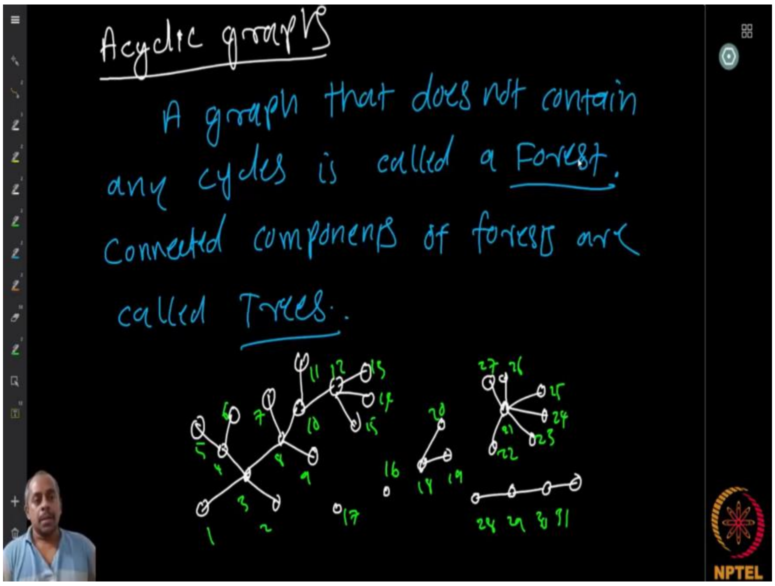
Combinatorics
Professor. Dr. Narayanan N
Department of Mathematics
Indian Institute of Technology, Madras
Acyclic Graphs

(Refer Slide Time: 0:12)

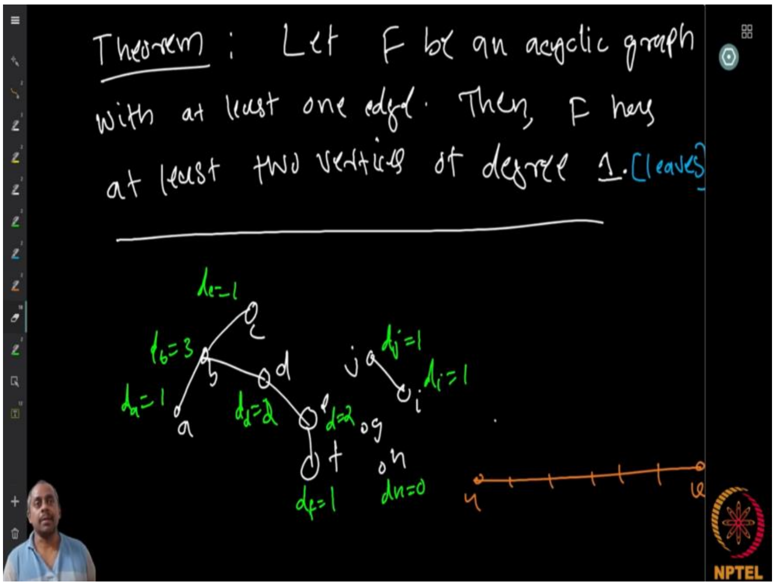


So, we continue with our lectures on graph theory. Today, we are going to discuss what is called acyclic graphs in the first part. So, what is an acyclic graph? As the name suggests, a graph that does not contain any cycles. So, they are called acyclic graphs or a more commonly used word is Forest. So, acyclic graphs are called Forest. Now, here is an example of a forest. If you look at this, you can see this graph represented below does not have any cycles in it. So, all the components of a forest are called trees.

So, components are the maximal connected subgraphs, if you remember. So, you can see that there is this component, then there are these two single vertices called 17 and 16 are components. Then there is this path on three vertices which is another component and then you have a vertex with several neighbours, this is another component and then there is a path with four vertices. So, these are all examples of trees and then together form an acyclic graph which we call it as Forest. So, here we have a forest.

Now, there is a special name for this type of graph where we have one central vertex and every other vertex is the neighbour. Such a tree is called a star and stars are also some important classes of trees that we will see in some time. Now, so, this is an example of acyclic graph.

(Refer Slide Time: 2:22)



Now, here is one theorem. Suppose, you are given an acyclic graph let us say F and assume that this forest has at least 1 edge and you can have isolated vertices but apart from that if you have this graph with at least one edge, then this forest has at least two vertices of edge exactly equal to 1. So, the vertices of degree 1 are called leaves. So, in a tree if you have a vertex of degree 1, they are called leaves of the tree.

So, how do you prove this theorem? So, you want to show that given an acyclic graph with at least 1 edge, it contains at least two vertices of degree exactly equal to 1. Now, here you should think about this for a few minutes, it is very easy to come up with a proof but there are several ways to do this, but here is one. So, we already looked at this kind of external questions. So, take any non-trivial component of the forest and then, consider a maximum length path in that.

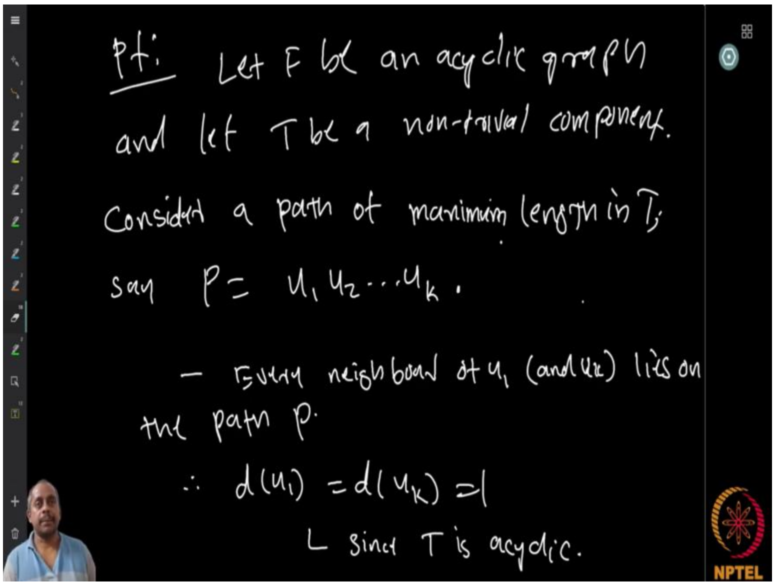
So, given any graph, what you do is that you look at the longest path in this graph and then we will look at the endpoints of this, say, u and v. So, we have this path uv which is the maximum length path in one of the components. Any component that you are taking and then look the end vertices u and v. We are looking at non-trivial components so, there is at least one edge.

Now, what can we say about the degree of u? So, I claim that the degree of u cannot be more than 1, I mean it has to be exactly equal to 1. So, why is this? So, if you look at the vertex u, suppose it has any other neighbour. If it is any other neighbour, let us say u′, the vertex u′ cannot be outside the path because if it is outside the path, we can get a longer path by starting from u′ to u and then continuing on the path.

So, this will give me a longer path, but we started with the maximum length path. So, therefore, u cannot have any other neighbours, I mean outside the path. Now, if it has a neighbour inside the path, then what does it say? Suppose, this neighbour of u is inside the path, then you have some neighbour in the path. Then you already have the connection between these two vertices along the path u-v and then, you have an additional edge which creates a cycle but we started with the assumption that the graph is acyclic.

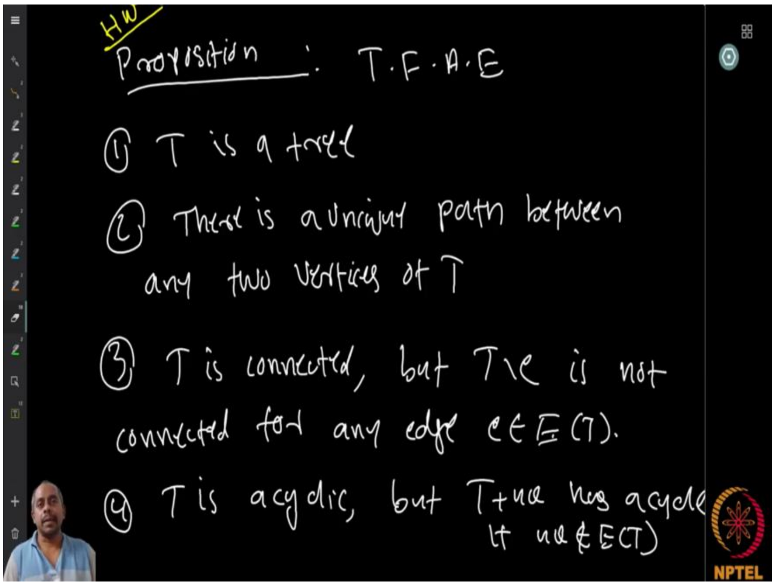
So, therefore this is also not possible which says that the vertex u must have a degree exactly equal to 1. Now, the same argument was for v. So, therefore, u and v must be vertices of degree equal to 1. So, this is true for any maximal paths. So, therefore, we can show that, any non-trivial forest contains at least two vertices of degree exactly equal to 1. We prove it for just a single tree, any non-trivial tree. So, if there are several components, then you can say that there are several leaves in the tree.

(Refer Slide Time: 6:41)



So, here is the proof. Let F be an acyclic graph and T be a non-trivial component. Now, consider the path of the maximum length in T, where P = $u_1 u_2$ ... u_k. Where every neighbour of u_1 it is mentioned and of course u_k also lies on the path P, because otherwise, we said that there is a longer path. If there is a neighbour which is other than the immediate neighbour in the path, then there is going to be a cycle. So, therefore, the graph is acyclic. Therefore, the vertices have degree 1 because the graph is acyclic.

(Refer Slide Time: 7:25)

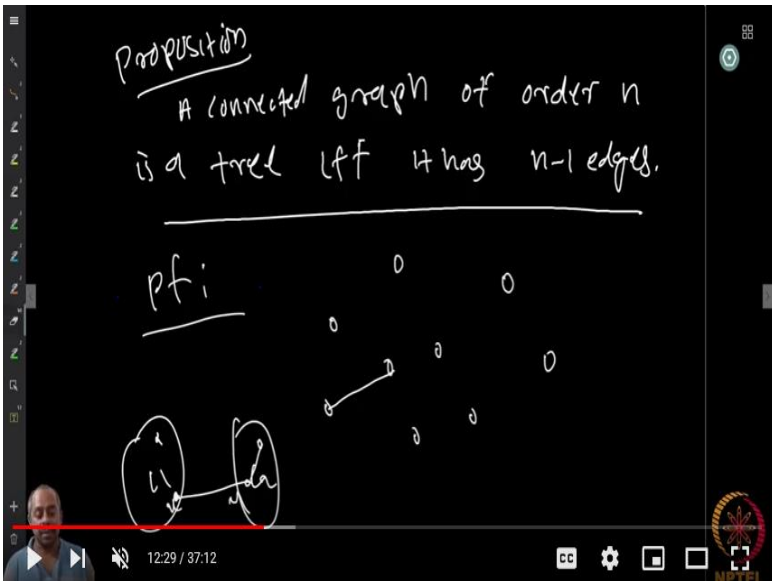


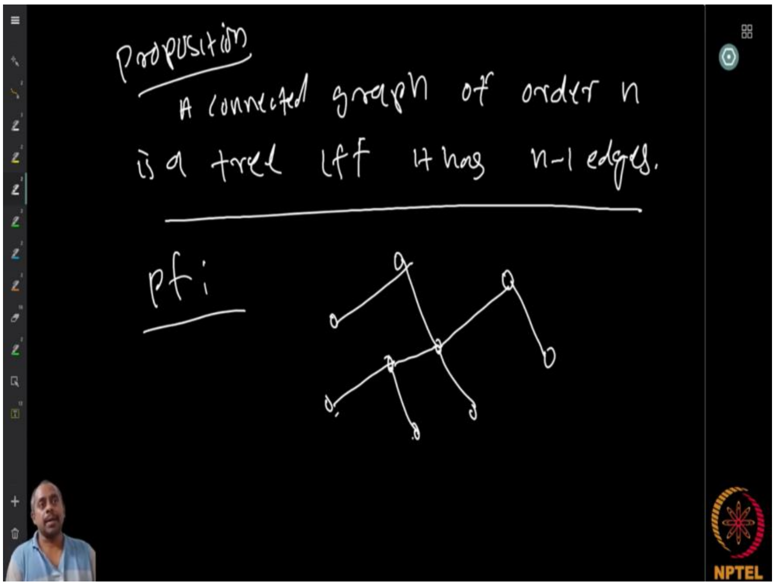
Now, here are some here is a homework question. Show that the following are equivalent. Well, this is a very important result in the study of acyclic graphs. So, these are the statements. The first statement is that T is a tree. So, given a graph T, the graph T is a tree that is equal to the statement that there is a unique path between any two vertices of graph T and they are also equal to saying that T is a connected graph but T\e is not connected for any edge e in the graph. For every edge, T\e is disconnected, but T itself is connected.

So, in this sense we can say T is minimally connected. So, if the graph is minimally connected then it must be a tree and the graph T is acyclic, there is no cycle but if you have any edge which was not present. So, suppose u and v are not adjacent, you have an edge u to v, then it creates a cycle in the graph T.

If this is true for every non-edge u and v, so if uv is non-edge, you have to say it creates a cycle. If that is true for every non-edge, then we say that is also a tree. So, these are all equivalent. So, we have to show that let us say something like,1 implies 2 implies 3 implies 4 implies 1. So, whichever way you want you can show this. But show that these statements are equivalent.

(Refer Slide Time: 9:21)





So, the proposition is as follows. A connected graph of order n, the number of vertices is n is a tree if and only if it has n - 1 edges. So, we are given that like the graph is connected which means that any two vertices there is a path connecting them and it has exactly n - 1 edges. Now, suppose we have a tree, so we have to show two things. Suppose that the given graph is a tree, then we have to show it has exactly n - 1 edges and we have to show that if a connected graph has exactly n - 1 edges, then it is acyclic.

So, here is the proof. So, let us first show that if the graph is a tree, then it has exactly n - 1 edge. So, because it is a tree of course we know that it is connected. Now, let us take an n vertices in this graph without any edges and then add edges one by one. Now, what happens when I add an edge? When I add an edge, if I take two distinct components, two different components which are not connected by an edge and put an edge between them.

If I take any vertex here and any vertex here and then connect them with an edge, then what happens that this entire thing becomes a component because in this component, I can go from any vertex to any other vertex because there is a path by definition and similarly, from this vertex let us say this is x and y, then from y to any other vertex in this component. Now, to find a path from any vertex to any vertex in this big part, all I have to do is that take a path from the starting vertex to let us say u, then take the path from u to x and x to y and from y to any other vertex that we have here. So, this should give us a path between any two vertices.

So, therefore, if I take two distinct components and put an edge between some vertices of these two, then I will get a larger component. So, this is so, it says that if I add an edge between two distinct components, the number of components decreases by exactly one. And of course, it cannot decrease by more than one because an edge only can connect two components.

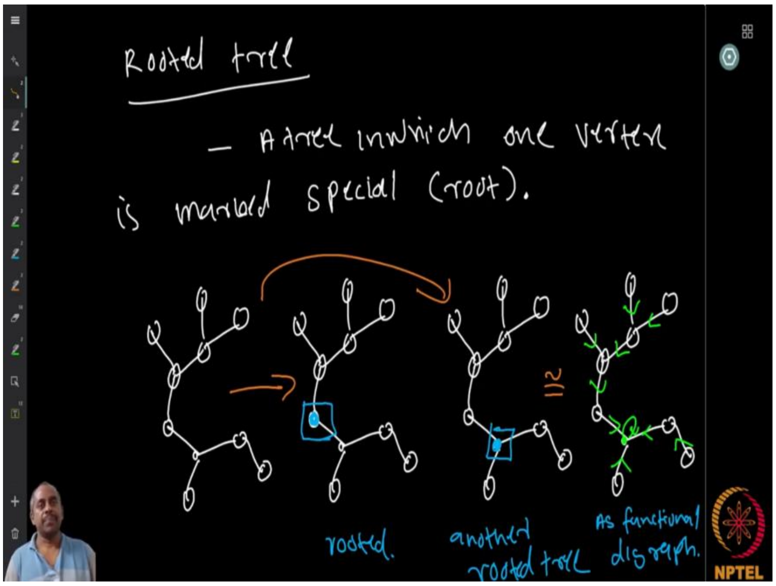
So, if I start with n vertices, then when I add one edge, it can reduce the number of components by one. So, I start with exactly how many components? n components. Then I add one edge, it decreases the component by 1. Then I add one more edge, it decreases the components again by one, if I am connecting two distinct components. Now, if I add an edge between the vertices of a component, it will not decrease the number of components, but it will create a cycle. So, for the time being let us not add such an edge and let us say that we just keep on increasing the edges so that the number of components decreases.

Now, if I do this to minimise the number of components, then because I start with n components, I need to add at least n - 1 edges to make the graph connected. So, every time I add an edge, it only decreases the number of components by one. So, start with the n then n - 1, n - 2 etcetera. If it becomes one, then I have added at least n - 1 edges. So, before adding n - 1 edges, we cannot make the graph connected. So, we need to have at least n - 1 edges. Now can we have more than n -1 edges?

So, at least n - 1 edges are required to make the graph connected. But suppose, you add one more edge so, if the graph is already connected and then you add edge between vertices of some component, then you will automatically create a cycle because there are two disjoint paths between these vertices: the edge and the path in the component.

So, that is not there. So, we see that if the graph is a tree, which means that connected and acyclic, then it has exactly n - 1 edges because it cannot have more than n - 1 edges because once it becomes connected, you cannot add any more edges and to become connected, you can make it more precise if you want.

(Refer Slide Time: 14:59)



Now, here is another observation, that is, something that we defined earlier but we can define in a different way. So, we defined a rooted tree in the earlier phase when we look at the functional digraphs, we defined rooted trees are trees where there is exactly one cyclic vertex. So, those were called the roots and then we have that rooted tree as this functional digraph. Now, we can also view rooted tree by the following.

We have our undirected graph and you just make one of the vertices as special vertices and call it a root, that becomes a rooted tree. So, this is the natural or the first way that I think you can go for the definition of rooted trees because you take a tree and then consider one of the starting vertex or initial vertex as special vertex which we call root.

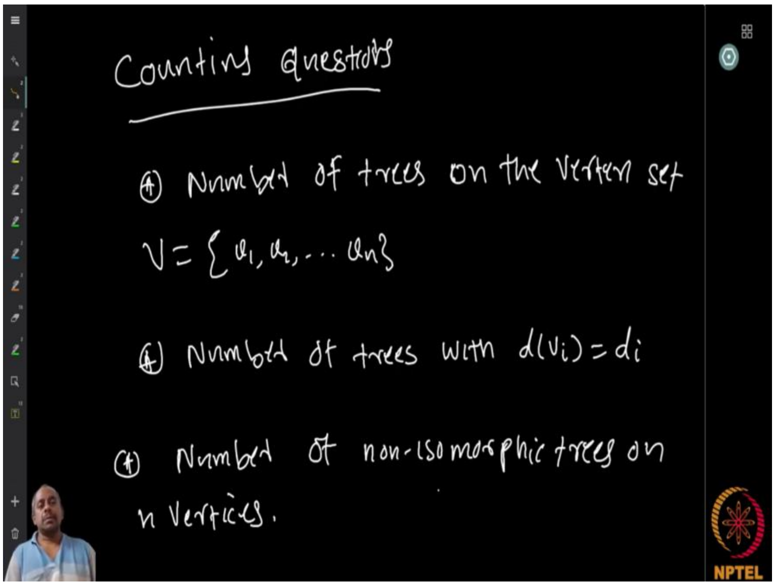
So, you take any arbitrary tree, pick any vertex and make it a special vertex and say that this is a root. So, that becomes a rooted tree. We can start from a vertex and pick any of the vertices to make roots. So, this graph that we get this is the rooted tree where this vertex is special and in fact, I take this same graph and make this as special and then I get another rooted tree.

Now, there is a one-to-one correspondence between the rooted trees that we come up like this and a functional digraph which is rooted because we observe that in the functional digraph, if you have a cyclic vertex, then every path from other vertices must lead to this vertex. It should all be coming into this because this vertex already has an out degree right. So, it cannot have any other outgoing edges. So, therefore, every edge to it must be incoming.

Now, because of this, this vertex's neighbours must also be incoming edges. So, we can follow this argument and show that once you fix a root, then all the path must converge to this cyclic vertex or that. So, that is how we get the functional digraph. So, that rooted tree is exactly, if you are into this because all you are doing is that, whatever is once you have finished the root every path converges basically from every leaf, you go find a path to this vertex which is directed path.

So, because that is a unique way to do it, we see that there is bijection between these two. So therefore, we can now call any of these as rooted tree without worrying about which definition it comes from. And we can use it as it is and we can go from one to the other and this. So, this is a simple observation.

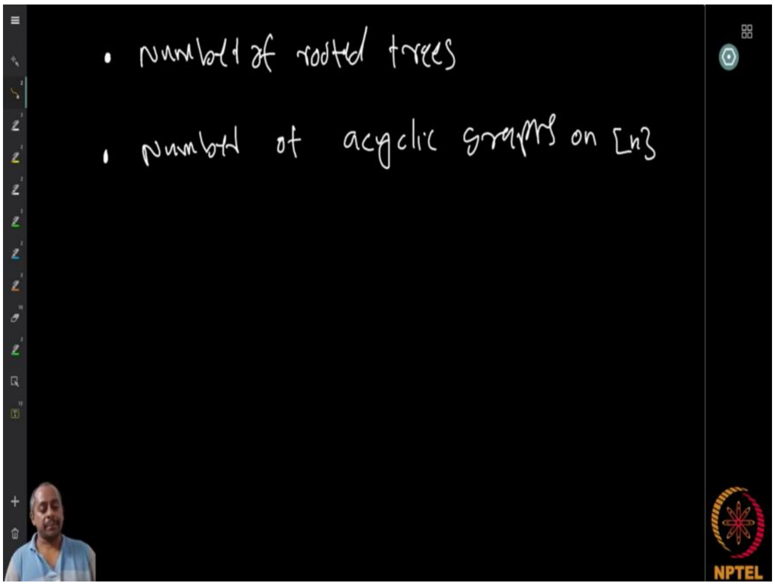
(Refer Slide Time: 18:34)



Now, what are the main questions that we are interested in Combinatorics is of course counting questions. So, here are some natural counting questions that arises, of course, there are many more I just mentioned a few. So, one question is that what can we say about the number of trees on the vertex set V, let us say. So, given a vertex set V= $\{v_1, \ldots, v_n\}$, how many distinct trees you can make on this. Now, the number of trees, let us say that where each vertex degree is now specified, let us say v_i have a degree exactly d_i.

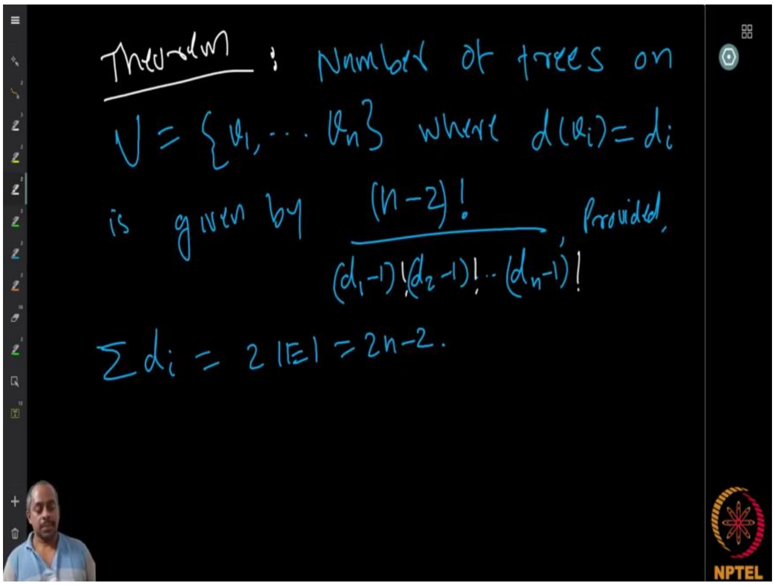
Now in this case, can we count the number of trees? Or if you say that the number of non-isomorphic trees on n vertices because when you have labelled a tree, there will be isomorphism between the corresponding trees, some of them. So, if there are isomorphism, when we do not want to count all of them, we want to guess how many trees are there which are equivalent under the isomorphism. I mean, distinct under the isomorphism. So, how many classes are there under this equivalence relation of isomorphism? So, that is another question one can think about.

(Refer Slide Time: 20:00)



You can ask number of rooted trees, number of acyclic graphs. All these we can ask about just the acyclic graphs. So, these are all questions, some of these we will try to tackle now and some of these we try to work out when we look at further topics like or things like that we may find the easier way to do this. But we will look at a couple of things at the moment.

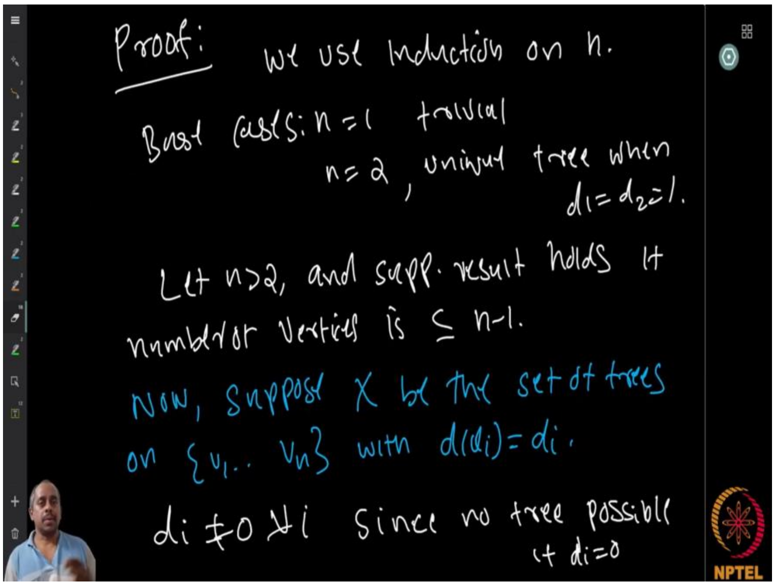
(Refer Slide Time: 20:48)



Now, here is one of the questions. So, what was this question that number of trees with degree of v_i is equal to d_i. So, here is a theorem which explains or counts this. So, number of trees on the vertex set V, where degree of $d(v_i) = d_i$ is given by $\frac{(n-2)!}{(d_1-1)!\,(d_2-1)!\,\ldots\,(d_n-1)!}$

Where $\sum d_i = 2\,|E| = 2n - 2,$ because the tree has exactly n - 1 edges. So, now how do you prove this?

(Refer Slide Time: 22:20)



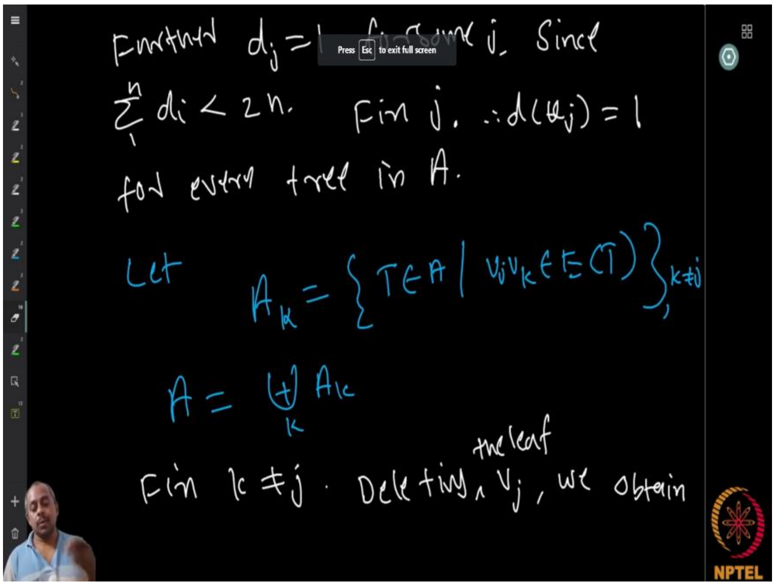
So, here is the proof. What we are going to do is to use induction on n. So, the base case, n = 1 is trivial because there is no edge, there is one tree and that is it. I can say that the formula holds and for n = 2 again there is a unique tree. What is the unique tree? The single edge, that is a tree and this happens when $d_1 = d_2 = 1$ and in other cases $d_1 = 2 \; and \; d_2 = 0$ we can see that there is no tree.

Something that you can immediately see. So, therefore, again we can say that the formula holds what just verified it. So, once you have the base cases, let us assume that n is strictly greater than 2 and now suppose the result holds if the number of vertices is less than or equal to n - 1. So, if the number of vertices is strictly less, then it holds. Now what we will do is that we take the set of trees on the vertex set v_1 to v_n with the $d(v_i) = d_i$.

So, consider that and call it as X. So, we can observe that di cannot be 0 for any i. Because we are talking about trees and because they are connected, this is one way to look at this. We cannot create any tree if $d_i = 0$, for some v_i because it cannot be connected, as far as we are not looking at the trivial tree where n = 1, we are assuming that n is strictly greater than 2.

So therefore, we can think of this and say that now, d_i cannot be 0, because the graph is disconnected, so this is no tree possible. So, if $d_i = 0$,, we will see that like in this formula we will get the $(d_i - 1)!$, that becomes problem in our definition that does not make sense. So, that is not possible. So, therefore, $d_i = 0$ is ruled out.

(Refer Slide Time: 25:14)



So, suppose $d_j = 1$, for some j. This we can always assume. Why? Because, suppose every d_i was greater than or equal to 2, then we have n d_i's so, therefore, the sum is going to be greater than 2n - 2. If every vertex has degree at least 2, there are n vertices, total degree will be more than 2n. So, therefore, we know that since d_i is non-zero, it must be equal to 1 for some vertex so that is why d_j is the vertex whose degree is 1, that is at least 1. So, let us fix this particular vertex v_j whose degree is going to be 1.

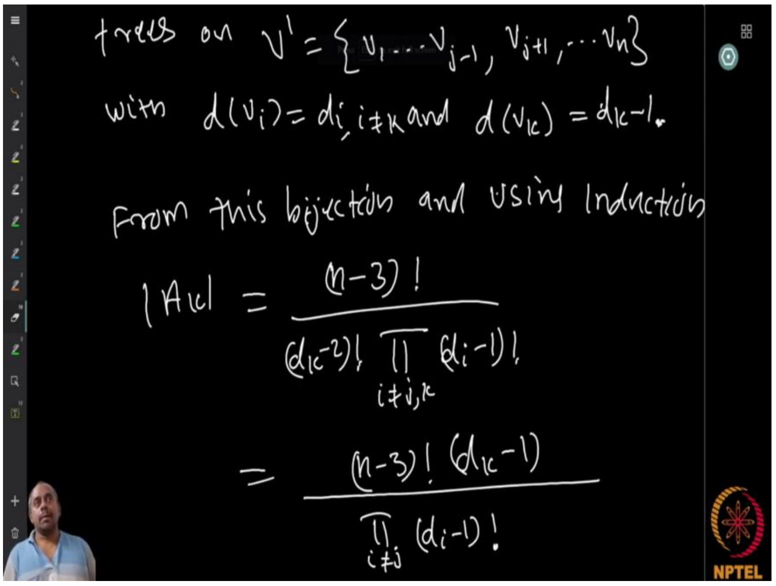
Now, what we notice is that, by the definition of the way we construct this $d(v_i) = d_i$, for all the graphs in X. So, it means that, for the graph that we are going to consider in X, the degree of v_j is going to be always 1. So, v_j is going to be a leaf in all the trees in X. So, what we can see is that in all the trees that we are considering, v_j is always going to be a leave. It means that it has exactly one edge. Now, we collect these neighbours and then make another set like this.

For every tree in A, let A_k is the set of all trees such that $v_j v_k$ is an edge in the tree. Because v_j has a unique neighbour. It is going to be exactly one v_k for a given tree. So, that k it will be different from j because there is no loop in trees. So therefore, for every k's, we can consider the possible case. So, A_k basically the set of all trees such that $v_j v_k$ is in the edge of T. So, we are now collecting with the index of the neighbour of this particular vertex v_j that could be different in different trees.

But of course, the degree of v_j is actually equal to 1 but the neighbours could be different in each of the different trees. Now, because this A_k must cover all of A in at least some of these cases, A is going to be disjoint union A_k and this is disjoint union because you cannot have two different k intersecting. There is a precisely unique neighbour for v_j. So, therefore we get a disjoint union of A_k which is equal to A. Now, let us pick some k which is different from j.

So, what we do is that we are going to delete this single degree vertex which is a leaf v_j . Once I delete the vertex v_j, what happens? Because it is a leaf, it is not connected to anything other than its unique neighbour, and every path from this vertex (because the tree is connected) to every other vertex is actually going through this neighbour. So, if I remove this particular leaf, the remaining graph is still going to be connected, it is something that you can immediately see from the structure of a tree, if you remove the leaves, it does not affect the connectivity of the remaining part.

(Refer Slide Time: 29:21)



So therefore, we are going to get trees on the vertex at V \ $v_j = v_1, \ldots v_{j-1}, v_{j+1}, \ldots, v_n$. So, on n - 1 vertices, we get trees which are again connected and what is the property of the trees that we are proving this way? For all the trees, $d(v_i) = d_i$ because we have not changed the degree of anything except for v_k. So, for i ≠ k, $d(v_i) = d_i$.

Now once you observe this, what happens to degree of v_k? Degree of v_k reduces exactly by 1, because we just removed one of its neighbours. So, it is going to be $d_k - 1$. Now, we can see that there is a bijection between the trees A_k that we are looking at and the set of trees that we obtained here, by deleting this. Because we precisely have the same thing here except, we removed one leaf and if you have anything here with this degree sequence, we can also add a leaf make it adjacent to one of the vertices and then we will get a tree in A. So, it must be present in one of the A_j .

So, therefore, there is one-to-one correspondence so, the cardinality of A_k is precisely the cardinality of the trees that we obtain this way. Now the trees that we obtained this way is all trees that we can get where the degree sequence is given here, all trees on n -1 vertices where the degree sequence is given here, here $d(v_i) = d_i, i \neq k$ and $d(v_k) = d_k - 1$ So, now we can use induction. So, what is the induction?

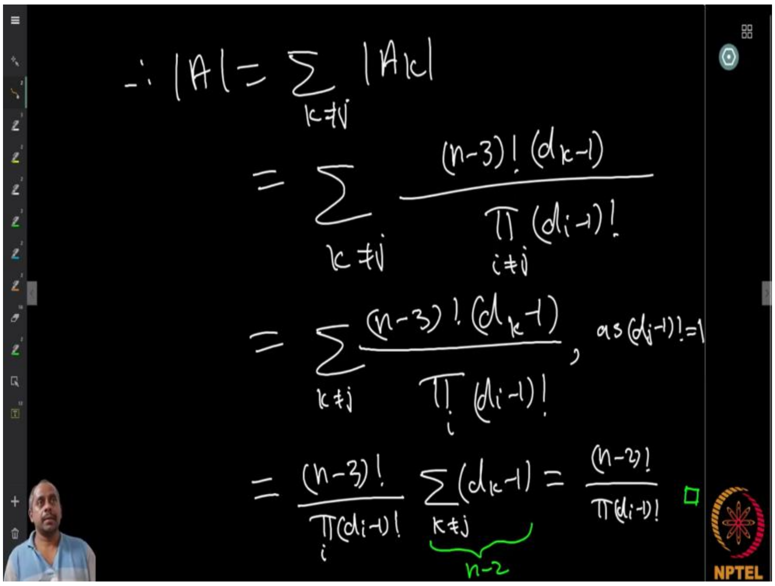
$$|A_k| = \frac{(n-3)!}{(d_k-2)!\prod_{i\neq k,j}(d_i-1)!}$$, multiply $d_k - 1$ above and below.

$$= \frac{(n-3)!(d_k-1)}{\prod_{i\neq j}(d_i-1)!}$$

Now, here is something that we should observe that so, i not equal j, there is one term still missing from our general formula, but what is that term? That term is $(d_j - 1)!$. But we know that $d_j = 1$. So, $(d_j - 1)! = 0! = 1$.

So, I can also put that thing here, it does not make any difference because d_j is 1. So, then I can write it as $\prod_i(d_i - 1)!$ because it does not matter whether I add that particular case of d_j. So, therefore, I will get this without i not equal to j.

(Refer Slide Time: 33:38)



Now, so, what is the cardinality of A? Well, it is a sum over all k different from j, cardinality of A_k , by definition here right. It is the union over all k, A_k.

$$|A| = \sum_{k\neq j} |A_k| = \sum_{k\neq j} \frac{(n-3)!\,(d_k-1)}{\prod_{i\neq j}(d_i-1)!}$$

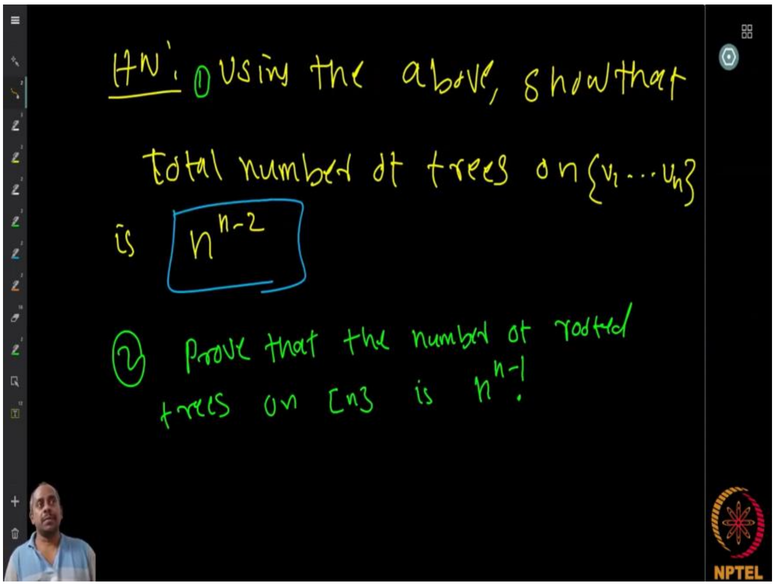
$$= \sum_{k\neq j} \frac{(n-3)!(d_k-1)}{\prod_i (d_i-1)!}, \text{ since } \left(d_j-1\right)! = 1$$

$$= \frac{(n-3)!}{\prod_i (d_i-1)!} \sum_{k\neq j} (d_k-1)!$$

$$= \frac{(n-2)!}{\prod_i (d_i-1)!}, \text{ since } \sum_{k\neq j} (d_k-1)! = n-2$$

So, we have proved that. Now, since we have this we can use this to find the number of all the trees on $\{v_1, \ldots, v_n\}$. Because for each degree sequence now we know how to find it so, you can sum all this thing again.

(Refer Slide Time: 36:26)



So, that is a homework for you. Show that the total number of trees on $\{v_1, \dots, v_n\}$ is n^{n-2} by using the above result, that for a given degree sequence we have $\frac{(n-2)!}{\prod_i (d_i - 1)!}$. So, that sum should be equal to n^{n-2}. So, show this by using the previous result. Now, again once we have this, we can immediately show that the number of rooted trees is on the vertex set 1 to n is n^{n-1} and this is also the immediate consequence once you have it, this is easy.