crossing but the 10th edge you cannot draw without crossing. This is an example of a graph which is not planar.

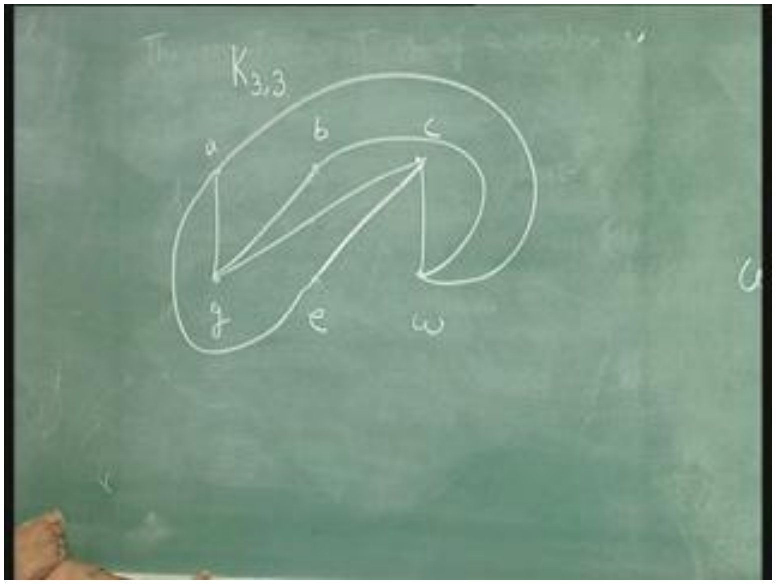
(Refer Slide Time: 34:36)



Then the other graph which we consider is Kuratowski's second graph (check 34) which is known as $k_{3,3}$ that is a utilities problem, we had three houses and three utilities gas, electricity and water and they have to be connected and so on. So there are three houses like this; a b c and they have to be connected to gas, electricity and water and how do you do the connection without the lines crossing over.

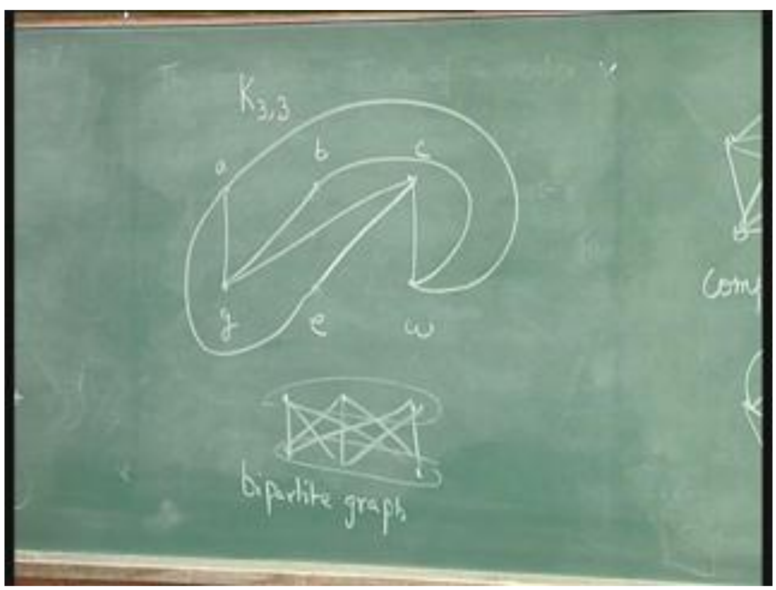
Here again how many edges are there? There are 6 vertices and 9 edges from each one you have 3 so there are 9 edges and 6 vertices but you cannot draw it without cross over. Suppose you draw like this you can draw this, you can draw this, five of them you can draw like this then a to b you have drawn, a to c you have drawn, e to c you have drawn e to a you can draw like this 6, 7 and 8 you can draw like this. But what is the one which is remaining? e to c you have drawn, e to a you have drawn. If you want to draw from e to b if you draw like this it will cross this edge, if you draw like this it will cross this edge. Anyway you cannot connect e to b without crossing one edge. So this particular graph you cannot draw on the plane without cross over there will be at least one cross over there and this is known as $k_{3,3}$ and these two are examples of, generally it is drawn in this manner. It is known as a Bipartite graph.

(Refer Slide Time: 36:37)



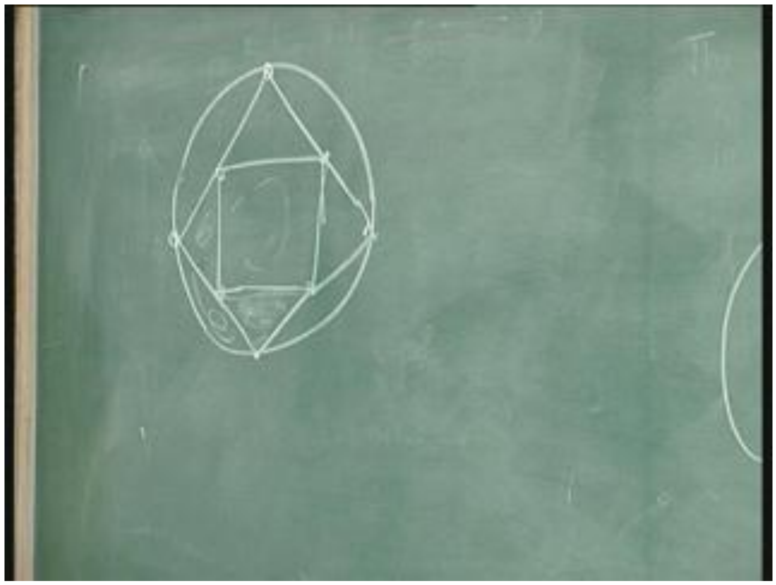
The set of vertices are divided into two portions and all edges are such that they have one end in one portion and another end in one portion. Such a graph known as a Bipartite graph. This is an example of Bipartite graph. It is a complete Bipartite graph on 3 plus 3 vertices that is why it is denoted as $k_{3,3}$.

(Refer Slide Time: 37:42)



Now we have seen this these two graphs are not planar. How can you find out whether the graph is planar or not? And also in planar graphs you talk about regions. Take this example, this is the planar graph you can see that it is drawn on the plane without any intersection, these are called regions, this is one region, this is one region, this is one region, this is one region and so on. So if you draw a graph on the plane without cross over there is some connection between the number of edges, the number of regions and the number of vertices, what is this.

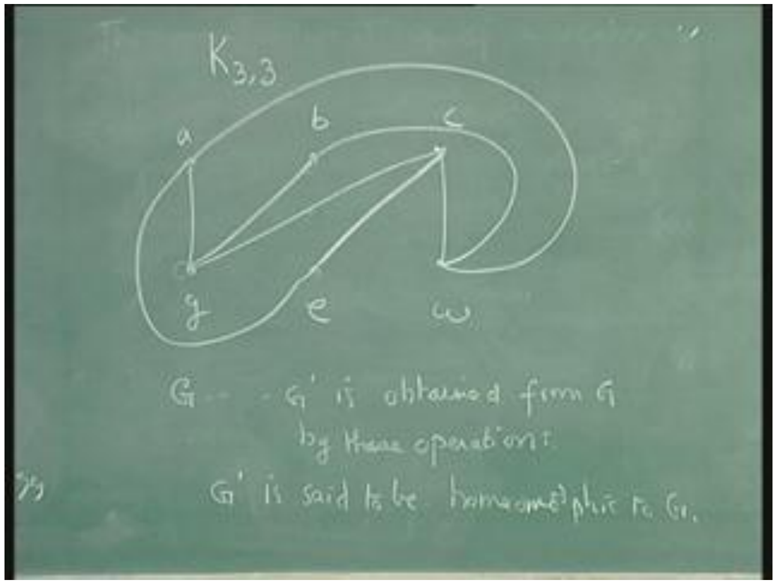
(Refer Slide Time: 38:20)



Let us see what is the relation? Now, what is the condition that a graph is planar?
Now you see if you have a graph and if you have a self loop wherever you want have a self loop you can draw it in a small vaner and so without ant problem that will not constitute any cross over and so on. Similarly, if you have parallel edges, between these two if you have parallel edges you can draw them as close as you want without any cross over and other cross over will not be affected by this. So without loss of generality you can remove loops and parallel edges.

Also, if you have a graph like this, some graph, if you have a vertex with degree 2 instead of having this if you draw a single edge that is not going to affect the cross over in anyway. So you can remove such vertices and merge the two edges. So what you can do is remove vertices of degree 2. That is, if you have vertex of degree 2 like this and merge the two edges. If you have a vertex v of degree 3 you can remove this and then merge the thing and this will be replaced by an edge like this. This will not affect the planarity or non planarity of a graph. Now, if you do that what happens? If G is a graph and G dash is obtained from G by these operations then G dash is said to be homeomorphic to G.

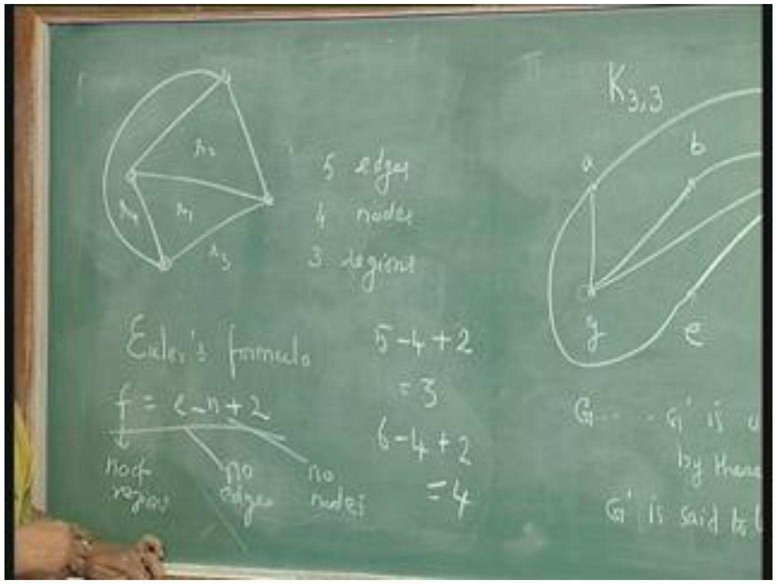
(Refer Slide Time: 42.02)



Now, if a graph does not contain k_5 or $k_{3,3}$ as a subgraph or any graph homeomorphic to it then it is planar. It should not contain k_5 or $k_{3,3}$ as subgraph. If G does not contain k_5 or $k_{3,3}$ as subgraph or any graph homeomorphic to it then it will be planar. Both these are necessary and sufficient conditions you will not go into the details of it let us take the result as it is. Like this you can find out whether a graph is planar or not.

Now as I told if you have a planar graph like this, this is a graph with 5 edges and 4 nodes how many regions it has got? It has got region 1, region 2, region 3 so 3 regions. And what is the connection between them?
The connection is brought out what is known as Euler's formula, e minus f is equal to e minus n plus 2. So f denotes the number of regions, e is the number of edges and n is the number of nodes. Look at this, how many regions are you having? You are having 3 regions. What is e minus n? e is 5 minus n plus 2 is 3. So you are having 3 regions here, it satisfies. Euler's formula like this brings out the relationship. Suppose I have one more edge between this and like this, how many regions I have now? I have 4. Obviously there are 6 edges now, so 6 minus 4 plus 2 is 4, 4 regions I have, this is known as Euler's formula.

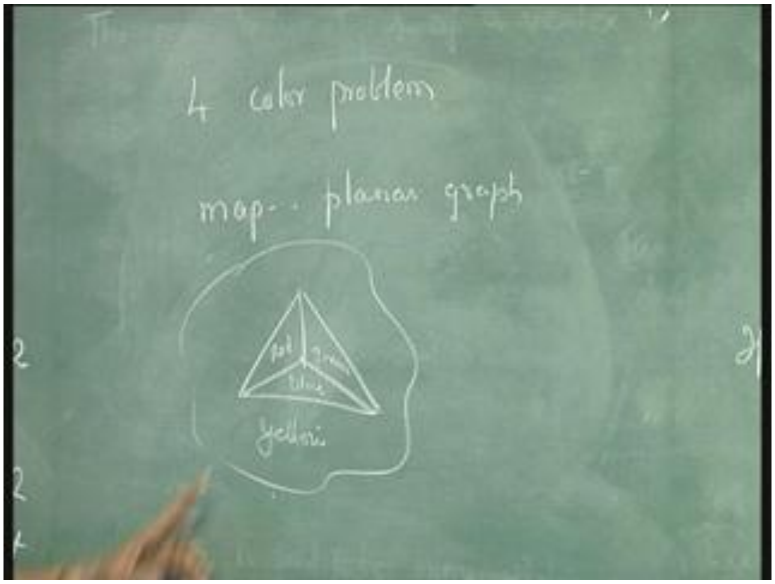
(Refer Slide Time: 45.20)



And you might have heard about the famous four color problem, what is that four color problem?
A planar graph is sometimes also called a map planar graph. Look at this, and each region I can color with different colors. For example, you can use red here, green here, blue here, yellow here, you can color this with blue, you can color this with red and you can color this with green and so on. How can I color a map which is a planar graph with minimum number of colors?
People have seen that you require at least four colors.

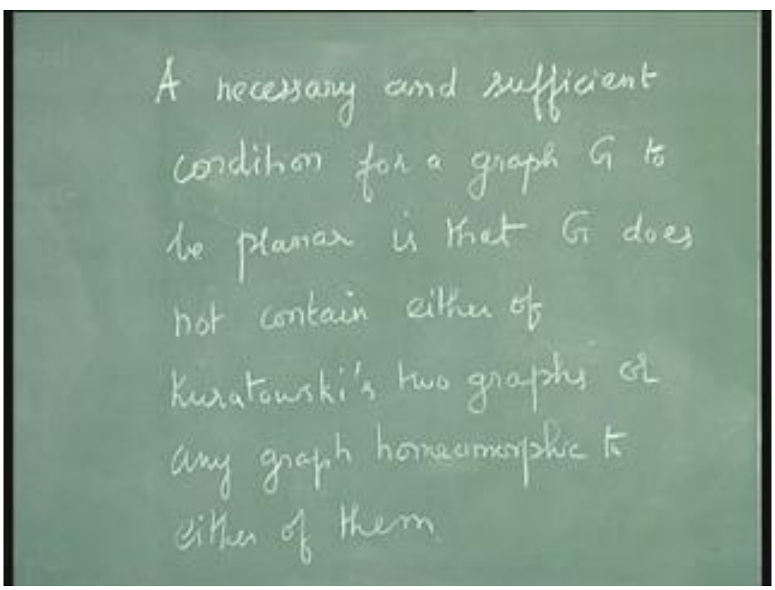
Take this simple example, this is a planar graph, to color this you require at least four colors you cannot do with less than three colors. Suppose I use any other color instead of yellow if I use red, green or blue again two of the adjacent regions will be colored with the same color. So I want to color this so that no two adjacent regions get the same color. But I want to use the minimum number of colors. Long back it has been shown that four colors are necessary you cannot do with three colors and five colors are sufficient that was also shown quiet early. But whenever people tried they were able to color the graph at least up to forty without any problem they were able to color with four colors. So it was conjectured that four colors were enough but it was not proved, it remained as a conjecture for a long time and it remained as a four color conjecture for a long time. But a few years back it has been proved, this conjecture had been proved to be correct, that is four colors are sufficient to color a planar graph such that no two adjacent regions get the same color. This is the famous four color problem.

(Refer Slide Time: 46.18)



So we have seen that a necessary and sufficient condition for a graph G to be planar is that G does not contain either of Kuratowski's two graphs k_5 or $k_{3,3}$ or any graph homeomorphic to it to either of them. Now, we have also seen that a planar graph can be colored with four colors so that adjacent regions do not get the same color.

(Refer Slide Time: 49.02)



We have also seen Euler's formula connecting the number of edges, number of nodes and number of regions on the number of faces which is denoted by f by this formula. Let us derive this formula, how do you get this formula?

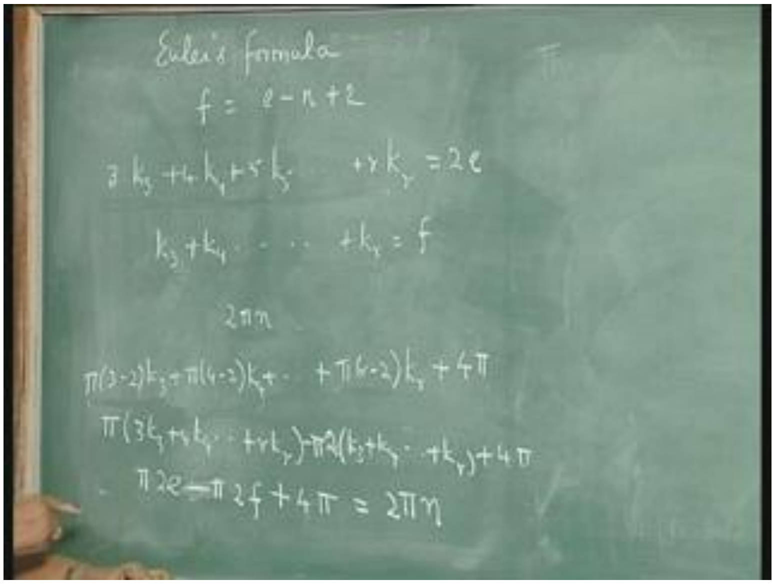
Take any planar graph, suppose the number of triangles is given by 3, the number of quadrilaterals is given by k_4, the number of pentagon is given by k_5 and so on you may have a graph like this and there may be some triangles the number of such triangles is given by 3. Now count the edges, the external will be a polygon again internally you may get some quadrilaterals, pentagons, etc. If k, r denotes the number of polygons with r sides count the edges the number of edges in the triangles will be 3 k_3, the number of edges in the quadrilateral be 4 k_4 and so on, if you count that will sum up to this. But each edge will be adjacent to two regions.

So when you count the edges covering a polygon each edge will be counted twice so you get 2e so this sums up to 2e. And the number of faces or regions is given by k_3 and k_4 plus etc, k, r is equal to f, one of them will be in external region. Now, if you take each node the total angle around that is 2pi that we know there are n nodes so if you count all the angles it sums up to 2pi m. But how do you get that?
There are some polygons, if a polygon is p sided the sum of internal angles adds up to pi cross p minus 2 sum of the internal angles. Sum of the external angles adds up to pi cross p plus 2. Now out of these triangles, quadrilaterals etc one will be an external polygon one you have to count as external and the rest of them as internal.

So if you count the number of angles depending upon these polygons, the number of internal angles of the triangles will sum into pi cross 3 minus 2 cross k_3 and the sum of the internal angles of the quadrilaterals will sum up to pi cross 4 minus 2 cross k_4 and so on and the sum of the internal angles of r sided polygon is given by this.

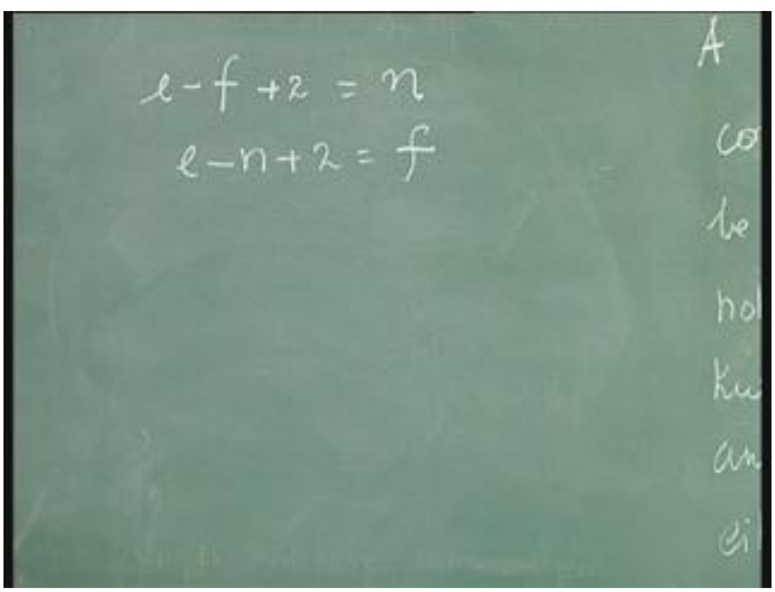
But you must remember that one of this polygon covers the outside region so for one polygon out of this you must use pi cross p plus 2 rather than pi cross p minus 2. So we add we do not know which is the outside polygon so whatever it is for one of them you have to add 4pi. So if you count like this the angles around each node will sum up to this quantity. And you know that, let us simplify this expression, this is equal to 2pi m.

(Refer Slide Time: 53.46)



This you can write as pi cross 3 k_3 plus 4 k_4 plus etc plus r k, r twice k_3 plus k_4 two times minus pi cross two times k_3 plus k_4 plus k, r plus 4pi and this quantity by this we know that it is 2e minus pi cross 2 and this quantity by this factor we know is f plus 4pi and all this is the sum of the angles around each node. There are n nodes, around each node the sum of the angle is 2pi so this will add up to 2pi n. Now, if you divide by 2pi this will come to e minus f plus 2 is equal to n. You will get e minus f plus 2 is equal to n or e minus n plus 2 is equal to f which is known as Euler's formula. And by this we get a result about regular polyhedron.

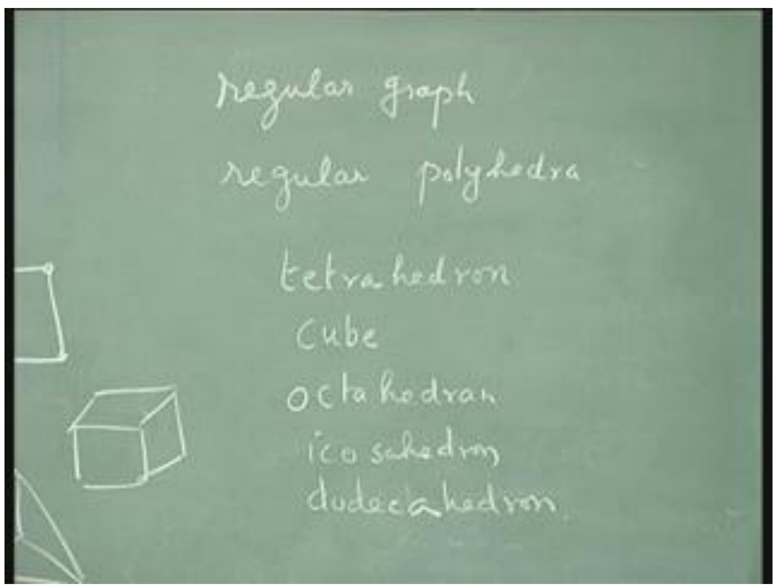
(Refer Slide Time: 54.07)



A regular graph is one in which each node has the same degree. For example, if you take this, this is a regular graph because each node has degree 2. A regular polyhedron is one which has got all the faces of the polyhedron or regular polygons. For example, if you take a cube each face is a square it is the regular polyhedra. If you take a tetrahedron consisting of equilateral triangle then that is the regular polyhedron. And by this result about Euler's formula you can show that there can be only five regular polyhedra.

The Greeks in ancient Times realized that there can be only five regular polyhedra and that result is proved by Euler by making use of the Euler's formula. The five regular polyhedra are these; tetrahedron, cube, tetrahedron consists of four triangles, Cube consists of six squares, Octahedron consists of eight triangles, Icosahedrons consist of twenty triangles each face is a triangle, dodecahedron each face is a pentagon. And this has been derived from Euler's formula.

(Refer Slide Time: 55.45)



So these are some results about a graph theory. The graph theory itself is a very wide field. What we have seen is only a glimpse of what is graph theory. This has lot of practical applications in electrical switch circuits and computer networks, flow analysis and things like that. So, next we can consider some more properties about relations.