Graph Theory

Prof. L. Sunil Chandran

Computer Science and Automation

Indian Institute of Science, Bangalore

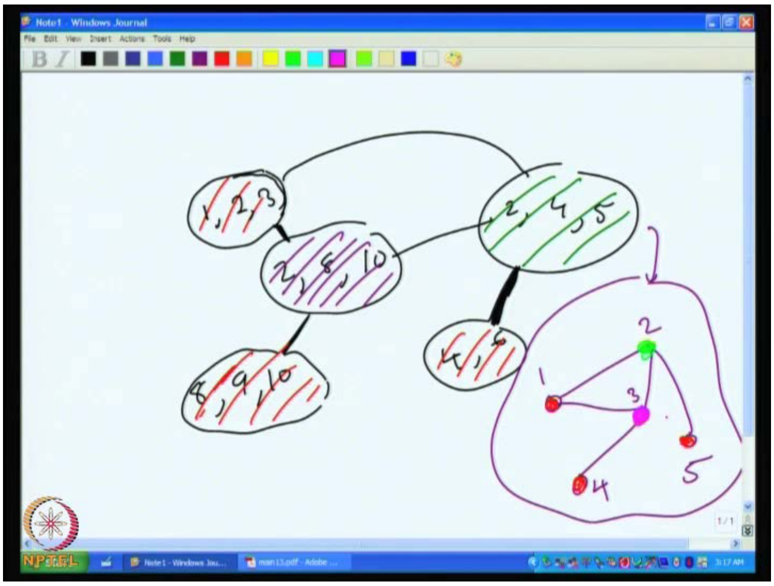
Module No. # 03

Lecture No. # 13

Vertex Coloring: Brooks Theorem

Welcome to lecture number thirteen of graph theory. Today, we will consider a new topic namely graph colouring. There are two types of colourings that we will consider - the vertex colouring and edge colouring. First, we will deal with vertex colouring and then go to edge colouring.

(Refer Slide Time: 01:04)



To introduce the problem, let us consider this following question. Suppose, there are some committees and we will say the members of the committee that is a set. So, we can say first, number 1, 2, 3 belongs to a committee and say 2, 4, 5 belongs to a another committee, say 4 and 6 form another committee, 8, 9, 10 forms another committee and 2, 8, 10 forms another committee like this.

Now these committees meet. When a committee meets, it takes about one day to finish - full day meeting. We have to schedule the meetings of all these committees, but the problem is that some committees contain the same person. For instance, committee 1, 2, 3 and 2, 8, 10 have 2 in common.

Therefore, the day this first committee meets, These two committees cannot meet on the same day. This committee and this committee cannot meet on the same day. So they have to meet on different days. So, the question is how many days are required. How will you solve this problem?

First, we would like to convert to a graph theory problem. The question is, for instance, why cannot two committees meet on the same day? It is because there is a common person. Therefore, we will consider each of this committee as a vertex of a graph and then if there is any common member in the committee, we will put an edge between the corresponding. So, this is the vertex now; this is a vertex now. We put an edge between them. So, between this and this, there is an edge, between this and this, there is an edge, between this and this, there is an edge, between this and this, there is an edge and see 4, 6. Now, there is nothing common here. Therefore, these are the edges. It means that whenever there is an edge between two vertices, the committees corresponding to those two vertices cannot meet on the same day.

Now, we will find out the committees, which can meet on the same day. For instance, this committee and this committee can meet on the same day because there is no edge between them and is there any other committee, which can meet on the same day. This and this can meet on the same day because they do not share anybody common.

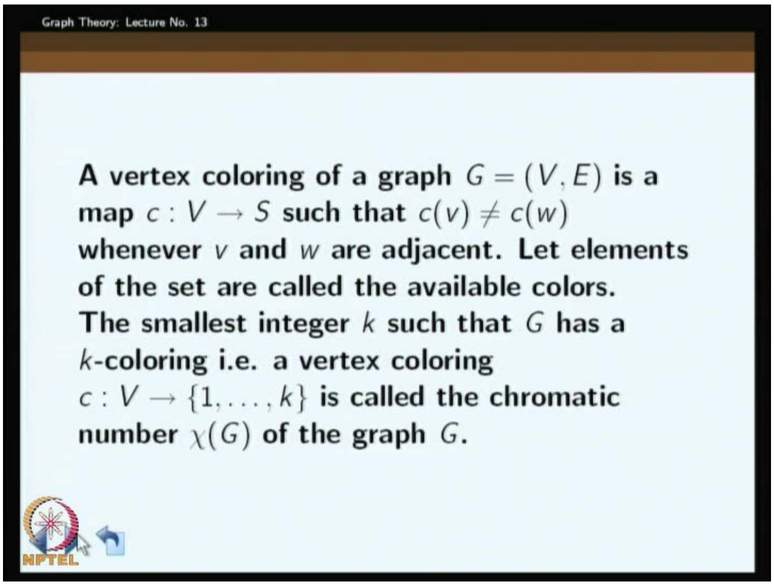
Now, this cannot meet on the same day, this also not. Of course, now, we have to use another day. So, this red corresponds to the first day and another day to schedule this meeting. Then same day, this cannot happen. So, we need a third day for this meeting. Here I used some colours to find The red colour indicates the first day, the green colour indicates the second day and this violet colour indicates the third day.

These three colours correspond to three days. So, we need three days to finish all these meetings, to schedule all the meetings. So, this corresponds to a graph colouring problem. Here, we were looking for the minimum number of colours, which correspond to the number of days required.

The colours are given to the vertices of the graph such that whenever there is an edge between the two vertices, they should not get the same colour. This is the kind of thing. So, here the graph colouring graph We could have drawn the graph more neatly like this. This is our graph.

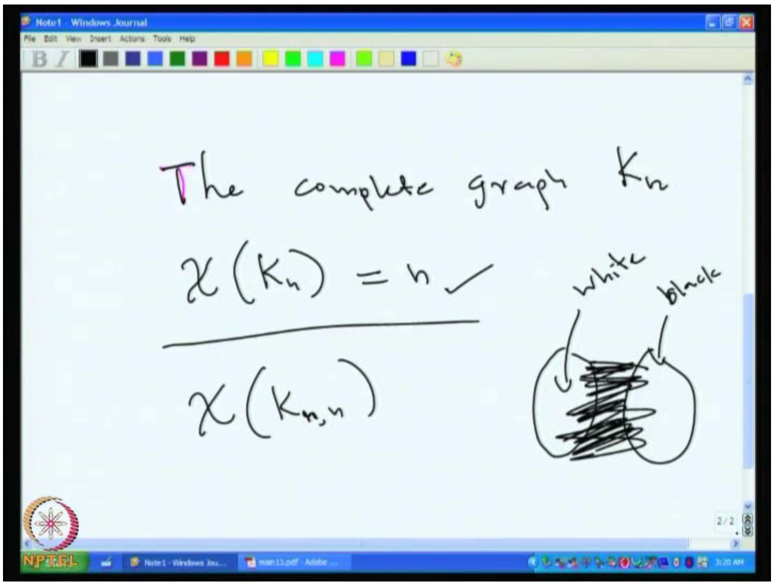
So, this is the graph. The vertices correspond to the committees and the colours will correspond to the vertices. This is the colouring problem. The red colours is given If you look at the red colour vertices, they form an independent set. That means there is no edge between them. So, here, this is the colouring and then here, we have a violet colour.

(Refer Slide Time: 06:10)



Therefore, let us formally define graph colouring problem. A vertex colouring of a graph G equal to V comma E is a map is a function c from V into S, S being the available colours such that c of v not equal to c of w, whenever v and w are adjacent. So, let elements of the set S are called be called the available colours. The smallest integer k such that G can have a k colouring, that is a vertex colouring from V to 1 to k is called the chromatic number, khi of G of the graph. This is the formal definition of this. In other words, we are supposed to colour the vertices of the graph in such a way that whenever there is an edge, they should get different colours and the number of colours should be minimized.

(Refer Slide Time: 07:18)

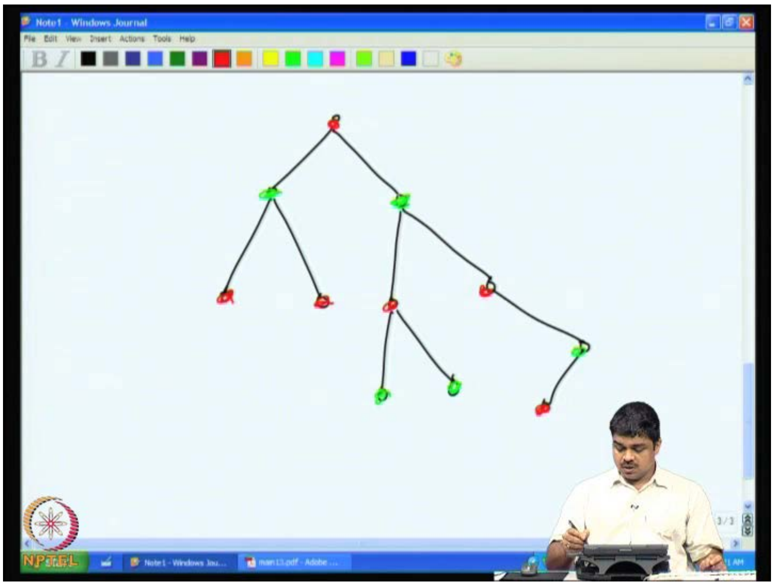


We can look at some examples and try to understand the colouring problem again. So, for instance, let us take our favourite example, the complete graph K n. So, the minimum number of colours that we will use will be called the chromatic number. The notation is khi. So, we are interested in khi of K n. What will be minimum number of colours?

It is very easy to see that it has to be n because you cannot share any colour because any two vertices if you consider, they should get different colours in the complete graph because there is always an edge between them. Therefore, they should get n colors.

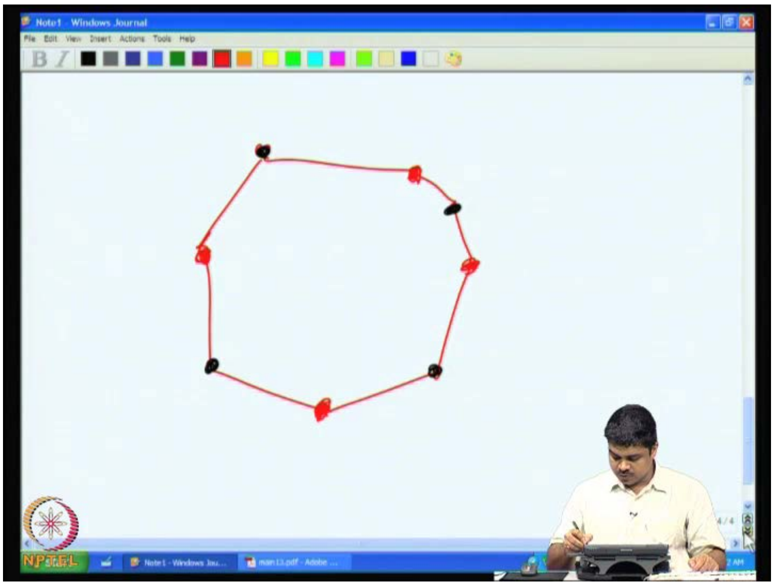
Now, what about a complete bipartite graph K n, n. This also has lot of edges, but still it happens that it requires only two colours. Why is it so? Why it require only two colours? So, this complete bipartite graph is like this. n vertices and then you have all the connections between them; so, all the connections between the two sides. So, this side can be given say, white colour and this can be white colour and this can be given black colour.

(Refer Slide Time: 08:52)



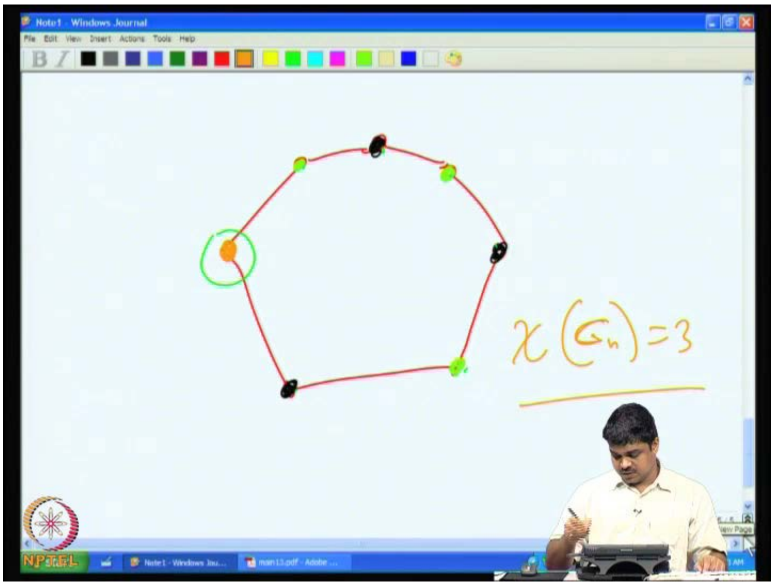
So, this is the complete bipartite graph. So, two colours are enough and now we can look at some other cases. What about trees? For instance, let us look at this tree. How many colours are required to colour the tree? Some thought will reveal that only two colours are required because you can colour it red and you can colour it green here, then you can colour it red, layer by layer then you can colour it green here, like that.

(Refer Slide Time: 09:34)



So, this requires only two colours and what about a cycle. This is the cycle. So, whenever we say a cycle, even and odd cycle matters. This is an even cycle. How many colours are required? Say, a little consideration will reveal that only two colours are required. So, we can give this colour black and the other colour say, we can give red, we can give. Using two colours, we can colour this one and what about an odd cycle?

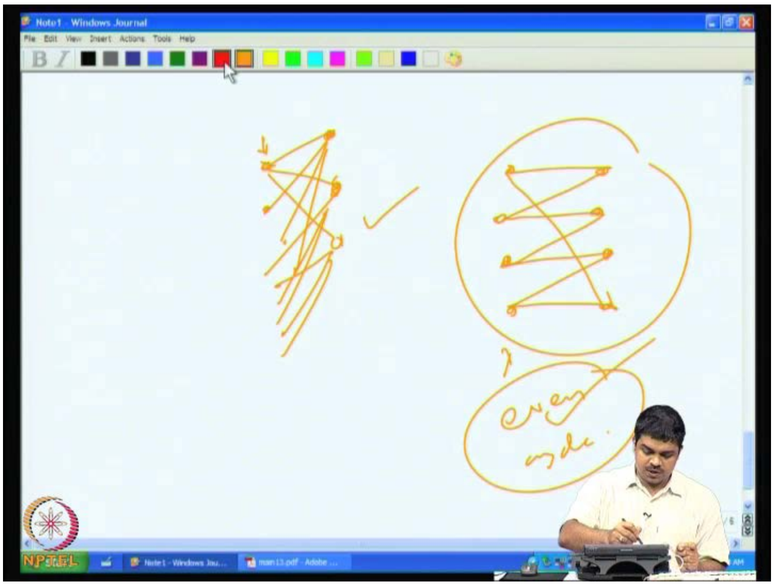
(Refer Slide Time: 10:17)



Odd cycle may be like this. So, you can try to colour like this, like I did in the last case - this one, this one and this one, this one. So, here I have a conflict because I have green here, then black here, green here, black here, green here, black here. Here, what will I do? Here, I cannot give green; I cannot give black. So, I will have to use a new colour. So, you can see, it requires three colours - khi of C n equal to 3, when n is an odd number.

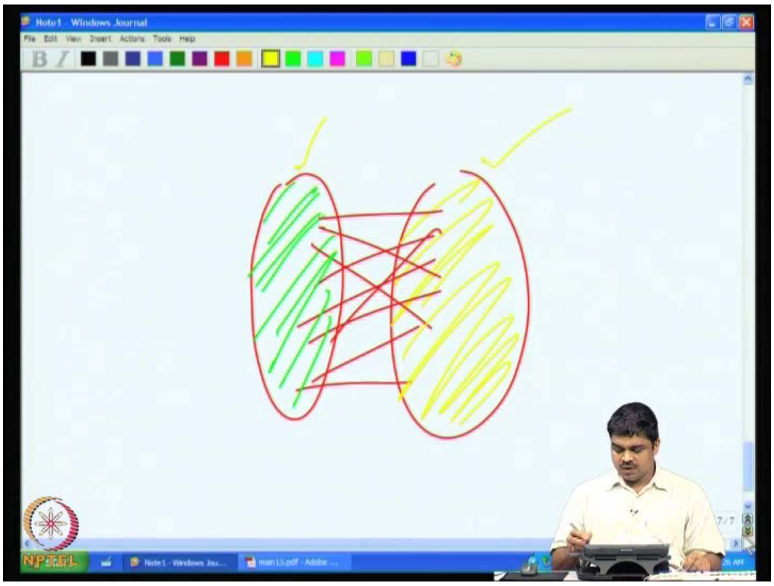
If it is an odd length cycle, then we need three colours. So, between the odd cycles and even cycles there is a difference; odd cycle requires three colours and the even cycles required only two colours. So, when you consider The couple of examples, we considered - the even cycle and trees, both of them required only two colours. They are, if you remember, what the definition of the bipartite graph, both of them are bipartite graphs.

(Refer Slide Time: 11:51)



So, why are both of them bipartite graphs? It is because the tree is like this. So, we can consider the first route of the tree here, the immediate neighbours on the other side and then the immediate neighbours of that on this side, like that. So, tree is a bipartite graph. Similarly, even cycle is a bipartite graph because we can always draw this cycle like this. The even cycle can be considered like this.

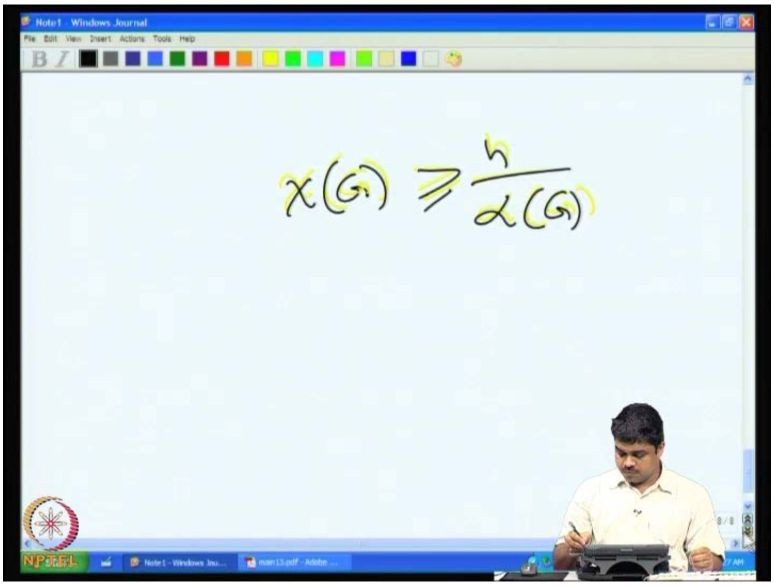
(Refer Slide Time: 12:39)



So, both of them are bipartite graphs. So, it is not surprising that they require only two colours. Why is it so? Because if you take any bipartite graph, it will look like this - two parts and the edges always going from one side to the other. Here, it is like, you can use one colour for this side and the other colour for this side because edges are always from one side to the other. It so happens that all bipartite graphs are two coloured – essentially, two parts. So, it is not very surprising because if you look at the colouring question carefully, it so happens that when you look at a colour class - a colour class means the vertices which got same colour, they should form an independent set in the graph and it is because there is no edge between two vertices of the same colour.

So, it is like, we are trying to cover the graph with independent sets. Each colour corresponds to an independent set; another colour corresponds to another independent set, like that. We just have to put every vertex in one of the classes so that we are covering all the vertices with independent sets. What is the minimum number of independent sets required to cover the vertex set? That is what we are interested in. In a bipartite case, there is clear that we have two independent sets and then it covers the entire graph.

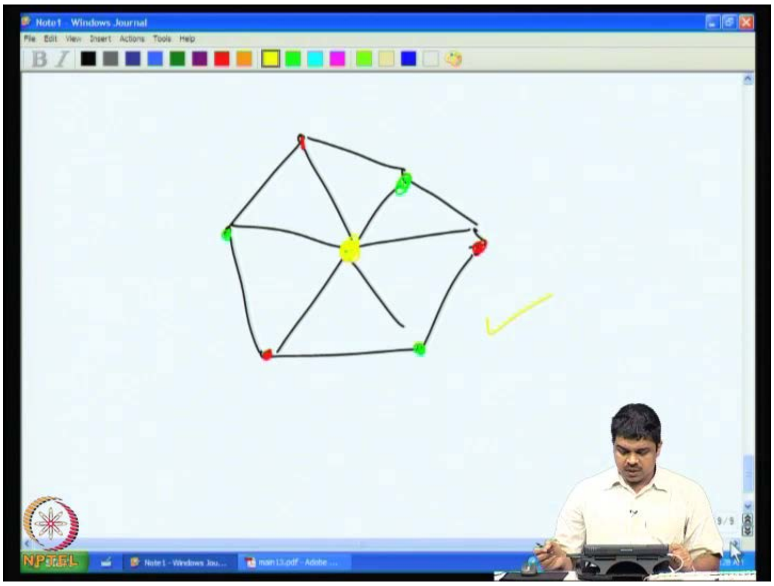
(Refer Slide Time: 14:24)



So, in a clique, we have only independent sets of size one. Therefore, you need n of them. So, this will give raise to this thing. If n is the number of vertices, khi of G has to be greater than or equal to n by alpha of G. khi of G has to be greater than or equal to n by alpha of G. Why because each colour class can contain only a maximum of alpha of G vertices.

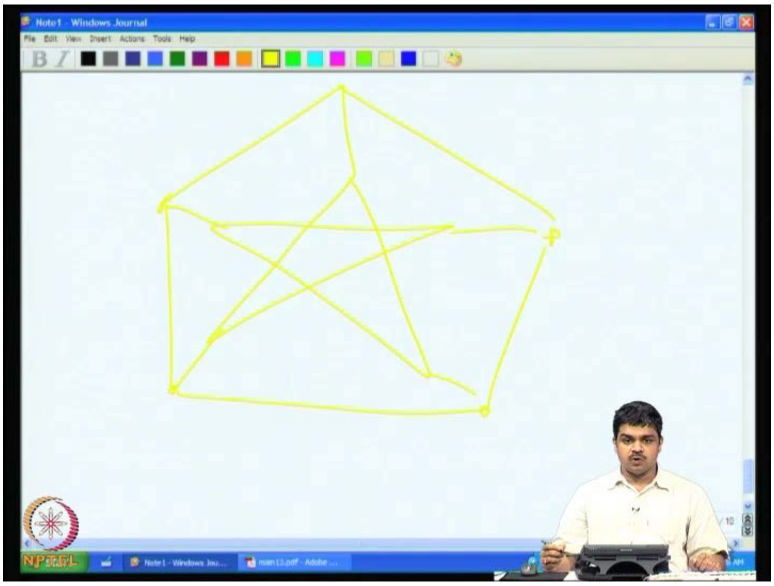
So, one colour class contains only alpha of G, but there are n vertices. So, we should get at least so many colours; there should be at least so many colours. That is what we can tell about the relation between independent set and the chromatic number and number of vertices in the graph.

(Refer Slide Time: 15:26)



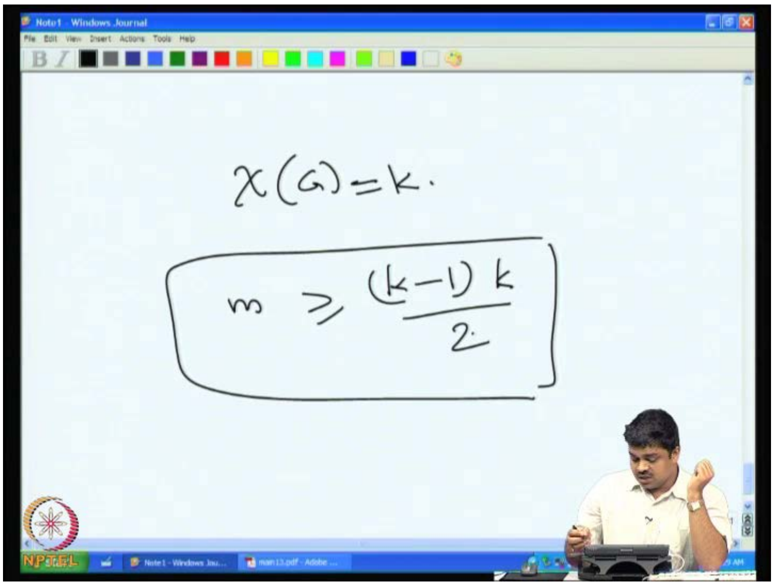
Now, of course, you can take some little more complicated examples. For instance, if there is a wheel, how many colours are required? Of course, it depends on how many colours are required. This will always require a new colour. For instance, here this was green; say, we may use this one and we may use this colour and then we may use this colour and we may use this colour here. This is and Then we will have to give a new colour. So, this colour is not possible or neither red colour is possible. So, you may have to use a different colour for the middle vertex.

(Refer Slide Time: 16:14)



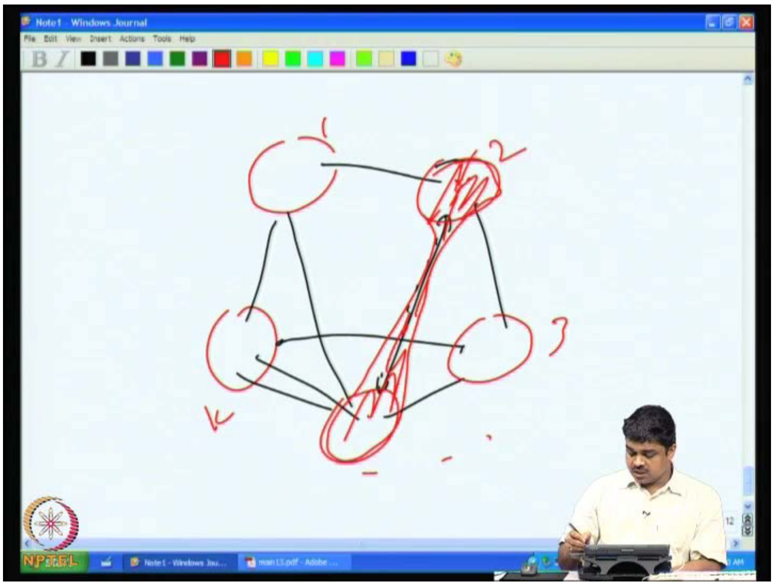
So like that we can consider several examples. Another example you may want to consider is the Peterson graph. how many What is the chromatic number of Peterson graph? This is the graph. How many colours are required to colour this thing? I leave it you to figure out how many colours are required for this graph.

(Refer Slide Time: 17:34)



Now, let us look at Suppose, we want to come up with an algorithm to colour the graphs; not necessarily optimum, some colouring is required. So, we want some method to come up with the colouring. What should be the easy and very straight forward strategy to come up with colouring for a given graph? So, this is like this. Before that, let us consider another question. What about the number of edges? For instance, can I say that if the chromatic number of a graph is k, khi of G equal to k, then can I somehow say that the number of edges m in the so number of edges m in the graph is greater than or equal to some function of k. Is it possible?

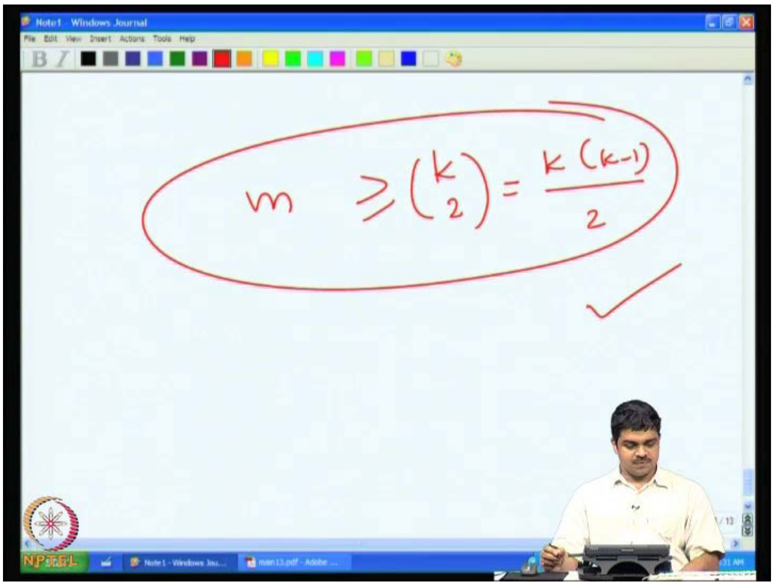
(Refer Slide Time: 18:08)



What we can show is that m will always be greater than or equal to k into k minus 1 by 2. If k is chromatic number, there should be at least k into k minus 1 by This k choose to edges in the graph. Why is it so? Because if we consider the colour classes, these are independent sets, the colour classes. There are k colour classes; there are k colour; the first colour class, second colour class. So, there are k colour classes.

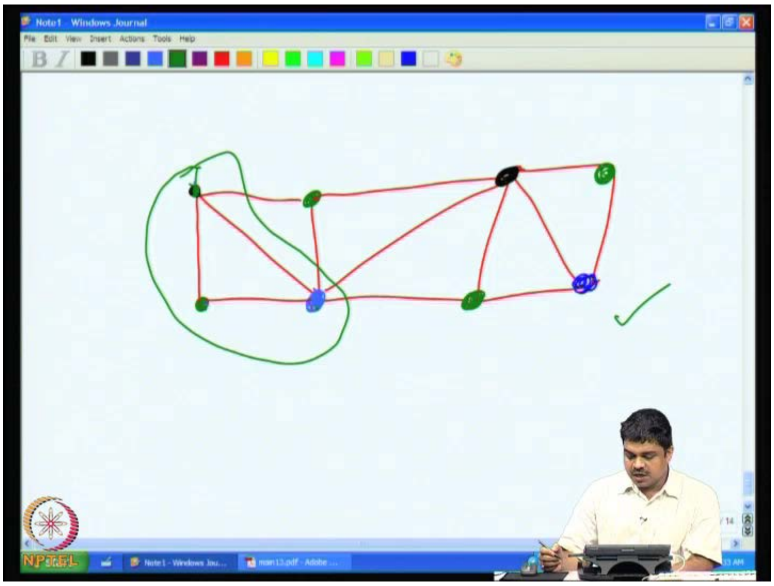
Between any two colour class i and j, there should be at least one edge. Why is it so? Suppose, this edge was not there, there should be at least one edge between any two colour classes - at least one edge, may be more. Suppose, it is possible that there is no edge between these things, then I would say, you merge these two colour classes and you can make it a new colour class together because anyway there is no edge between them. Why do you want to give two different colours to the vertices of this and this.

(Refer Slide Time: 19:07)



Therefore, what we infer is between any two colour classes, there is an edge. There are k colour classes. Therefore, k choose two edges should be there; k choose two because there are k choose two pairs of colours; that is k into k minus 1 by 2. So, this will be the lower bound for the number of edges. This is what I am saying - k into k minus 1 by 2.

(Refer Slide Time: 19:59)



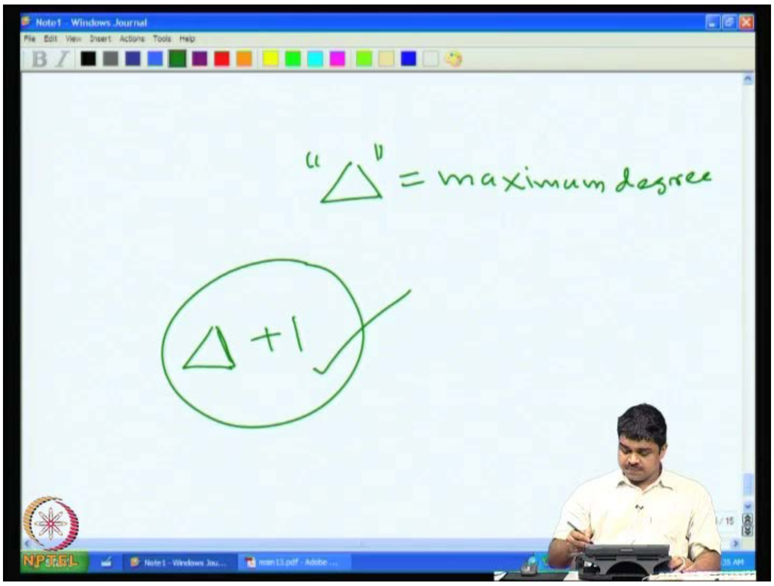
Next question, I am considering is how do we come up with a colouring for the graph? What should be an easy way to colour the graph? Suppose, let us take this graph, maybe we were looking at the Peterson graph or maybe this graph. I am just drawing a graph.

So, I want to colour this graph. One strategy I may want to adopt is to give a colour for an arbitrary vertex and then I take another vertex. Say, I took this vertex and I see that this is already coloured, then I can give a different colour to this.

Now, I take this vertex, I see that these two colours are already used. I may want to use a new colour for this. When I go here, I see that this black colour and blue colour is used, but green colour is available. So, I can use the green colour? Whenever I see a new vertex, what I do is I look at its neighbours which are already coloured and I try to reuse a colour, if there is anything available. If all the colours which are up to now used is there in the neighbourhood, then we cannot re-use in a colour. We have to use a new colour; this way, we keep going.

Here, we can use the black colour; here, we can use the green colour; here we can use the blue colour; Again, here we can use the green colour once again. So, this is the strategy of colouring. So, one question that comes to your mind is why is this an optimum colouring? In this case, it looks like that is an optimum colouring. The three colours are required here. Why because if you take a clique triangle here, this requires all the three colours because you cannot repeat the colours here anyway. Therefore, in this case, it is an optimum colouring, but in all cases, it need not be an optimum colouring.

(Refer Slide Time: 22:58)



In some cases, it may so happen that we end up using too much, therefore, but still we can try to come up with in it with some kind of an upper bound like saying that whatever it is, it may not be optimum, but we have only used these many colours. So, what kind of upper bound is possible?

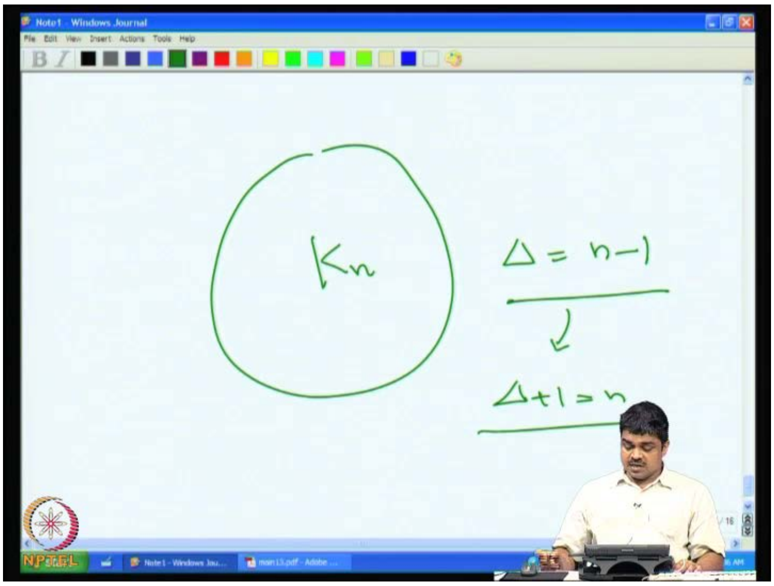
So, the easiest way to get an upper bound is to note that whenever I consider one new vertex for colouring, what it does is it looks at its neighbours, which are already coloured, but how many neighbours can it have maximum. We can have only maximum of delta neighbours, delta being the maximum degree; delta neighbours only, it can have.

Therefore, in whatever way, we can see only at most delta colours on its neighbours. Now, it will use a new colour It is possible that you may use an new colour. So, delta plus 1 colours, it will use any time.

In another case, if you see delta neighbours, this delta plus 1 colour is already there in the graph, it is already given to some vertex in the graph, we can reuse that colour; there would not be any conflict. Therefore, it is so happens that you can colour the graph with at most delta plus 1 colours. It will not be possible. I mean there will not be any situation where you will have to use more than delta plus 1 colours.

But again if you carefully look at it, it may come to your mind that maybe, it is a little too much. So, you may not use delta plus 1 colours in many situations. Is there any particular situation when you really use delta plus 1 colours? A tight situation, a tight example like we have to use delta plus 1 colours.

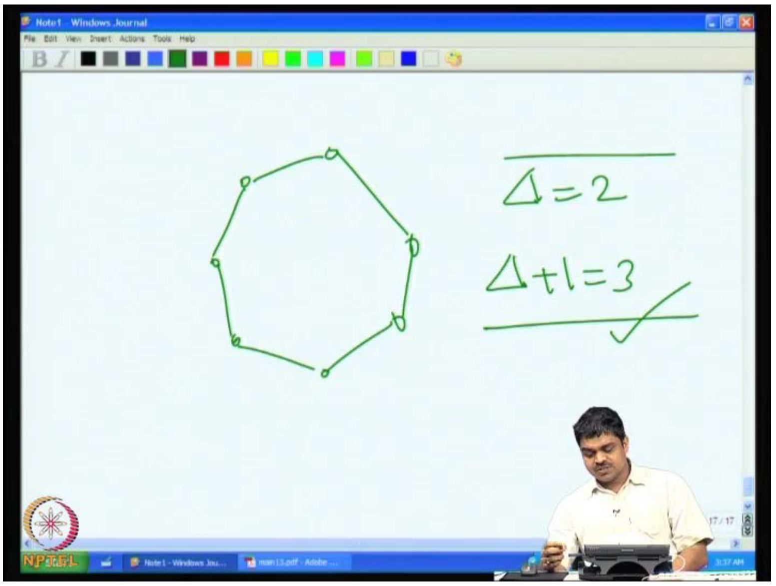
(Refer Slide Time: 24:29)



So, one such example is the complete graph. For instance, if you consider a K n complete graph on n vertices, what is delta? delta equal to n minus 1. Now, how many colours are required? delta plus 1 colours are required which is equal to n colours are required.

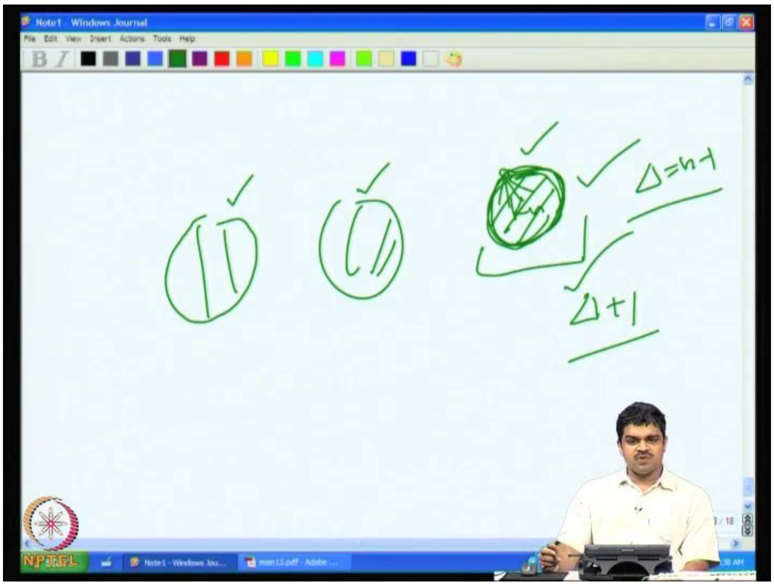
You cannot avoid this situation here because if I start doing, however you try, you will end up using delta plus 1 colours. So, there are situations where you need delta plus 1 colours. Can you quickly think of another example where delta plus 1 colours are required?

(Refer Slide Time: 25:06)



One such example is this, say, an odd cycle - cycle with odd number of vertices 3 plus 3 plus say 7 cycles - odd cycle, odd number of cycles. How many colours are required? delta equal to 2 here - maximum degree is 2. The numbers of colours required is delta plus 1 equal to 3. So, this is also one such situation where delta plus 1 colours are required. So, the odd cycle and complete graphs are some cases, some examples, where we will need delta plus 1 colours

(Refer Slide Time: 25:52)



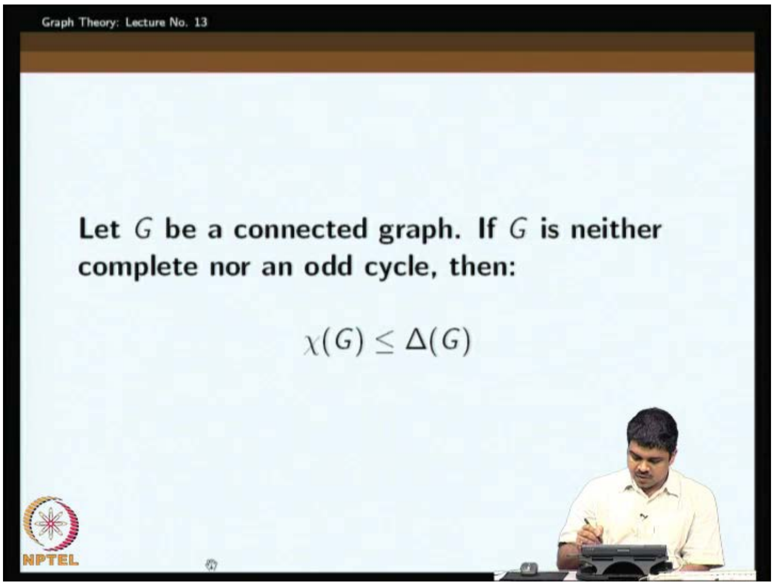
Some other situations of course, you can think of some situations, where there is a graph with several connected components and one complete graph - one K n in it. In that case, for instance, if this happens to be the component with the largest chromatic number among its components. So, we can independently colour these things and reuse the colour among these things.

But suppose, here is where I take the maximum number of colours to colour, then it will happen that here maximum degree also is here. Suppose, then it may so In fact, I want to say that if the maximum degree comes from this component, then that is delta equal to n minus one. Then here, because of this clique here, we have to give delta plus 1 colour.

So this as some Similarly, one of the components can be an odd cycle. Let us forget this kind of situation. Let us say that we are only interested in connected graph because if there are several connected components, each connected component can be coloured separately without worrying about the other part. Therefore, we will say that the odd cycles and complete graphs are examples, where you need delta plus 1 colours, when we consider only connected components.

Now, we are going to study an interesting theorem. where This theorem states that the number of colours required is in fact, at most delta for all other cases, that is interesting. So, the only connected graph which requires delta plus 1 colour is the complete graph and the odd cycles.

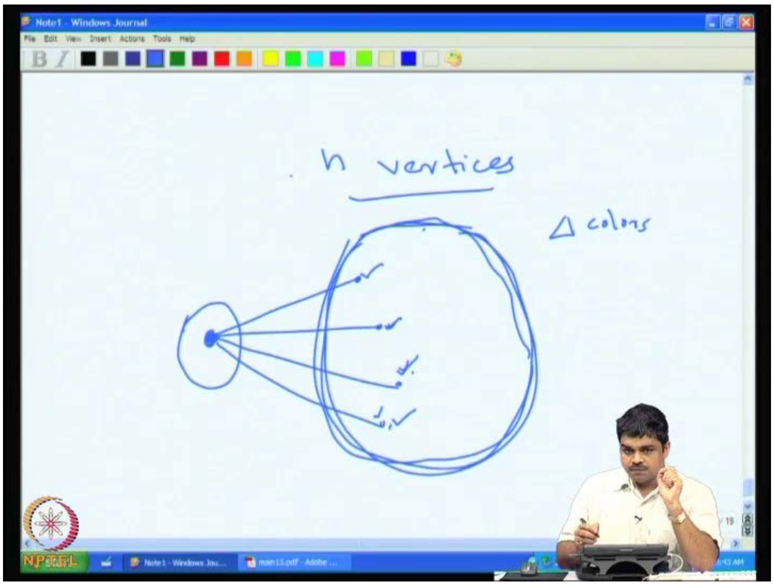
(Refer Slide Time: 28:16)



So, this is the statement, we are going to look at next. Let G be a connected graph and if G is neither complete nor an odd cycle, then khi of G less than or equal to delta of G. So, we will look at the proof of this thing. This theorem is called Brooks theorem and we will study that.

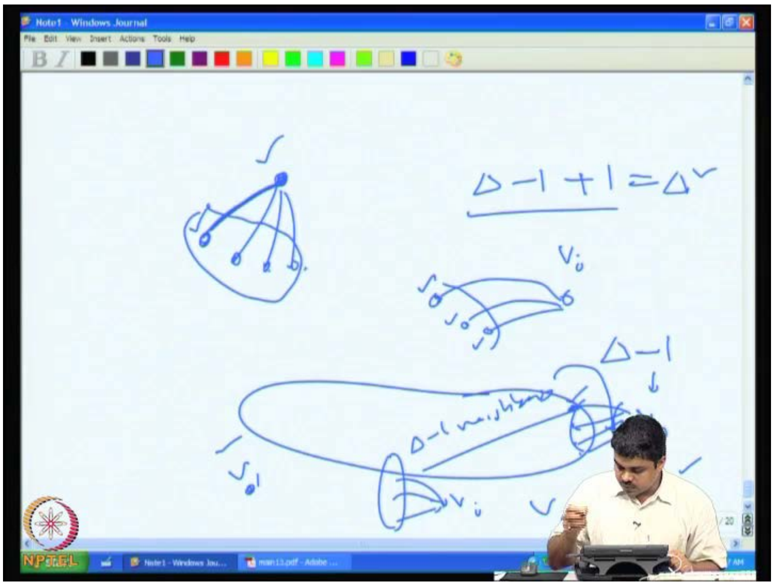
Now to come to the proof of Brooks' theorem, let us do an induction. We will assume that for all small graphs, the statement is true. So, when I say small graphs, for 1 vertex graph, it is definitely true; for 2 vertex graphs, you can always check; to 3 vertex graph also you can check. Let us assume that for smaller graphs it is true; smaller graph means smaller number of vertices, the theorem is true. Now, we will assume that we are considering a graph on n vertices.

(Refer Slide Time: 29:28)



Now, we consider some vertex and now you look at the graph after removing that. So, it has its neighbours; these are its neighbours. So, of course, this graph having only n vertices can be coloured with delta colours. Now, one thing we should first notice is that the degree of this vertex is delta because if it is not delta, suppose, it has only delta minus 1 neighbours. How many colours, it will see here? This is already coloured by induction hypothesis; it can be coloured using delta colours; delta colours or delta plus 1 colours depending on whether it is complete graph or odd cycle this may be. Whatever it is, as far as the original graph is concerned, if this was not a delta degree vertex, it will see only delta minus 1 colours here and so, it can be coloured using so many delta colours.

(Refer Slide Time: 31:40)



We can in fact, argue that any graph which violates this statement, that means, which requires delta plus 1 colours should have to be a regular graph on with delta vertices. Why is it so? Because the earlier strategy we considered, we were in fact finding a certain kind of ordering. We just found out a vertex and then we found out another vertex and we gave a colour to these things and we found out another vertex and then we looked at the neighbours of this and decided to colour this based on whatever colours were already given to the neighbours.

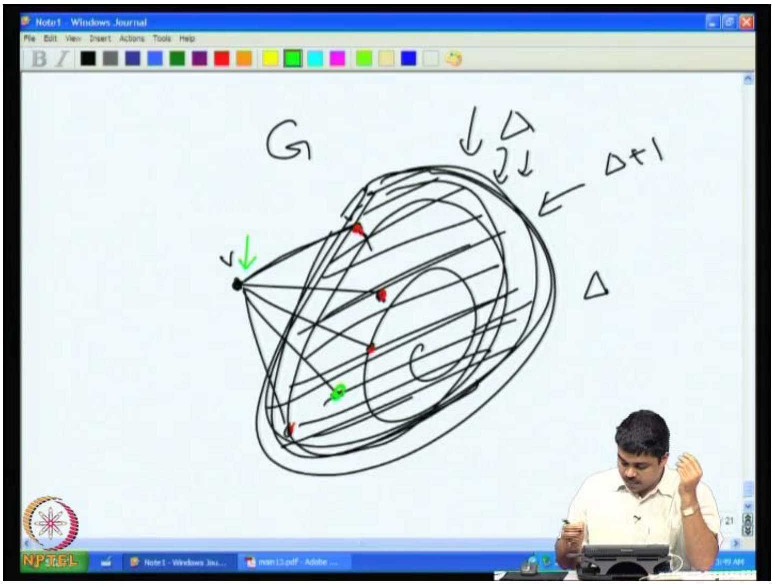
So, when we do this procedure, we notice that all the time when I pick up a new vertex, I will be seeing only at most delta neighbours. Therefore, the delta plus one the colour is available for this vertex, but then if I was doing it carefully, what I would do is I would try to order the vertices in such a way that every time I select a new vertex, I will pick up the vertices in that order and in such a way that the number of coloured vertices, which means so already considered vertices that neighbours it has already considered neighbours it has, is minimized.

So, of course, I will be trying to pick up an ordering in such a way that every time I consider say vertex V i, the i th vertex, the number of neighbours of V i which is a number lesser than i is as small as possible. For instance, if there was a vertex of degree delta minus 1 in the graph, I can always keep that vertex as the nth vertex because anyway, it is going to see only delta minus 1 neighbours in its neighbourhood. Therefore, its neighbours with lower number will be at most delta minus 1.

Now, once you remove this vertex, there will be at least one vertex touching it. Therefore, its degree will reduce. So, we can pick up that vertex because it is another delta minus 1 degree vertex in the induced sub graph after removing V n; that we can consider as V n minus 1. and after that So, I can remove that vertex. Now, I can get a vertex with a lower degree. So, of course So, that degree is delta minus 1, I can pick up. So, everywhere, I will be seeing only delta minus, you can see in this order, the reverse order is this thing. V n, V n minus 1, V n minus two - I can write down the things.

Now, if you start eliminating vertices starting from V 1 onwards, it is very clear that every vertex V i will be seeing only at most delta minus 1 neighbours in the neighbourhood of it. Delta minus 1 neighbours in the neighbourhood of V i. Therefore, we need only delta minus 1 plus 1 colour. It is equal to delta colour. So, this was possible because we assumed that they was a vertex of degree delta minus 1 or less in the graph; that is why we could do that.

(Refer Slide Time: 36:39)

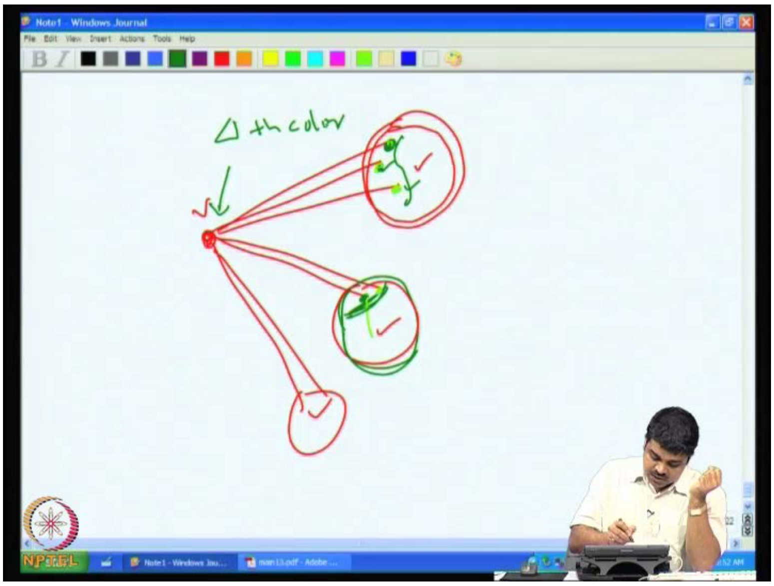


Now, coming back to the proof of Brooks theorem, we can assume that if over a graph G needs delta colours somehow. so by contradict For contradiction, it seems that the n node graph G requires delta colours. In that case, it will be a regular graph; otherwise, it will need delta plus delta colours because if you need delta plus 1 colours, it has to be a regular graph. Now, we removed a vertex and so, this vertex V has delta neighbours here and of course, there is a colouring of this here.

Now, is it possible that this is a complete graph? Here, this is a complete graph. Now, if this is a complete graph, now you can see that the degree of all these vertices here will have same degree in the complete graph. Now, we have delta plus 1 colouring for this Now, you can say this also has essentially because this vertex is connected to this, this degree will have to go one more. So, essentially, it is the delta colouring for this thing. This is the delta colouring for The original graph will have degree delta and the maximum degree of this complete graph will be one less. Otherwise there will not be So, this itself will be a complete graph and we do not have to prove anything.

So, it will so happen that So, we can assume that this sub graph has a delta colouring. Now, we look at the colours on this and if the colours are all distinct, then only we have problem. because if any colour is repeating, then There are only delta neighbours. If one colour is missing, only delta at most delta minus 1 colours I used on the neighbours. So, we can use the next colour delta th colour for V.

(Refer Slide Time: 38:12)

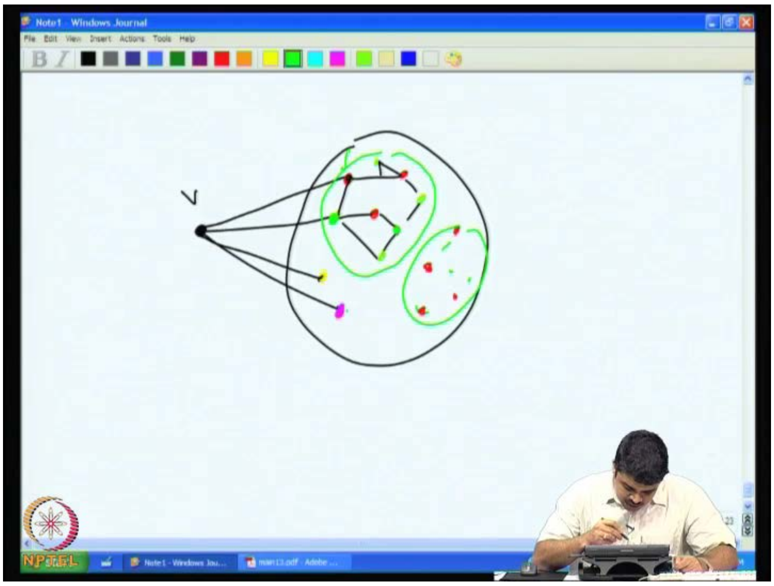


Therefore, we can use the delta th colour for V. It so happens. See one thing we have to consider, we have to carefully to consider. Is it possible that this is somehow disconnected graph? For instance, when I remove this, we are talking about connected graph; when I remove vertex V, its neighbours are going to be disconnected components; is it possible? It is possible that vertex V is connecting these parts together.

In that case, what we can do is these two are disconnected parts. so this colours If at all there is a colour here - green colour here, so I can try to change the colour on this thing with a green colour. For instance, the only problem as I mentioned is that all the colours are different. I can try to get a common colour here, between this and this because within this component, I can exchange colours. So, rename the colours; I can make the green colour red and red colour green So that because these are totally different colours from this.

These colours are different; these colours are different again. Whatever colour comes here, I can rename with the colour, which is available here, which will make a repetition of colours on the neighbourhood and therefore, it will so happen that there is a repetition of colours and since there are only delta neighbours, we are using only less than delta colours, we can use the delta th colour for this vertex.

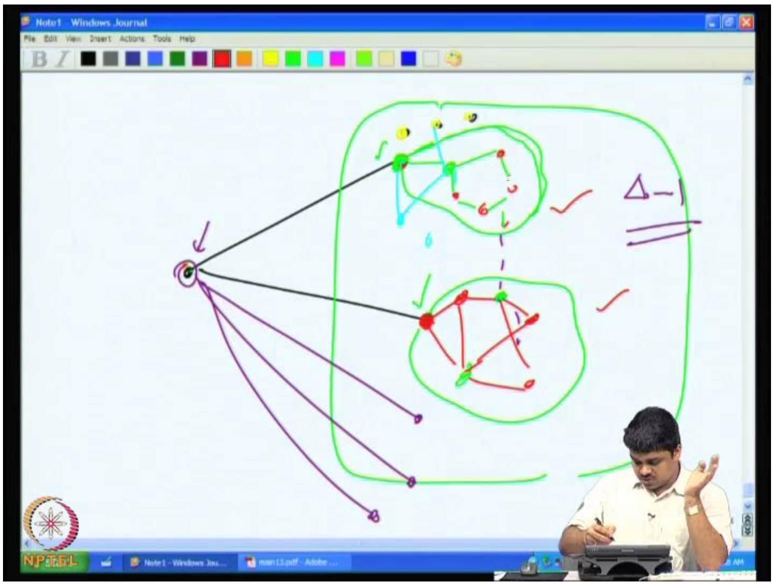
(Refer Slide Time: 40:22)



Now the next point so What I have argued now is, if I consider any vertex V, essentially this will be one component and there will not be several components and all the neighbours are going to this component and again the colours on the neighbours are different. So, I can just use some different colours. to show that different colours So, this is violet, yellow, green and red. I can show like this.

Now, what I can do is to consider say, here I have green and red colour. So, I can try to study the components involving green and red colours alone. There are many other colours - yellow, violet, all these things are there, but then I will look at the sub graph of vertices induced on green and red colours alone. See that will form some connected components. For instance, it may be like this. Of course, it would not be like this. Essentially, I am trying to locate the green and red components. So, that will form some connected components.

(Refer Slide Time: 42:12)



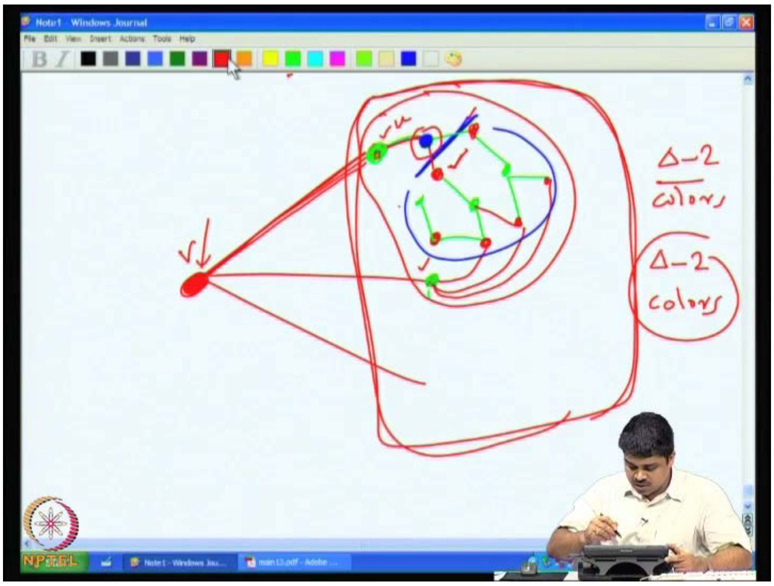
Now, I can look at those green and red connected components and then ask this vertex and this vertex; I will draw it like this. so this is be This is its red neighbour and this is its green neighbor. Now, I am only interested in the green and red coloured vertices of the remaining graph by induction. We could colour these remaining graph and now, there will be a connected component of green and red here, which involves here. There will be a connected component of green and red, which involves here. There will be red vertices, green vertices. yeah This will be just red and green. All these, some connections will be there between them.

Similarly, there will be green and red here also, is it possible that these two connected components are different or will they be same? Though at first look, at the look like they can be different or same whatever, but then suppose, they are different; if they are different then what we can do is in this connected component, I will rename the red and green. For instance, I can make all red vertices green here and all green vertices red. It will not affect anything; the colouring will be still valid. Why is it so? Because if you take any other outside vertices, they may all have some different colours only.

So they whether it seeing neighbour For instance, whether it is yellow or violet or something, whether in the neighbourhood, they see green or red does not matter to them. It has to be different, but then if I change the colour of these red to green, the only vertex for which it matters is neighbours, which has already got a green colour, but then that I am making it to red. So, I am consistently doing it in the entire component. Therefore, there would not be a problem. It will be still a proper colouring. So, if this component is disjoint from this component, there is no path starting from here to here. Then, here I can keep the initial colouring and here, I can convert all the red to green and green to red, which will essentially make a green vertex here because red will become green now. What is this vertex? See two green vertices on its neighbourhood. If it sees two green vertices on its neighbourhood, its meaning is that

So, it is the maximum The number of colours, it sees on its neighbourhood is at most delta minus 1 or less - at most delta minus 1. So, we have one more colour, we can use that colour to this and from what, we can conclude is we cannot do these things because of some reason. What is the reason? The reason is when you try to exchange the green colour to red and red colour to green, that process should go to the entire component. So, here everywhere it will get exchanged and this will become, this green will become red and while this red becomes green and this green becomes red. So, there would not be anything good because there was a green, red, green before. It will become a green, red; so, there would not be any difference. It will so happen that the number of colours, the vertex sees on its neighbourhood remains the same. The entire set of delta colours are seen on that.

(Refer Slide Time: 46:17)



So, we can assume that this should happen. Otherwise, we will solve the problem here itself because it will be a delta colouring. So, that assumption that it could not be coloured using that delta colours will be wrong. So, what we can say is that these two components are same. In other words, if I take the vertex V and look at its red neighbour and green neighbour and the connected component between this is green neighbour, the connected component of red and green vertices between them, it should be that

So, this should be the situation. That means your red and green connect should come together; both these neighbours should be contained in the same connected component of red and green components. Now, we are interested in how this connected component will look like. Now, will it be this complicated, some complicated structure like this. So, if you look at this carefully, is it possible to have two green neighbours, as I drew here, is it possible to have two green neighbours to this red vertex.

So, if it has two green neighbours, how many as far as this vertex is concerned, how many neighbours in this graph, it has because its maximum degree is delta only. Now, one edge is towards this, outside vertex, this V. So, this u can only have at most delta minus one neighbours here and if two of them has same colour, only delta minus 2 colours will be used in its neighbourhood, only delta minus one neighbour are there in the sub graph.

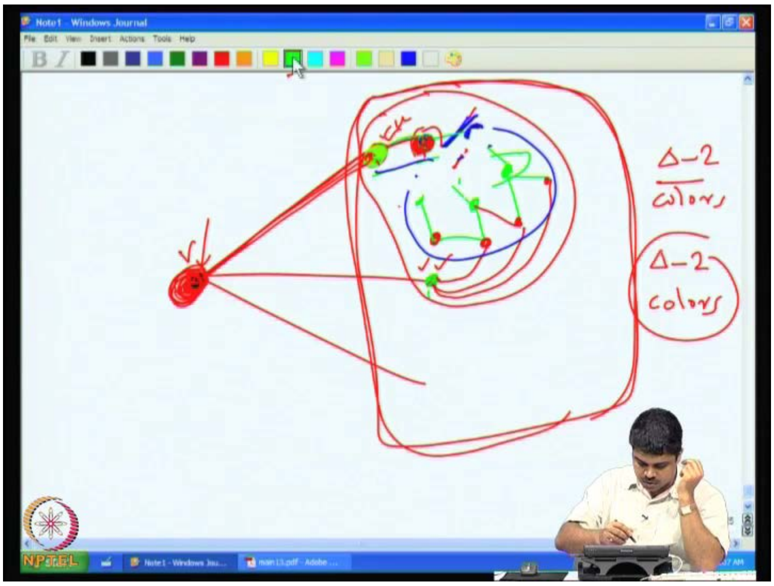
Now, out of that two of them are using the same colour. That means, only delta minus 2 colours are used. We have two extra colours, one of which is already there - red colour one this. We have one more; so, we can use it; may be it is some blue colour. I could have re-used that colour for this thing. I will re-use the blue colour on this thing and it will help me to use that blue colour for this thing and then red colour will be released because red colour will not be there in any of the other things. We have not done anything here.

So, therefore, the red colour can be given to this vertex is and it. in other words are release the colour from this vertex and given to this vertex. So, we can do it that way. so that is what we can do. So what I conclude is you cannot have this u vertex u cannot have more than one green coloured neighbour of it. In other words this is I can clearly say that this is not correct, this picture is not correct. So, this edge is not there; this cannot be there. So, it can only have one green neighbour and this connected component.

Now, I will concentrate on this green neighbour and ask how many red neighbours it can have. So, look, this is a red neighbor; this is a red neighbour and suppose, it has two other red neighbours here. So, this is already a red neighbour and these are two more red neighbours. So, out of the delta neighbours this has, 3 of them are red, which means only delta minus 2 colours are used. Total - delta neighbours and out of that, 3 are using same colour. So, delta 2 colours are used around in its neighbourhood.

So, that means two colours are free, but out of the two colours, one is already there; the one is the green colour because it is already there here. So, there should be one more colour. Let us say this is blue. Then we could have given blue here. So, if it is blue, then see there is no problem. This gets disconnected from the rest of the red, green thing. So, I can safely change this to green. This and this will become green. Now, the red will be free for giving this. So, red can be given to this original thing.

(Refer Slide Time: 52:44)



So therefore, what I can infer is this is red; this is original; this is green. This green cannot have two red neighbours, maximum - one red neighbour; other than this earlier one, it can have only one red neighbor. What I can say is that this is not there, only one red neighbour it has. So, it can only have this. So, this is one red neighbor.

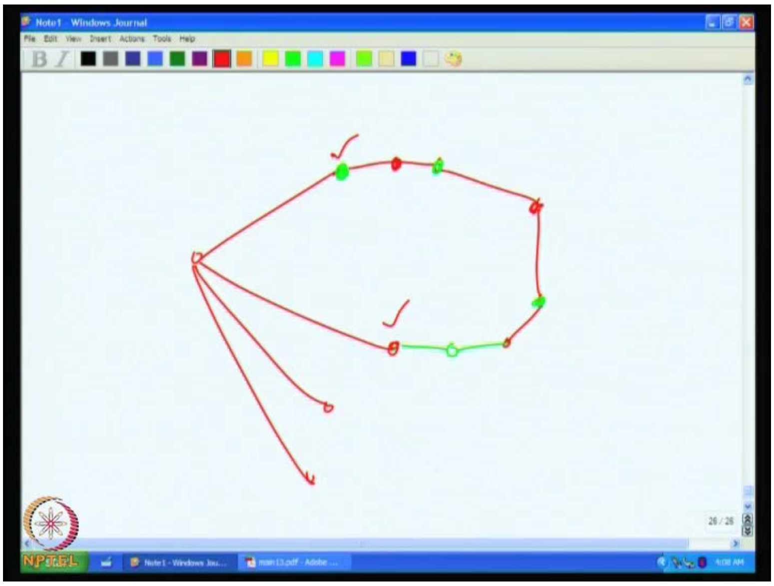
Now, the next red neighbour. So, this is red; this is green; this is red. It is coming in a path and then here if you see this red, how can how many neighbours can it have other than this green? Can it have Definitely, it cannot have another red neighbor. It can have more than one green neighbours. There is one. How is it possible to have one more green neighbour?

So it is because if you can have one more green Suppose, you have two green neighbours. Then the same thing can be told about this. These are delta neighbours of totally delta neighbours for this red vertex and out of that, 3 are green. That means, only delta minus 2 colours are used. There will be one free colour other than the red. one more free colour

So, delta minus 2 on the neighbourhood, one on itself - delta minus 1 and one more. It can be say, it is blue suppose. Then we could give blue to this, instead of red. That will cut away this portion from this. Then what will I do? I will change the colours exchange the colours of this and this. That means, this will become green and this will become red.

Essentially, now, you do not have to worry because the next is the green neighbor. There is no red or green attached to this. So, we can safely exchange these things. Now, this and this has become green; so, red is free. The free red can be given to this outside vertex. So, that will allow me to colour the entire graph in delta colour. What do I infer from all these things? It means that I cannot have this extra green neighbour here. It is only one green neighbor; that is what it means.

(Refer Slide Time: 55:50)



So, we can conclude like this. That, red green component will be a So, this is V. If we consider the red green neighbor, this is the green neighbour and this is the red neighbor. Then we will end up with a path rather than any complicated structure. So, this will be like this, in fact. This will be green; this will be red and this will be green. So, it will be rather, a path and this will be red like this.

So whatever we did between the arguments that we did for red neighbour and green neighbour can be repeated for any two colours, which means that if you pick up two colours in the neighbourhood, between those two colours, I will always get red path rather at A component of the two colours involving these two vertices has to be always a path, rather than any complicated structure. So, the rest of the proof will be considered in the next class.

Thank you.