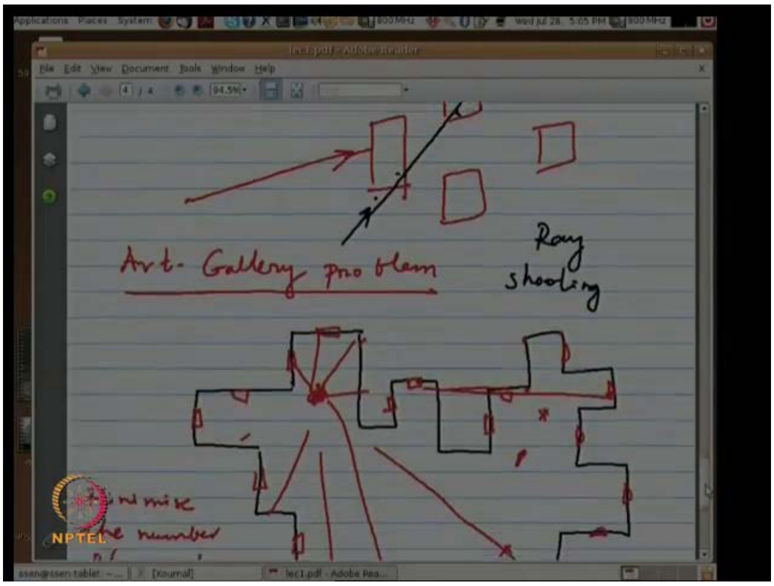
Computational Geometry
Prof. Sandeep Sen
Department of Computer Science and Engineering
Indian Institute of Technology, Delhi

Module No. # 01
Introduction Using Basic Visibility Problems
Lecture No. # 02
Visibility problems: Art Gallery Problems

Welcome to lecture two of the course in computational geometry. I would like to reiterate the couple of things that got left out. So, for recap I left you off essentially at this point that I was defining something called this art-gallery problem.

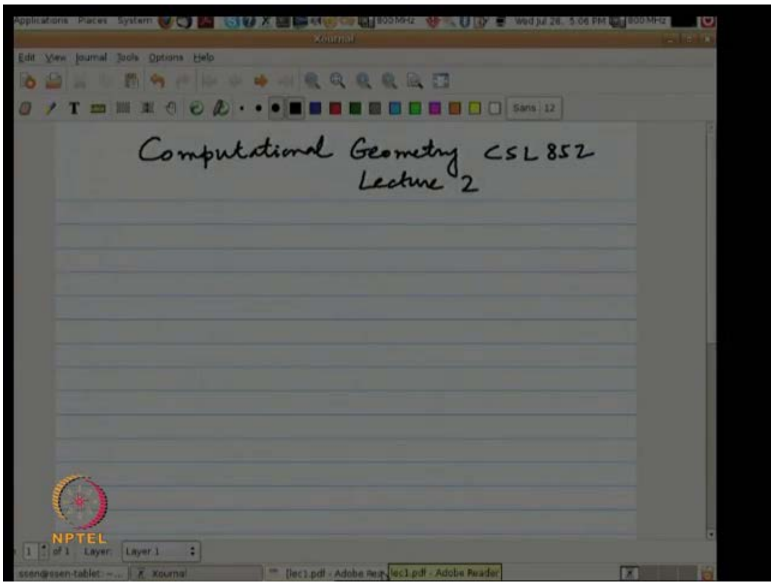
(Refer Slide Time: 00:43)



This is a huge hall and there are rooms in this, halls are only marked out the boundary of the hall, and there are essentially paintings on the wall, which must be looked after and monitored by guards posted. So, that you know they cannot be stolen. And we would like to post as the minimum number of guards, such that every painting must be visible to at least one guard. So, that is the art-gallery problem. So, how does one go about solving this problem? How does one even attack this problem?

So, let me take this simple case. So, this particular hall, you know, has these turned corners and this is not really the shape; the shape is not very even, it has this turns and twist which actually. Makes life order in when we want to decide where you want to keep guards.

(Refer Slide Time: 02:09)



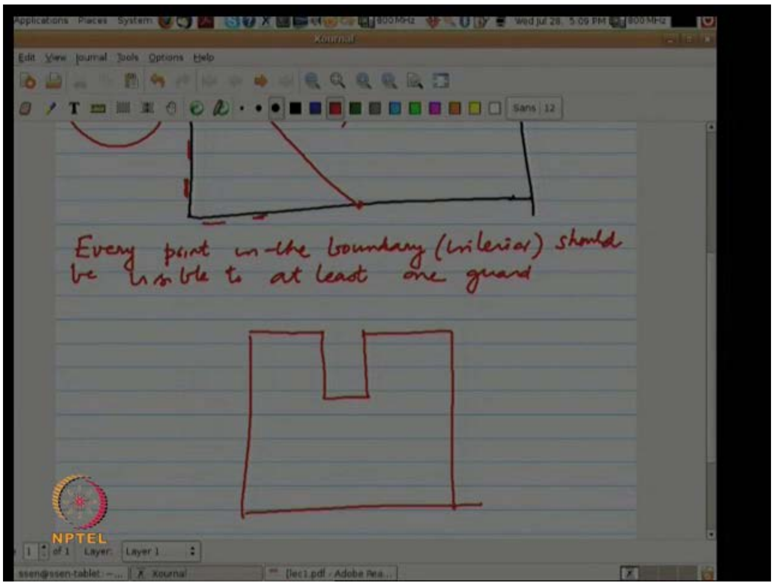
Instead of this if we had a situation where the hall look like this just it is say rectangular hall what the problem be an easier where should I post the a guard or the guards. So, that every painting is visible to at least one guard. So, these are the paintings you can for simplicity again assume that the wall is full of paintings. So, in other words what I am saying, is that every point in the boundary, of course, the interior boundary should be visible to at least one guard.

And the guard has circular region. So, this is like a snail it consist of sea. So, for this particular instance you know what the solution is. I can actually anywhere right. In fact, you know, I can choose this point or this point or this point, it really does not make any difference, because you look at any point on the boundary, choose any point on the boundary, and I can join this point on the boundary with the location of the guard, and this straight line segment is not going to intersect any of the walls.

So, that is what makes the problem in rather simple and straight forward. So, what this shows is that there are certainly situations where the problem is really simple and of course, you need at least one guard. So, this will take care of it.

So, this is about the case where it is a rectangle even if it were a square at which value will be the same or what about if it were a disk, it was some something like this, even then it seems alright.

(Refer Slide Time: 05:11).



So, let us try to make it little bit more complicated. So, how about this, well you are given me an answer, but the first question is that why do you think you know one guard is not sufficient, there is a cusp essentially there is a cusp.

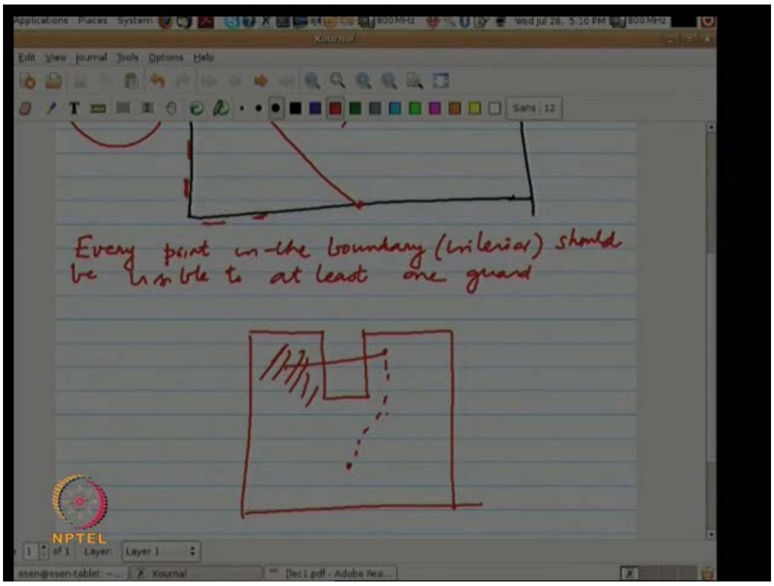
So, I have not formally define convexity if you are familiar with convexity you know is a good answer or at least it is a right direction, but then it appears to us that if I put a guard certainly, if I put a guard here I can argue that this portion is not visible to the guard.

So, this guard certainly cannot see anything here and vice versa, but you know if I move this guard and try to bring him somewhere here perhaps everything is visible. So, it is not just that I am constraint as that I am looking for a solution where I should be able to put the guard anywhere I also have, I can choose the location of the guard.

So, if you want to argue somehow that one guard is not adequate we have to argue that no location there is any location in this hall where everything is visible. So, can you argue about that; alright that is good attempt?

So, what you are saying is that suppose I have a painting here and a painting here and then I want to somehow find out if there is a location from which both this paintings are visible and will be hard pressed to find that.

(Refer Slide Time: 06:25).



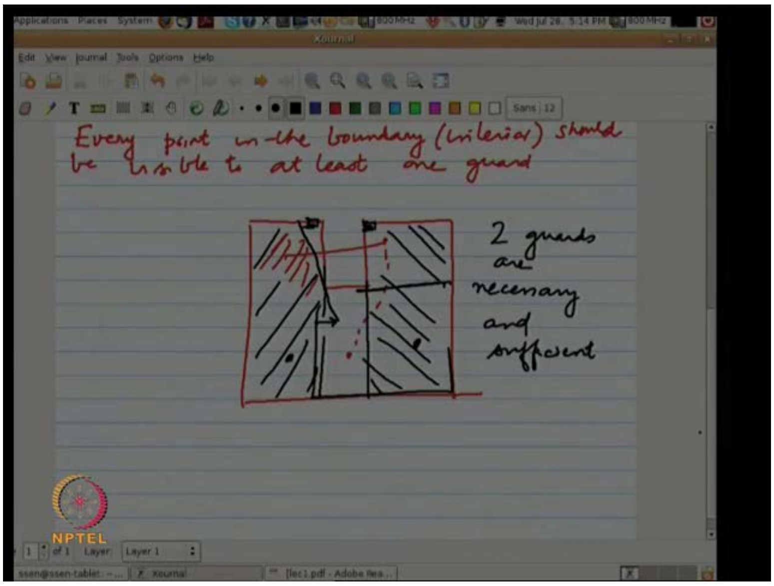
In fact, you can probably argue that for. So, suppose this is the corner. So, somehow since light travels a straight lines. So, the this painting can be visible only if the guard is in this portion you know the movement I move the guard here, this straight line basically will be such that it can never intersect that painting without defecting the walls.

So, that is essentially the idea about how one can argue, that more than one guard may be necessary, the other thing is to somehow argue that whatever do you think two guards are sufficient in this case do you think you ‘liquid require more than two guards we have argued that we need more than one guard, but are two guards sufficient.

So, I guess how you would argue about that. So, certainly if I am constrained to put a guard in this rectangular shaded region I am constrained to have some guard in this shaded region two because again this painting must be or this portion of the wall should be visible to that guard. So, how do I now argue that everything else is visible, well if I put a guard anywhere here, from our previous argument this entire rectangle should be visible from here. So now, you only have to argue about the left shaded portion and the middle. So if I post a guard here, even this guard can actually view everything within this rectangle.

So, if you look at this rectangle everything is visible. So, that is how you can argue that you know within should be visible to two guards two guards are sufficient So, in this case what we have seeing that two guards are necessary and sufficient it is a fairly strong statement. All right say this is by visual inspection the algorithmic problem is that I am given the description of this hall and the hall can be of any shape and as somehow we are supposed to let us say algorithmically determined you know what is a minimum number of guards that we required and that is a challenging problem.

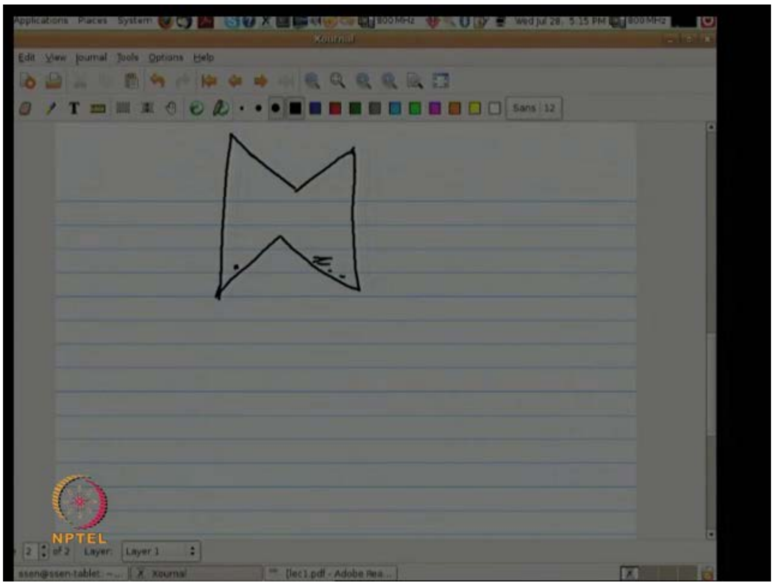
(Refer Slide Time: 09:51).



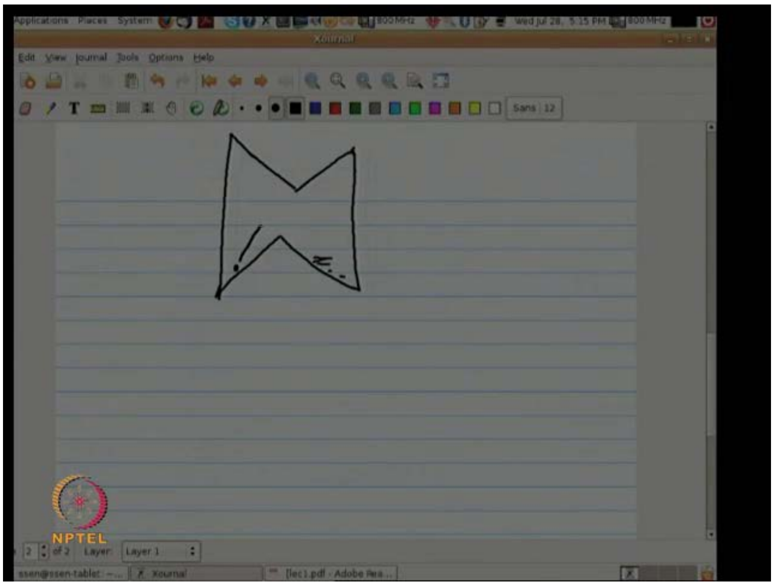
Let us try some more examples. So, we were worried about the something about convex. So, convex let me come to that, I will come to the definition just a little later may be today itself , but even prior to that let us try something again which has a cusp, let us try another room, let me relax again the condition that the hall's walls must be parallel to the it's coordinate axis.

You know let us relax the condition and allow for what is called essentially a simple polygon. So, I could have a situation like this. This is more modern let us say modern architecture hall. So, it need not constrained to build everything in the (( )) fashion. So, this has again that problem that it has cusps. So, essentially if I put some guard at this location you know this region is not going to be visible to this guard, but then how many guards do you require for this figure or this hall.

(Refer Slide Time: 11:03)



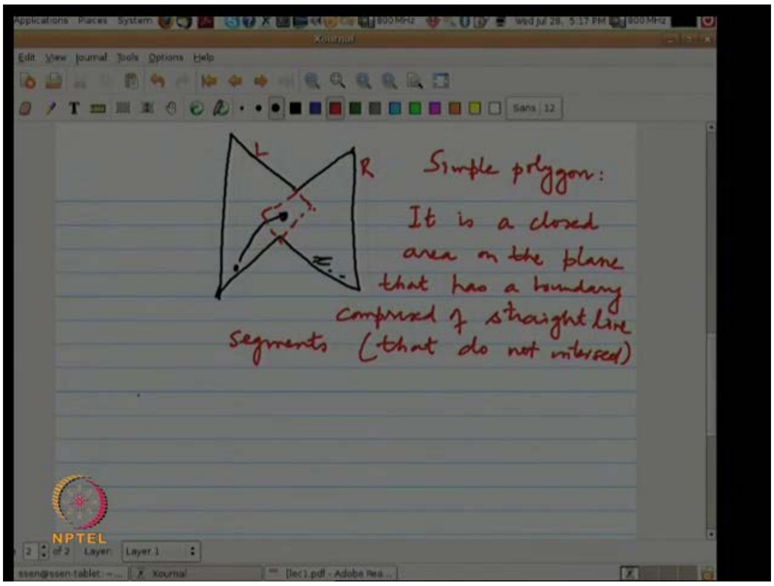
(Refer Slide Time: 11:10)



So, by inspection it appears that if instead of putting this guy in a corner, if I somehow move this guard to come towards the center, seemingly everything should be visible and why is that we can argue that again. So, I can complete this let us say this triangle and complete this triangle. So, this left triangle this is visible from any point in the triangle. So, l is visible similarly you know r is visible from any point in the triangle and this lies in the intersection of the two triangles, so the entire polygon interior of the polygon is visible.

So, just to make sure that we have our definitions consistent. So, let me define the notion of a simple polygon, let's say, closed area on the plane that has a boundary comprised of straight line segments that do not intersect.

(Refer Slide Time: 12:50).

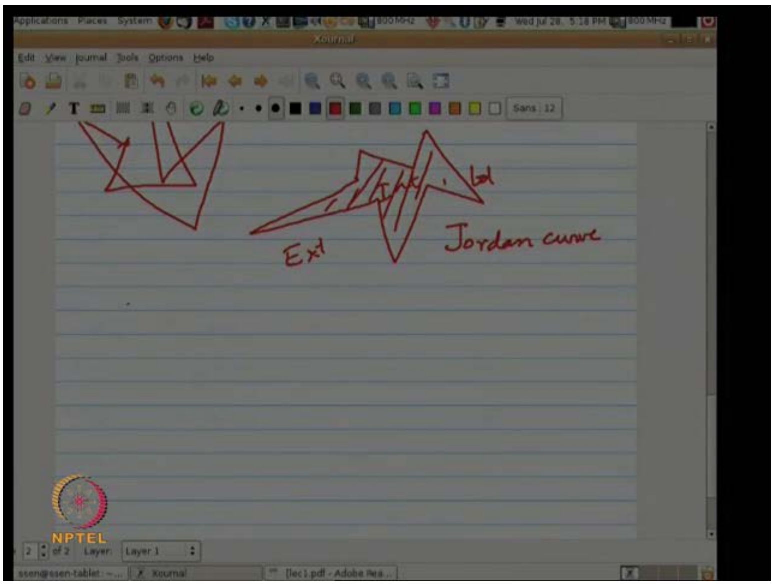


In other words, it is a closed figure. So, I should be able to draw it without lifting my pen. So, in other words I could draw, this is also closed figure except that it is self intersections the boundary intersect where as you know if I have something like this that there is no self intersection. So, this is a legitimate simple polygon. So, it has a well defined interior and it has a boundary sometimes will call b d.

So, this is an interior and this is an exterior. So, simple polygon is also sometimes called in a more general sense sometimes people refer to it as a Jordan curve. So, Jordan curve is a property that it actually partitions the plane into three parts namely the interior the boundary and the exterior.

Jordan curve can actually need not be a straight line segments, but you know any kind of curve. So, this particular simple polygon that we just investigated although it has these cusps, it has these bents which go inside as opposed to something that looks fact like this which is what essentially is a convex figure, even here there is no solution with one guard. So, these kind of polygons although they are not what we call convex, they are called star shaped polygons.

(Refer Slide Time: 14:24)



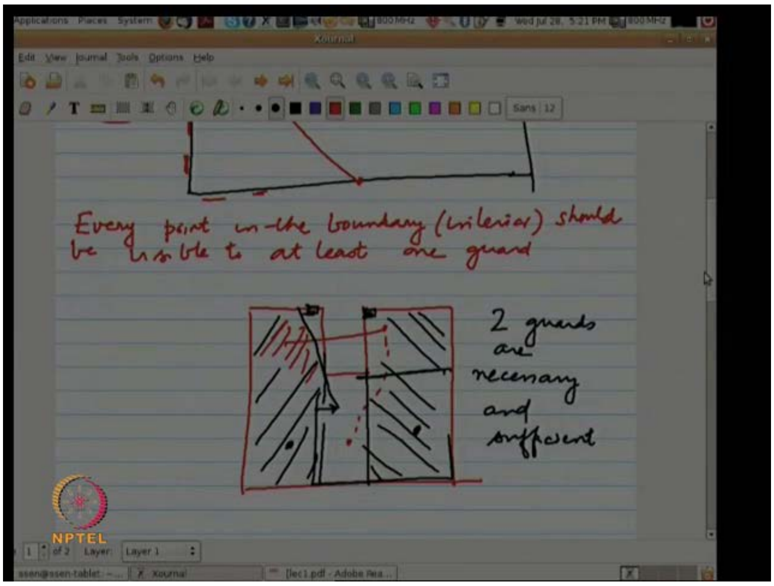
Of course, we have the notation of convex polygon. So, the convex polygon in short is something like this if I can draw. So, take any two points in the interior if the segment joining then does not intersect the boundary. Then it is convex.

So, this is the quantification is that for all pairs of points x one x two or let us say p one p two so on. So, all points p one p two they should be true and therefore, something like some, figure like this if I take these two points and I draw a segment that we intersect the boundary of course, So, there are points which I can join without intersecting the boundary, but if you take a convex polygon like this or even something like this rectangle, take any pair of points I can always join them so that it never intersects boundary.

And what is the smallest simple polygon or convex polygon triangle. So, triangle is a simple polygon and also it also happens to be convex. So for convex polygons we can essentially claim because of this property that one guard is adequate because whichever be the point on the boundary I can always draw or I can place the guard anywhere and I can join the guard with that point that does not intersect the boundary and therefore, it is visible by definition.

Let us again generalize sort of generalize the problem to what we looking at. So, you are only looking at every point in the boundary should be visible from here let us say that not only every point on the boundary every point inside in the interior should be visible.

(Refer Slide Time: 17:00)



Let me. So, what I am seeing that I am extending the problem the scope of the problem where I am saying that we want to actually place the guards in a way such that every point in the interior is visible to at least one of the guards. So, may be not constraint to put the paintings only on the walls, but you know sometimes we have this exhibits on the floor somewhere. So, every point should be visible to at least one of the guards no. So, there are some examples where it is not so.

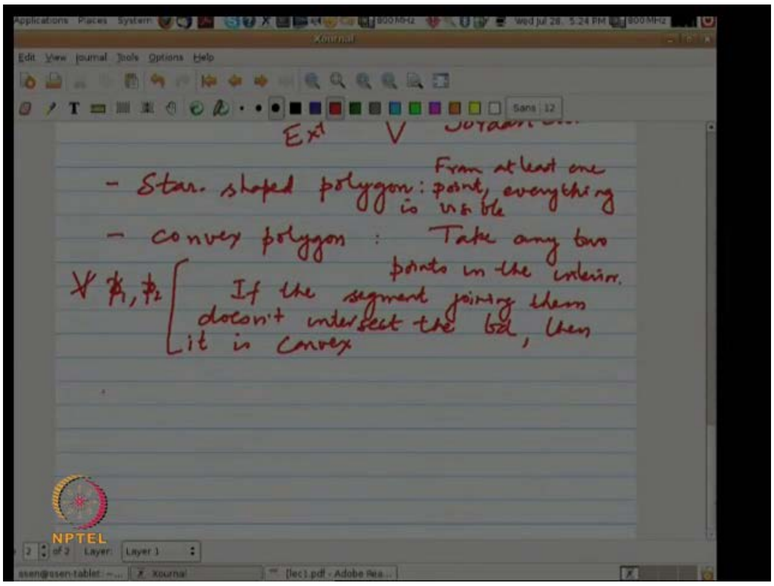
So, one has to work hard to create those examples, but they are actually examples where you can all the points in the boundary is visible, but not all points in the interior is visible. So, I will come to that may be I would not actually even take it up in the source and in this lecture.

So, let us investigate more. So, what about an arbitrary. So, we have seen that for a star shaped polygon a star shaped polygon what is the star shape polygon define the convex polygon I have not defined the star shape polygon.

So, star shape polygon basically has a property that from at least one point everything is visible well everything in the interior is visible. So, the star shaped polygon need not be necessarily convex as we have shown in this example, but there is here it is not just one point you know. This entire region is such that if you place a point in this shaded region, anywhere in the shaded region everything inside the polygon is visible and sometimes this is also known as the kernel of the polygon.

So, the star shaped polygon has a non trivial kernel. So, this is one of the definitions. So, it has a non trivial kernel there may be other alternative definitions equivalent. So, we know about star shaped polygon, we know about convex polygons, in both cases actually one guard suffice we have seen some cases like this one where one guard does not suffice.

(Refer Slide Time: 19:37).



And in general as I said, our goal is to the goal is to given any arbitrary let’s say now we dealing with polygons in an arbitrary simple polygon.

First let us say algorithmically the minimum number of guards, in other words we could have a really crocked polygon something like simple polygons can be actually quite complex as long as it would not intersect, this is the games that used to play as children, you have to draw things without intersecting. So, this is a simple polygon it has a well defined interior is it bad enough you are right. So, now, I can challenge you that I do not think in the next two minutes, anyone can actually tell me what is minimum number of guards required here and of course here. So, that we can make it much worse you may have to struggle for not only five minutes or ten minutes perhaps you know five hours. In fact, the problem gets solved very quickly.

And the answer is that well if the pessimistic answer is that this problem is a problem. So, for people who had done a course in algorithm they should know what this term means that if I give you a simple polygon and arbitrary simple polygon with n vertices there is no known polynomial time algorithm that can find out the minimum number of guards.

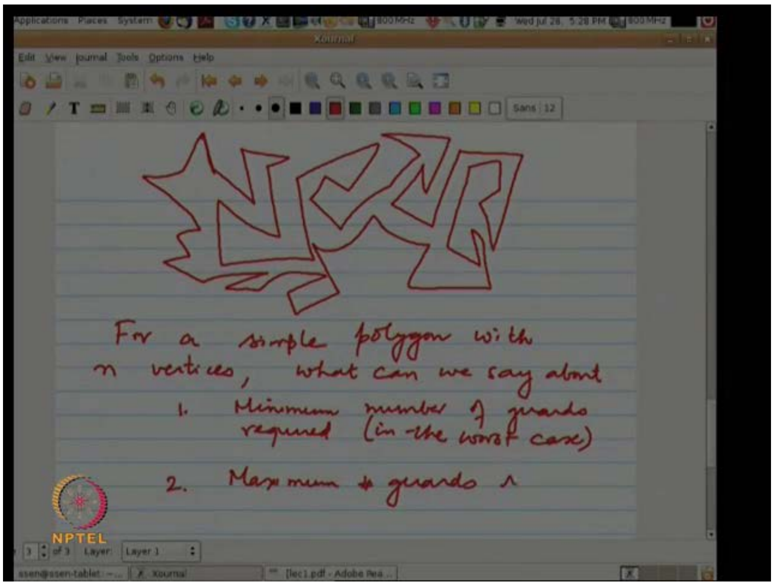
For an arbitrary polygon of course, you know for some special cases we can. So, that is a pessimistic answer. So, what we do at this point, not much, I mean algorithmically not much except that people who are believe in approximations we may still able to do something and I will say a few words about that too, but before that let's try to tackle at least some kind of commentarial aspect of this problem.

So, here is an alternate question which is not quite a algorithmic equation, but one of the commutative equation; for a simple polygon with n vertices, by the way how many edges to the simple polygon have, same n and n vertices n edges and what would be a good representation of a simple polygon, I mean after all if you have to solve some thing algorithmically you have to represent it inside using some kind data structure.

Just and least, but an ordered list we do not want to lose the information about the adjacent. So, that way we should know that you know after this point it is this point in this in point it is not just a set of points because set of points are actually the cornered points are connected in a certain way and that defines that particular simple polygon.

So, it is it is circular list essentially or you know what you are in array what you are is a natural representation for simple polygon. So, for a simple polygon with n vertices what can we say about number of guards required in the worst case.

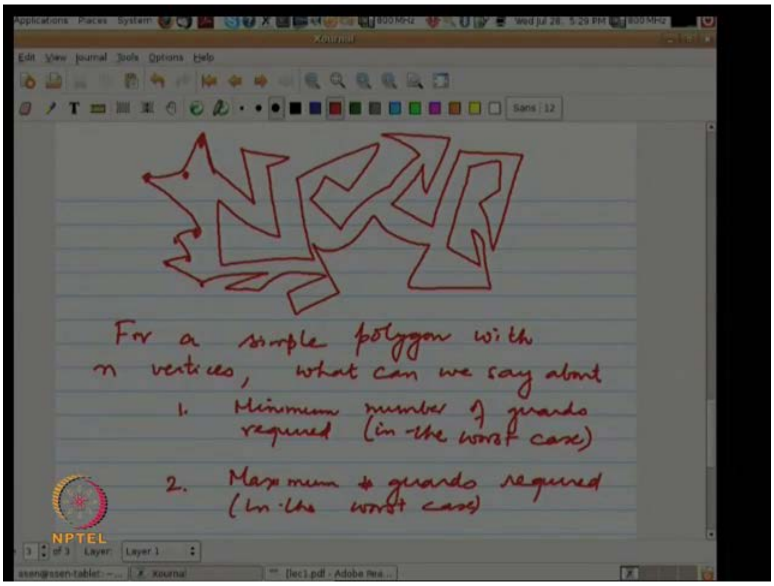
(Refer Slide Time: 23:49).



Maximum number of guards required in the worst case, and it will be great if these two numbers actually match. So, that you can come up with some answer where we can establish that answer to number one is the same as number two otherwise our estimates may not be a good estimate. So, one clear answer, let us look at let us look at number two first.

What is the maximum number of guards required for a simple polygon with vertices? So, why are you saying it should be so, if you have a guard posted at every vertex well the edges the boundary is certainly visible.

(Refer Slide Time: 24:48)

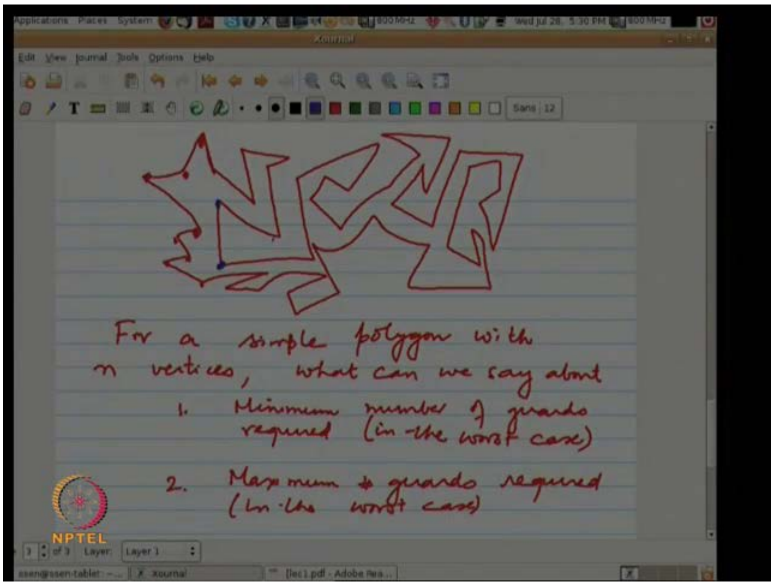


So, can you argue about why the interior should be visible, it is certainly, make sure because here we have, if you have a point in every edge that entire edge should be visible which means the union of the edges is visible.

So, n guards certainly suffice when will when we are talking about boundary visibility, but if we talk about interior visibility this may not be all that obvious you may have to use some kind of geometric reason.

What is that result? You take three consecutive things let us say I take three consecutive things you know.

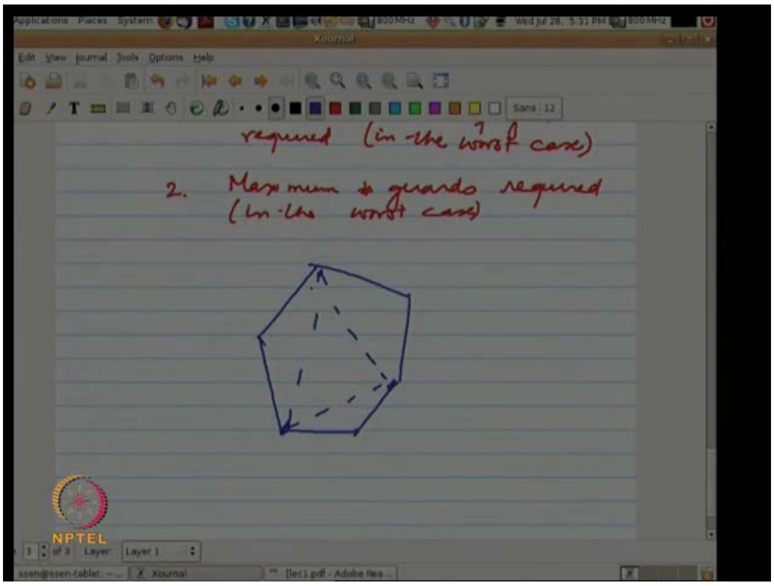
(Refer Slide Time: 25:44).



This triangle is actually even exterior or you know you could have a situation where essentially you do not even have a triangle, but you have hit upon some good idea and intuition that is if you can somehow say that we can partition a simple polygon into triangles then we are done.

Do you agree with that? Because we are the triangle must have at least one external edge or not necessarily well no yes no not necessary; let us try this, sorry just this is not a good example just one more and can have a triangle where no sorry this one not this one, but then it must at least have a corner point and if you place a guards on corner points. So, what I am saying is that it is not true that a triangulation is always has a property that one of the edges is a edge of the polygons.

(Refer Slide Time: 27:26)



So, this triangle does not share any edge with the polygon, but on the hand if you place a guard in any of vertices of course, entire triangle is visible. So, if you can somehow show establish that we can. So, Observation one, if we can triangulate a simple polygon then n guards posted at vertices suffice this should be this is the lesson that we have. There was some question about what was that about maximum.

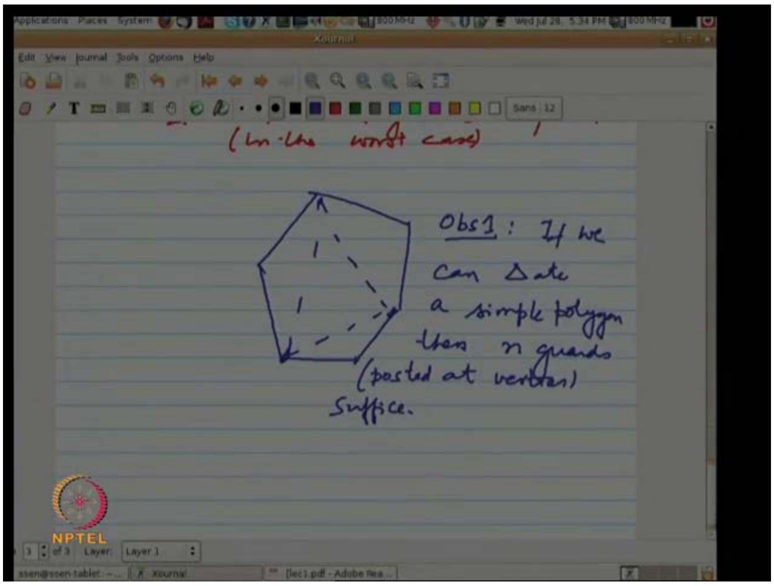
So, whatever is the shape of the polygon we should be able to prove that will no never require? So, this n guard is not dependent on the shape of the polygon whatever be the polygon, if I can triangulate that polygon if you can establish this at a simple polygon can be triangulated then n guards will suffice whatever be the shape of the polygon.

It does not depend on you know star shape convex or whatever. So, that is I mean by worst case, that is why I said when we have a solution to this two and solution with the one no. So, we want a solution two to one which should be matching that basically means the bound should be tight.

So, I should be, if you say that if the answer is n plus one then I should be probably also be able to demonstrate that in some cases n plus 1 is necessary.

So, that is what I meant by you know we should be able to get matching bound for one and two otherwise I can say n square n cube and some arbitrary number clearly that is why we are looking for tight bound such that one and the figure one and two should match that should be our goal.

(Refer Slide Time: 29:36)



So, can we try to kind of justify this thing that we can triangulate any simple polygon, let us first try this. Let us try that, even before we try that, see the challenge is basically it will be dealing with the you know this kind of crooked polygons. So, I should be able to demonstrate the following, that if I could draw for instance some edge connecting two vertices such that the edge completely lies inside the polygon, suppose I could prove that I could always do that, then I have a solution or I have an answer.

I have an inductive answer, why because I know that a triangle is obviously triangulated and any other simple polygon if I can draw, this is called a diagonal. So, if I can draw a diagonal that is an edge joining two points that does not intersect the edge of the simple any edge of the simple polygon, then I can apply this inductively.

It is also a kind of the constructive proof. So, all I need to demonstrate is that I should be able to draw a diagonal of a simple polygon and then we are done because now we are dealing smaller polygons. So, how do you justify that?

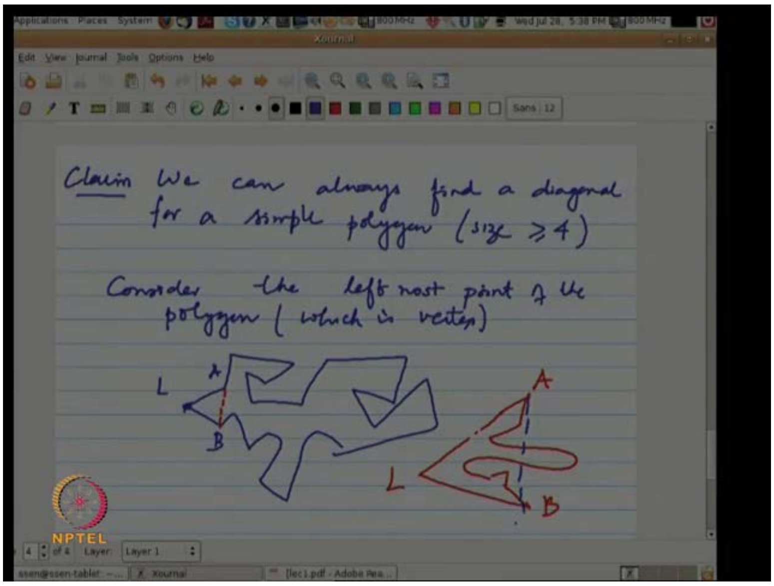
So, for that let us try this idea. So, consider what we are trying to prove, we are trying to prove that we can claim always find a diagonal for a simple polygon size greater than or equal to four. So, once it is not a triangle we should be able to find a diagonal.

So, let us I will do kind of a prove sketch you know. So consider the left most point of the polygon which is a vertex. So, we have a polygon, some arbitrary polygon etcetera something, this is what I mean by left most, let us call it lis, a left most of the polygon. Now consider the two adjoining vertices that are connecting to the l, call them let us say a and call them b, call them a and b respectively. They are both to the right l because this is a left most point.

Now, if I let us draw this diagonal a b. So, in this case great works out; however that may not always be the case, if could draw the diagonal we are done, but we could also have a situation where we have a situation where you know something like this.

So, this is my l, , this is my b or let us have a something even more general. Sorry this is like polygon. When I try to draw this diagonal, it is not this edge is not a diagonal, there intersects a polygon. So, if it intersects, then you pretend the you do the folly. You take this edge a b and translate it to the left towards l. So, you move this you, move this at some point.

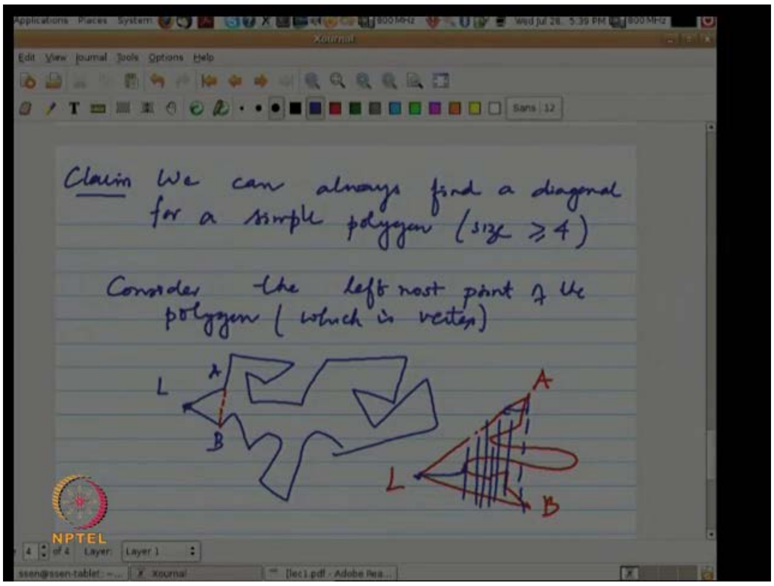
(Refer Slide Time: 34:06).



You are beyond which you know if you move you know it will completely lie in the interior of the polygon. So, clearly you know if it is very close to l it has to be completely interior of the polygon. So now, you see that there must be some last point that it must have touched and that has to be vertex.

So, if this is a vertex, then this must be completely inside the polygon. So, this must be a diagonal. So, this tells us that for a simple polygon, we can always find a diagonal and hence we can always triangulate a simple polygon.

(Refer Slide Time: 35:01)



Yes, but you can always a … So, the question is the following that stress that I have. So, let me just repeated. So, here I have taken a there is this one, but you know suppose I have l one and l two, that is what we are worried about. What I am going to do in this situation this is, you know this a l one l two is parallel to the y axis.

So, there is no you unique left most point, but then what I can do is I can always do random I can fix a left most point. I can always find a unique left most point by orienting the axis and that does not change anything about the triangulation.

So, many of this you know in many situations will come across this cases where I just make a reference to that you know you can always orient the coordinate axis and in most problems, you know the solutions will not depend on the orientation of the axis. In some cases they will, but for instance here you know does it really depend on the orientation of the axis.

So now, we know that we can triangulate a simple polygon. So, that is good news for us. So, which means that our that n guards suffice is certainly correct. Question is can we do anything better can we get a bound better than n square root of n. You know half n something like that. So, any guesses? Can it be less than do you think it can be less than n, something that you know is quite far from today lecture.(( )) no. So, when I talk about yeah. So, what you are saying that instead of triangles if I can partition the polygon into other kinds of convex shapes maybe I can get a better point. That's a good idea that something that we can follow, but let me work out at different proof which you know is a very clever and beautiful proof. So, that is the reason you now I will discuss it. So, something exactly things do appear redundant even in the case of triangulation. Why should I have a point, at every vertex of every triangle?

Something because they have kind of triangles in a share some common edges something like that. So, that is an intrusion why this bound that we have of n you should be able to improve. So, that is precisely what I will do you know. So, this is what I will use. I will make a reference to what I called coloring of the vertices of a triangle. So, coloring the vertices of the triangles of a given triangulation of the simple polygon. So, you consider any simple polygon and I consider a triangulation of that.

This is possible triangle a simple polygon. So, you want to color the vertices of the triangles. So, let us see this is triangle t 1, this is t 2, this is t 3, t 4, t 5, t 6, t 7, and t 8. So, they are these are the eight triangles and I want to color the vertices of the triangles is a such a way that no two vertices sharing an edge should have the same color.

That's the usual notion of vertex coloring. So, in way such that no two end points of a triangle have the same color, now what do you think the minimum number of that on what is the minimum colors that we require for this purpose?

So, even if you cannot guess let us assume it is some number let us say suppose it is c. So, suppose I require c colors to color this is a number of colors' that we require to achieve this coloring.

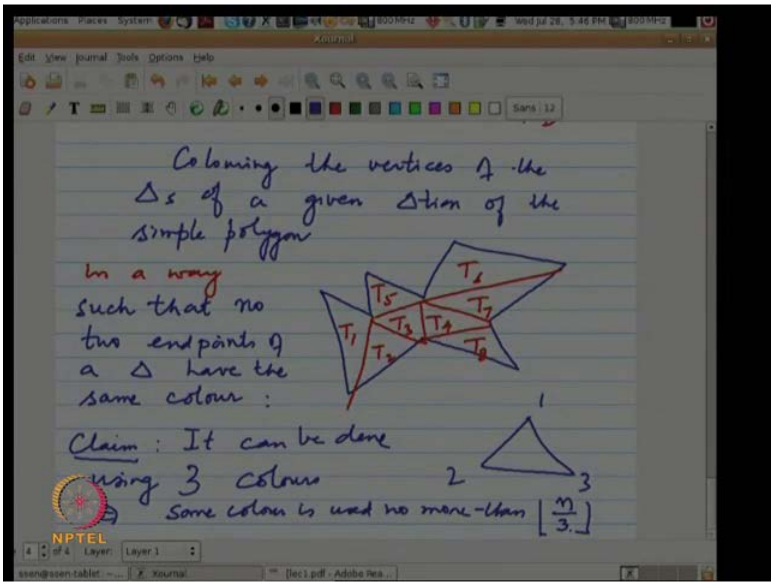
So, if this number is c then what can we do. So, we know that if in any given triangle it must use at least three colors because any triangle by this definition you want to color a single triangle the simple case. So, I need at least three colors' you know one 2 three because all these edges the two end points of the edges should have distinct colors.

Right now I am saying c may, it is not a good idea, what we see let me work with actually. So, let me actually give you the answer, let me go back to give you the answer.

So, claim can be done using how many, three or four I do not know how you got that, but three happens to be the right answer and you see the three is the minimum to even color a single triangle.

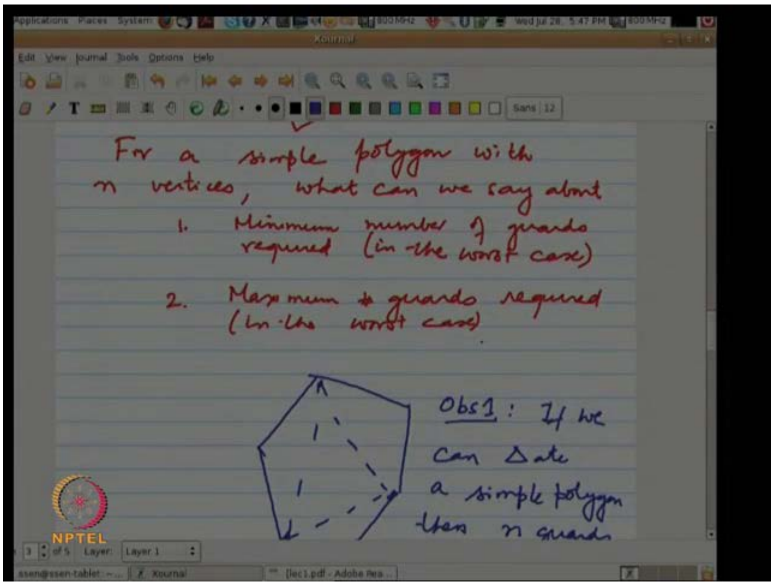
Now, if you can color this triangulation whatever may be the triangulation of simple polygon using three colors' it implies that some colour is used no more than, what wait n over three actually. So, even better n over three better n over three the flour of that some colour is not going to be used for more than n over three times now I claim that if I place a guard. So, whatever that colour is, suppose I am using red, blue, green suppose it happens to be the red colour is used no more than n over three times.

(Refer Slide Time: 42:32)



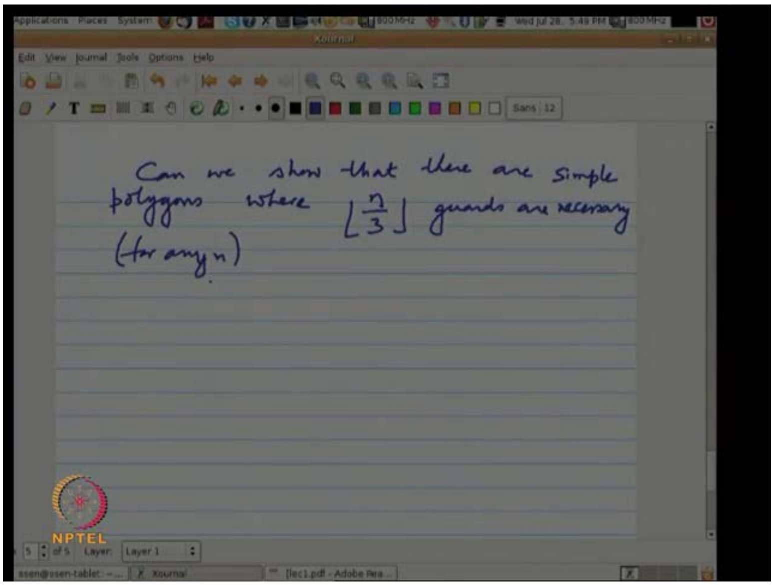
I claim that if I put guards in exactly at those corners the entire polygon is visible, see every triangle must have three colors' and therefore, it must have necessarily one of the vertex with a red colour and therefore, and the entire triangle is visible from that red colour. So, if every triangle has A guard posted at the vertex that it has red color then the entire polygon is visible the interior polygon is visible this proves that n over. In fact, let me go back. So, when I pose the problem was

(Refer Slide Time: 43:26).



So, the answer to the second question we have the answer to the second question maximum number of guards required in the worst case is n over three. It will be very nice if we can somehow show that there actually polygons. So, the next thing that one would like to establish is that. So, can we show that? There are simple polygons where n over three guards ,then we have a good bond because we have a tight bond that there are actually polygons that we required and this for today I am leaving it is an exercise for you . So, you should be able to do it for any n this one.

(Refer Slide Time: 44:46)



So, other thing whether you want to do prove is this claim. So, for today I am leaving with the exercise that you must prove somehow, try to prove that any triangulation of a simple polygon those triangles can be colored using three colors that I have not done the proof here, think about it how do you want to prove it is not hard and this one the other part can show that there are actually simple polygons which will require n over three guards you cannot do it with less than n over three guards.

So, they are two things I am leaving an exercise today. So, this one you only have to show you know somehow they must be actually show me a figure where n over three guards will be necessary in the shape should be such that about that these many guards should be necessary and the proof of this theorem is the art-gallery this is called the art gallery theorem actually the 1 and 2. Together that is known as the art gallery theorem and it was first and the proof was given by Chuntal, but his proof was not this proof was this is much simpler proof such was Chuntal proof was somewhat harder, this proof is really skewed beautiful proof. So, I could do it in its ten minutes. So, this is known as the art-gallery theorem.

So, let me again go back to the algebraic equations, we have answered some kind of a comment real question that there are polygons for which n over three guards are necessary and sufficient, that is what we have proved that is the art-gallery theorem there are for every n there are polygons for which we require n over three guards and n over three guards are sufficient t, but the algorithmic question as I said you know is cannot be solved. Because it is known to be or it has proved to be n p hard problem. So, what can you do? So, for that you know I will give you some idea and you know this kind of you know I you know techniques you know we will also you know later on and it is fairly intuitive thing. So, given a simple polygon you know any arbitrary simple polygon algorithmically how do even find out where to place the guards, how do you find out. So, what I am saying is that even brute force how do you solve that problem, how do you solve the problem in the brute force, if I want the minimum number of guards again the problem is here that it is not a discrete problem. So, it if you. So, when whenever you talk about n p hard problems in the contest of graphs,

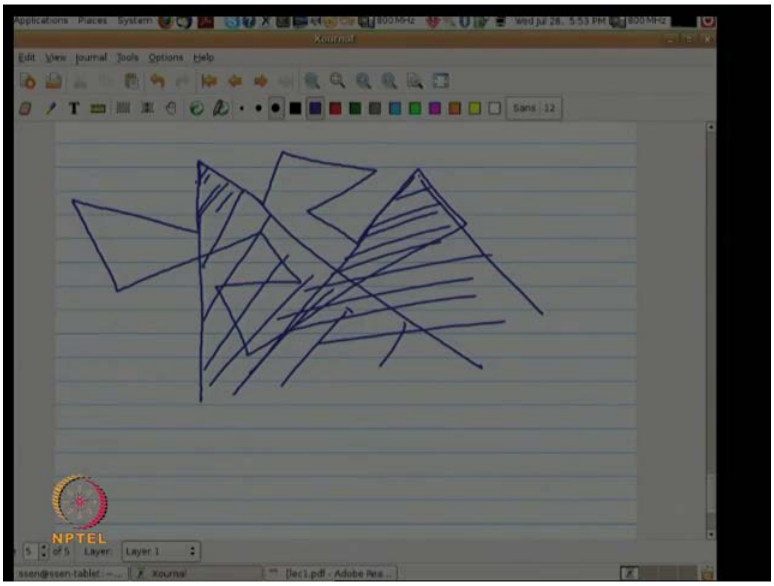
some permutations some things like that those are discrete problems that well I do not have that polynomial time algorithm, but at least I have some n factorial algorithm or 2 to the power algorithm some of that satisfiability has to the power n solution because I can enumerate all the solutions. So, here again you know because we are dealing with a continuous phase where do we actually place this guards to five and the minimum . So, you are on the right track. So, here is I guess let me translate. So, I have some arbitrary simple polygon, one thing I can certainly do is I can think about if I with this region has to be visible I need a guard.

You know this cone somewhere similarly for this one if I place a guard I need to place a guard for this point is visible in this cone. So, I place something in this intersection then these two cones to be visible.

No, but we cannot anything in that double that because horizontal I mean we cannot guard anywhere in that cone. No, you also have to respect the boundary of the polygon. You have to respect the boundary of the polygon also, but you can define these cones along with the boundary of the polygons in a way such that you can argue that, do it for you can find out all the you can, essentially enumerate all the feasible solutions how, do you enumerate all the feasible solutions, one way is to take all the possible cones look at the intersection of all the cones now.

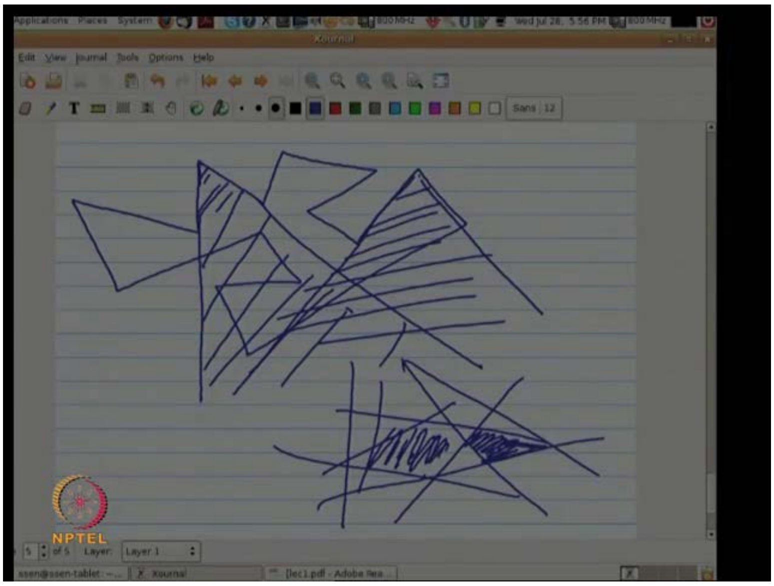
When you look at the intersection of the cones. So, it would look let us say something like this it is a partition of the plane. So, it will look something like this and the claim is that any phase if this partition the visibility information is fixed that whatever is visible from this point is also visible from this point, because you are not crossing any cone boundaries, if you do not cross any cone boundaries the visibility information. So, there are in some sense some kind of equivalence regions now, movement you can you reduce a problem to this then it becomes a covering problem. So, I need to find out. So, also for I can actually enumerate for every phase which parts of the polygon is visible then I want to find a minimum cover of that. So, this you can straight away sort of reduce it to way you know use one of your favorite you know.

(Refer Slide Time: 49:25)



Covering problem solutions algorithms you know if you use the greedy solution. So, then you can essentially what I am saying you can get some kind approximation algorithm and the greedy solution itself will give you something like and approximation this is for people who are somewhat familiar with approximation algorithms. So, the problem essentially is to reduced to the following that I have some I have some sets you know I have some elements the elements are basically the regions in the polygon that I want to cover and I want to choose some sets which are basically these faces. So, that there is union of those sets you know cover all possible elements.

(Refer Slide Time: 51:33)



So, that is how I descritize the problem and then you can solve it. So, I will stop here today.