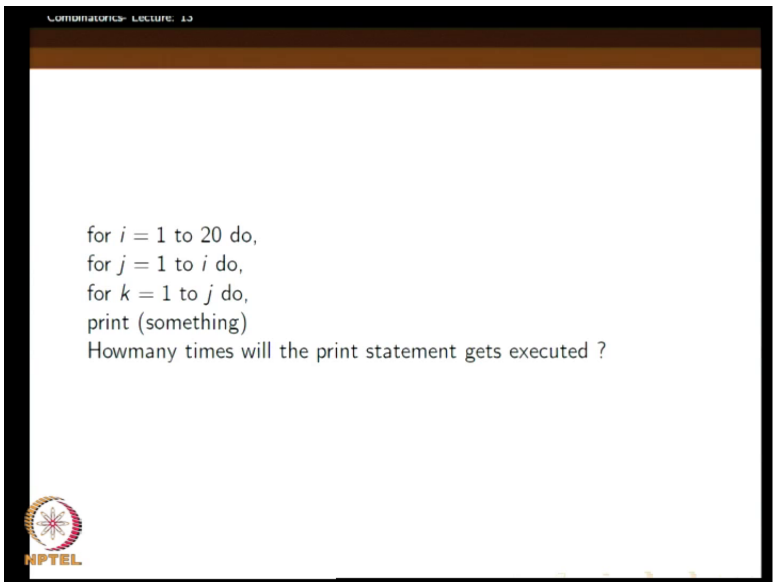
Combinatorics

Prof. Dr. L. Sunil Chandran
Department of Computer Science and Automation
Indian institute of Science, Bangalore

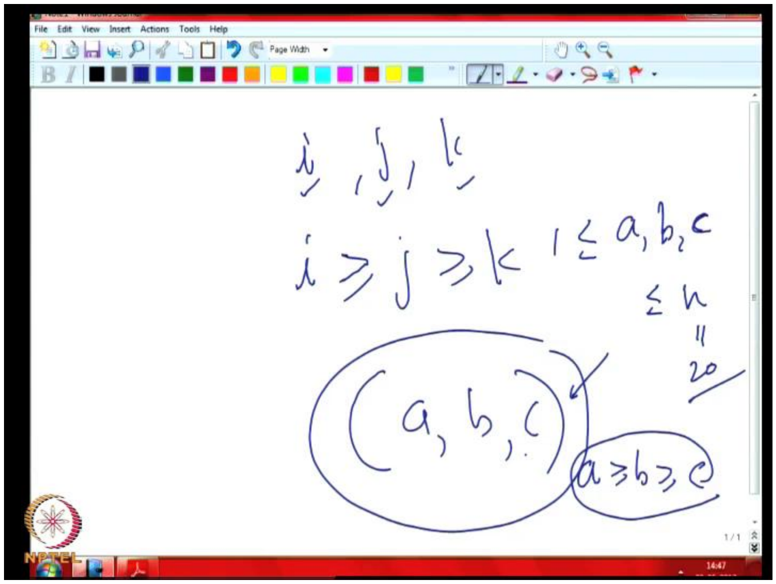
Lecture - 13
Combination of Multisets – Part (3) Bounds for binomial coefficients

(Refer Slide Time: 00:19)



Welcome to the thirteen lecture of combinatorics. In the last class we were discussing this problem, consider this program segment for i equal to 1 to 20 do, for j equal to 1 to i do, for k equal to 1 to j do, print something. How many times will the print statement get executed? So, this is a nested 3, 4 loops nested one inside the other. Now, the answer it is easy to see that by the time it reaches the print statement there will be some value for I, some value for j, and some value for k and these values.

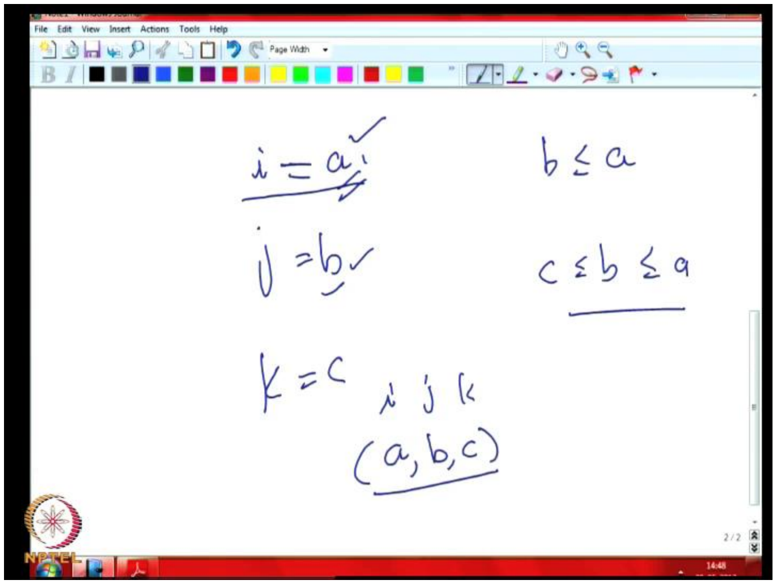
(Refer Slide Time: 01:11)



So i, j, k will satisfy this rule i always greater than j always greater and equal k, right. So, the claim is that any a, b, c, sorry. So, consider the triple a, b, c, where a, b, c, everything is in between 1 and n, say instead of 20 let us take n a, b, c; that means a comes from the set to 1 to n, b comes from the set to 1 to n, here in our problem it was n was 20. So, now we are taking n is generalize this a, b, c. And now such that this is satisfied a is greater than equal to b greater than equal to c. Consider the set of this kind of triples, right.

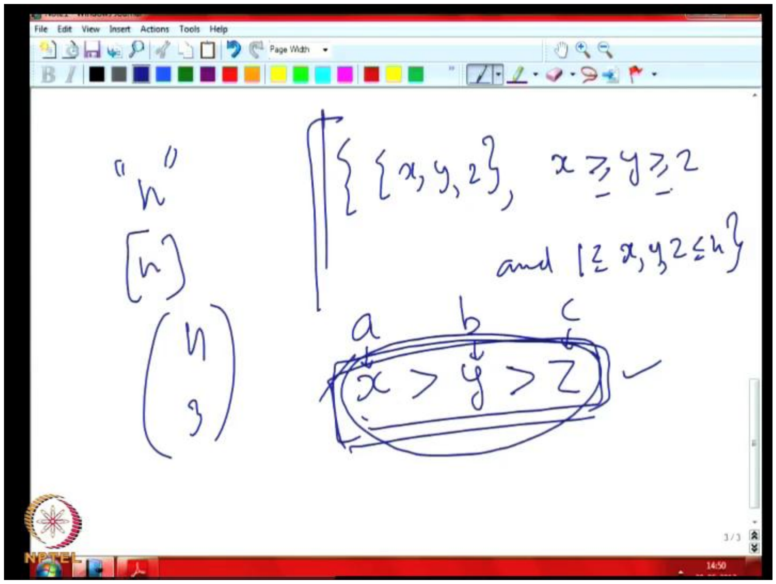
Now you know any i, j, k the value of i, j, k taken as a triple just before printing that will be from this sets, right; that means they will satisfy this conditions that a, b, c, all of them are coming from 1 to n and also a is greater than equal to b is greater than equal to c the nature of the problem, that is what is implied by that for loops, right, the way it written. Now, the other way suppose you get any triples a, b, c such that a is greater than equal to b is greater than equal to c and a, b, c, all are coming from 1 to n then you know that there is one print corresponding to such a triple.

(Refer Slide Time: 03:06)



Because see in the four loop I will reach the value a at some point of time, because i is running from 1 to 21 to n and then j will reach the value b for that corresponding value of a because j will run from 1 to a, b is definitely less than equal to a; by assumption b is less than equal to a. Therefore j has to take the value b when corresponding to i equal to a, right and then when i equal to a and j equal to b, k will also take the value c because c is less than equal to b, right. So, therefore at that time when it reaches the print statement this particular value of i, j, k will be a, b, c. So, we see that the number of print statements will be equal to the number of members of this set.

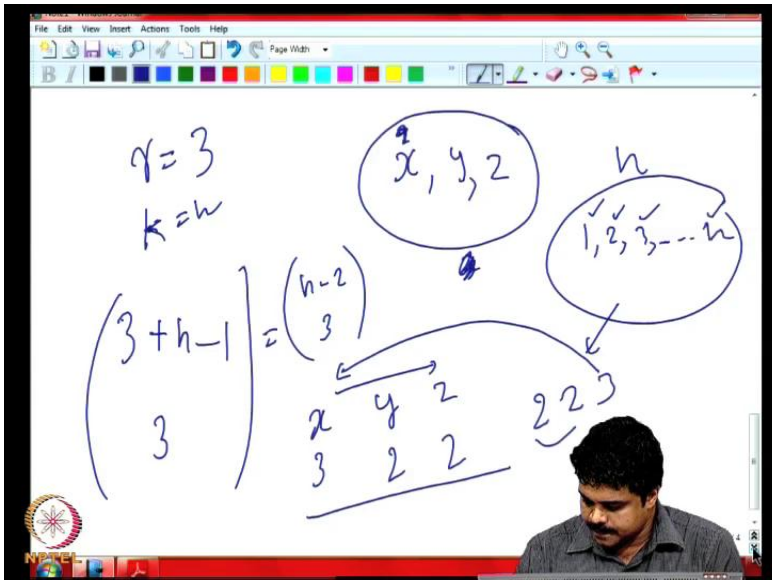
(Refer Slide Time: 04:01)



x, y, z such that x greater than equal to y greater than equal to z and 1 less than equal to x, y, z less than equal to n, right, the cardinality of this things what we are looking for. So, you see if this would have been very easy to count if your just saying x greater than equal to y greater than equal to z; that means x and y cannot be equal, y and z cannot be equal like that, right. So, then now what is the answer? The answer is just you just take three numbers out of the n numbers, see this n, right, 1 to n.

So, once you take that number a, b, c, so we can just give the biggest number to x, the next biggest number to y and the next smallest number to z because that is for any selection of three elements from 1 to n, there is only one way of assigning those numbers to x, y and z because x has to get the biggest among them, y has to get that second biggest and so on, right. So, there is no other way and similarly for instance you can see that any pair any triple corresponding to x, y, z, we will correspond to only one triple, I mean three element set from n. So, finally the answer will be n choose three, right, because any selection of three elements will satisfy this condition, right. We can assign those members of that three set to x, y and z such that this condition is satisfied, right. So, now the issue is that we have this less than equal to, so then it is easy now, so what to do is.

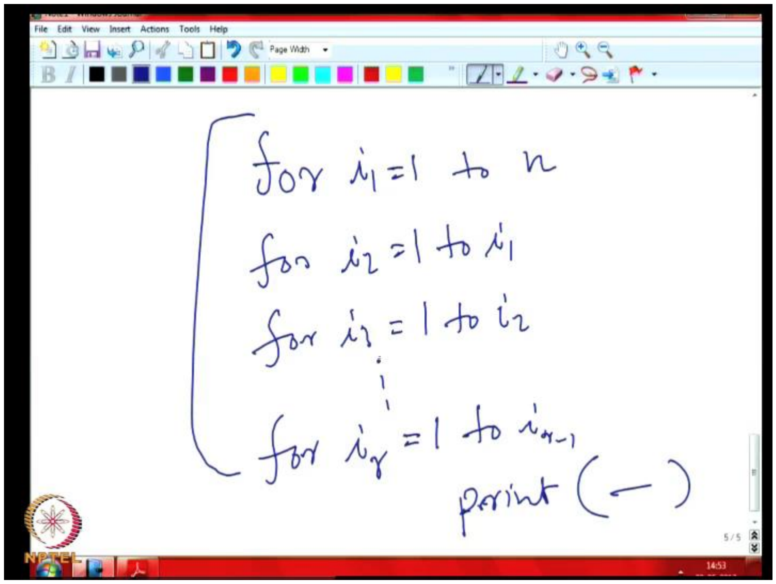
(Refer Slide Time: 05:56)



This we are selecting actually x, y, z, right, and x can be equal to y and z and these things are coming from 1 to n and then the repetition is allowed that is what; for instance we are selecting three things from n type of things, n type of things is 1, 2, 3 up to n, right, with repetition allowed that is what, right. So, you can select, say, 2 for x, not 4 x, see because once you select the three things; for instance I select 2 2 3, now there is no confusion which should be x, which should be y, because the biggest should be x, right, x equal to 3 and y should be next biggest, z should be the next biggest. It should be from x to x, y, z and the order in which you write would be non-increasing, right. The repetition comes means it will be equal to our increase, right.

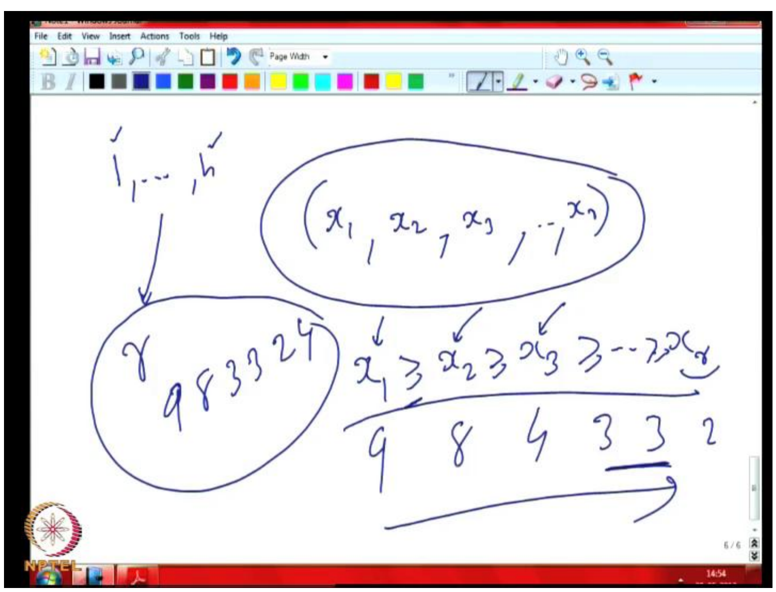
So, therefore it is just a matter of selecting three things with from n type of things, this is one type, second type, third type, n type, n things and objects with repetition allowed. So, there are any numbers of repetitions in that, right. So, we can see that the answer is because here r equal to 3 k or k or n equal to n, right, so the type number of types, right; therefore the answer is equal to r plus n minus choose r i 3 plus n minus 1 choose 3; this is how much? n minus 2 choose three. This is the answer, right, n minus 2 choose 3, correct.

(Refer Slide Time: 08:03)



So, now we can see that we can always generalize, so suppose the four loops were not just three for i 1 equal to 1 to n, right, and for i 2 equal to 1 to i 1, for i 3 equal to 1 to i 2, like that we could have written r such statement for i r is equal to 1 to i r minus 1 and then I say print, print something, right, so how many times the print statement would be executed. Now instead of three we have r things, right.

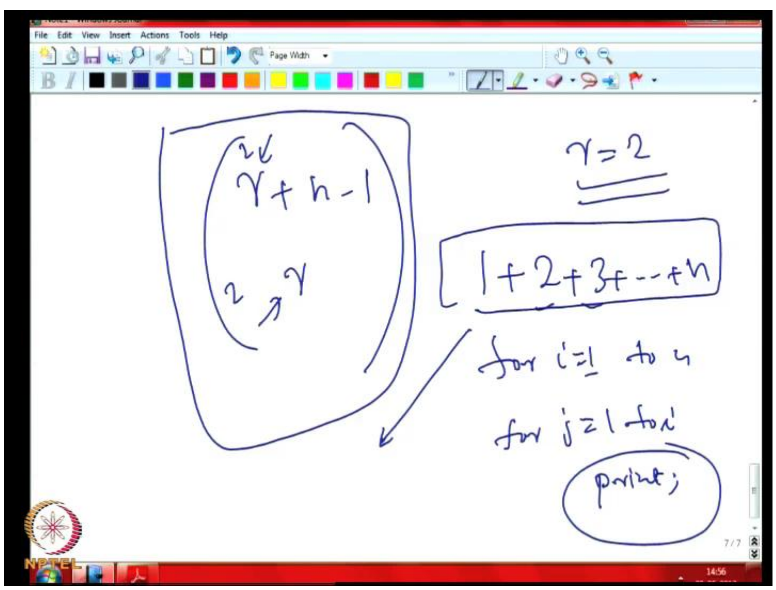
(Refer Slide Time: 08:42)



Then now we are talking about r topples, right. So, say x 1, x 2, x 3, x r should be selected in such a way that x 1 is greater than equal to x 2 is greater than equal to x 3 greater than equal to x r, right. Now you know once you select these things, the issue here is you can have equal value. So, from 1 to n you just select r numbers, right, but it is that some numbers can repeat, right. So, it is just like n types of things are there 1, 2, 3 up to n; you can take r of n but sometimes some numbers are repeating.

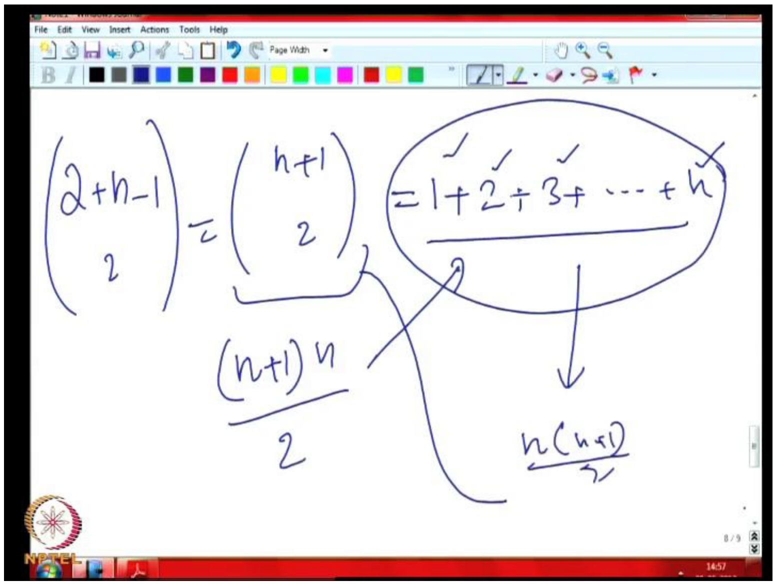
So, once you select it there is no confusion because the biggest should go to x 1, second biggest should go to x 2 and third biggest should goes to x 3. If there are ties we just keep on writing the same number and then till the last, right. So, for instance if you have selected 9 8 3 3 2 4, then we will give 9 8 4 then 3 3 2 like that and they decrease non-increasing order, right. It is not really decreasing here for instance is equal but non-increasing, right. So, therefore it is only a matter of just selecting r things with repetition and allowed from n 1 to n, right.

(Refer Slide Time: 10:12)



So, therefore that answer is r plus n minus 1, choose r that as we have already seen. So, when r equal to 2 for when we started with r equal to 3 to make it more nontrivial, then we discuss the r equal to 2 case and show that 1 plus 2 plus 3 plus n is the answer, remember for i equal to 1 to n, for j equal to 1 to i, print, right. This statement how many times print statement will get, so when i equal to 1 j will be 1, j will go 1 to 1. So, i equal to 2 j will go from 1 to 2; that is two times and whey i equal to 3 j will go from 1 to 3; that means three times and so it is just adding up these things and totally this sum number of times the print will be executed, right. Now we see that we could have also computed in this way, right, when we can put r equal to 2 here and then what is this happening? So, r equal to 2 here, the value of r is assigned 2, so that is 2 plus n minus 1 that is n plus 1, choose 2, right.

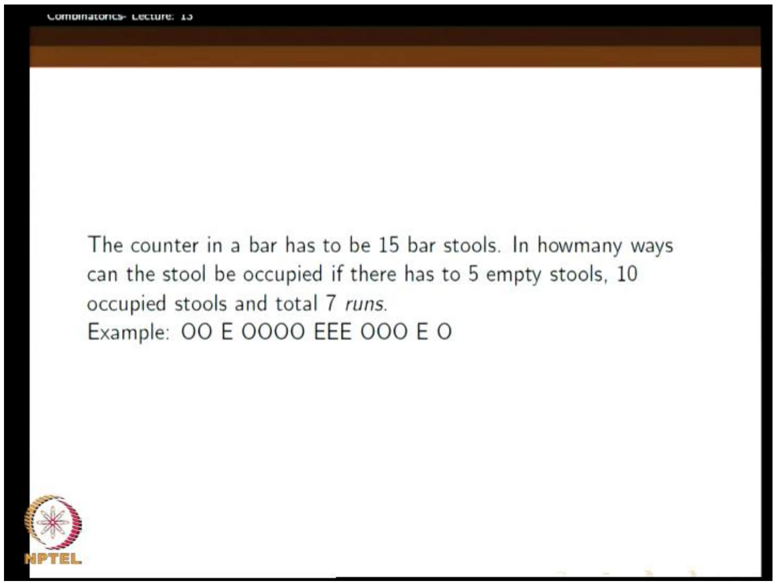
(Refer Slide Time: 11:27)



This n plus 1 choose 2. So, we are seeing that this will be the same as that sum, right, 1 plus 2 plus 3 plus up to n. So, this is what. This is our familiar n plus 1 into n by 2, this will be the sum, right. So, in other way this is a combinatorial proof that this sum is equal to n into n plus 1 by 2, right. What have you done? We have observed that when only two for loops are there, so this is actually the number of times the print statement will get executed will be this; this sum 1 plus 2 plus 3 plus this thing but another way of counting it is to count all the pairs to two element subsets from n with repetitions allowed.

That is we know that because there are n type of things and r equal to 2 that is 2 plus n minus 1 choose 2. So, this is exactly equal to n plus 1 choose 2. So, this both have to be same, so we see that this 1 plus 2 plus 3 plus up to n is equal to n plus 1 choose 2, so that is n into n plus 1 by 2, right. So, therefore that is why we say that we have combinatorially proved that this sum gives when you add up this 1 to n we get this number, right. So, yeah next one, so we will consider the next question of combinatorial.

(Refer Slide Time: 13:18)



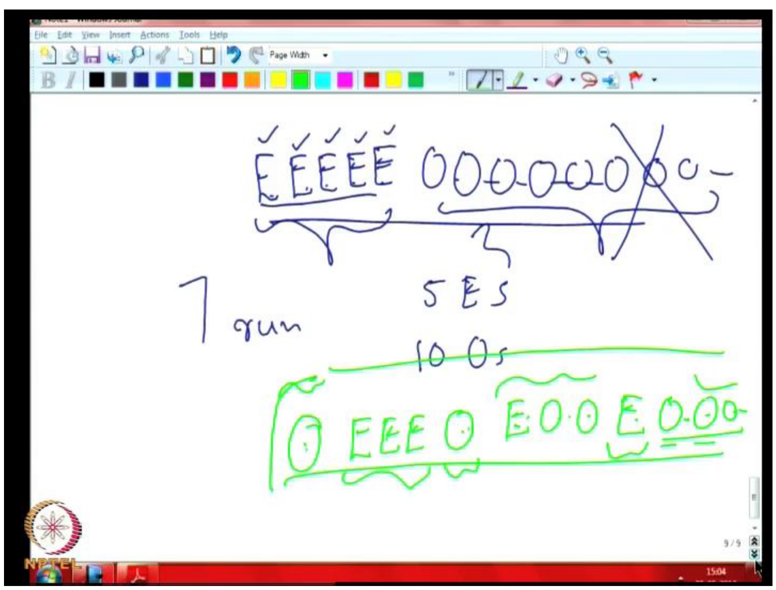
So, here is a different question. So, some customer goes in to a bar. So, in the bar he sees in the counter there are 15 bar stools in a row you can assume they are in a row there are 15 stools, right, and then he wants to occupied some stool, right, but he looks at the pattern of occupation. He just see's that some stools are occupied, some stools are not occupied. So, he can write down the pattern he sees. So, the first two stools are occupied, then occupied, occupied, then it is empty. Next one is empty, next is occupied, next is occupied, next is occupied, then you see empty, empty, empty, three seats then are empty, then occupied, then occupied, then occupied, then empty, then occupied, right.

So, from his perspective when he looks at he does not care about who is sitting on the stool; he is only worrying whether a stool is occupied or not and he is reading O and E, occupied or empty, from left to right and he can write down on a straight line OO EE OOE OOOO like that depending on. If eighth seat is occupied he will write in the eighth position O, if the eighth seat is unoccupied that means empty then he will write E in the eighth position. So, he says that this is some pattern, right, he observes.

So, he will say that now there is a run of OO first run, then this is just E that is to another run, then there is a run of four O's here and then there is a run of three E's here and there is a run of three O's here, then there is a run of one E, there is run of one O. So, he will say that in this pattern there are one, two, three, four, five, six, seven runs. So, 15 letters are written where each letter being either an O or E. A collection of a sequence of O's flung by two E's starting after an E and then the collection of O's the sequences of O's till the next E is called a run, right.

Similarly a collection of E starting just after an O and then running till the next O is encountered is called a run of E's. So, now we are interested cause in this question we are interested to count the number of patterns where there are exactly seven runs. So, this kind of questions comes in statistics. So, why this is done is unimportant for us; we just want to how many patterns will have seven runs that is what we are asking. So, we consider and we deal with this question in the following way.

(Refer Slide Time: 16:29)

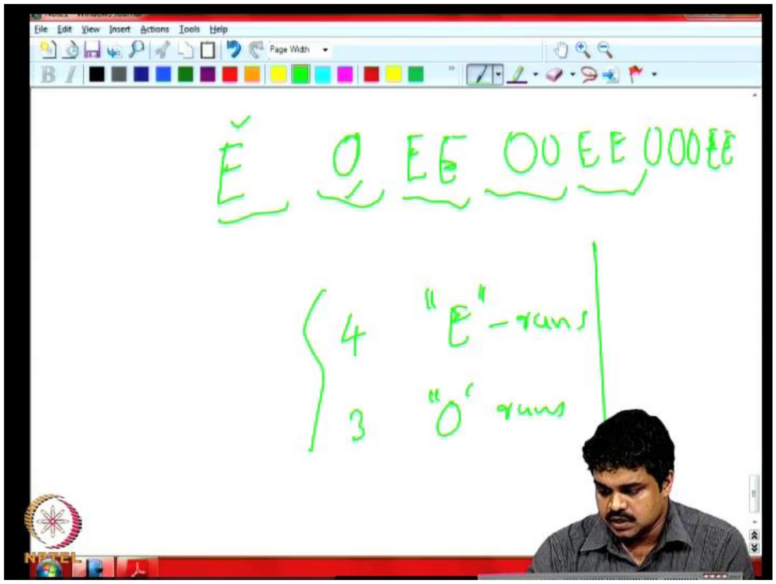


So, we want find for instance example seven runs, right; for instance I can start with E, then I can start keep on writing E's, right, how many E's I can write? So one, two, three, four, five, six, so okay seven runs I want and also there is a condition in that problem that if out of the 15 stools 5 should be empty and 10 should be occupied; five that is that is the first thing. So, we have two condition, there are 15 stools; that means sequence of 15 letters is written, out of that 5 are E‘s 5 E’s we should see and then remaining 10 should be O’s, right. So, suppose if you write initially itself 5 E’s, then we will have to write all the remaining 10 to be O’s, right. So, that means we will only get 2 runs right because this is 1 run and this is another.

So, we will not get 7 runs. So, this will not be a valid pattern for us; we are not counting this at all, we are not interested in this but we are interested in patterns like for instance I start with three E’s, right, then for instance okay if I start with three E’s also, we cannot do these things because then we cannot create, right. If we have we can put an O here, then we can put something like this, say, one two, say, and then remaining are all this. So one, two, three, four, five, six, seven and then till ten O’s I will put.

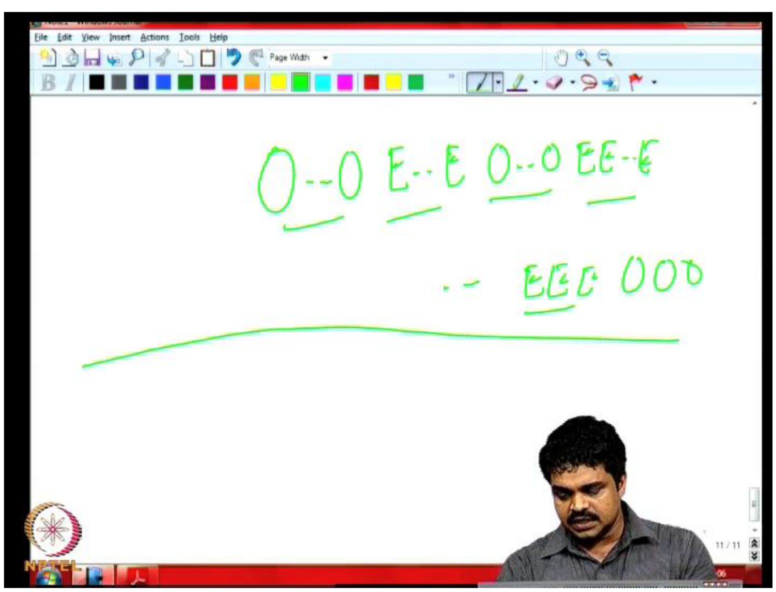
So, this is one run here and another run here, third run here and another run here and another run here, another run here, how many runs? One, two, three, four, five, six, yeah so maybe we cannot. How many runs? One, two, three, four, five, six, seven, right; seven runs are there, this is valid, right. So, now how many patterns are valid in the sense that in the sequence we have exactly 15 letters; each letters is either an E or an O and then total number of E’s is equal to five, total number of O’s is equal to 10 here also we also want to make sure that the number of runs is equal to seven. So, you can notice that if the number of runs is seven we can do it in two ways.

(Refer Slide Time: 19:40)



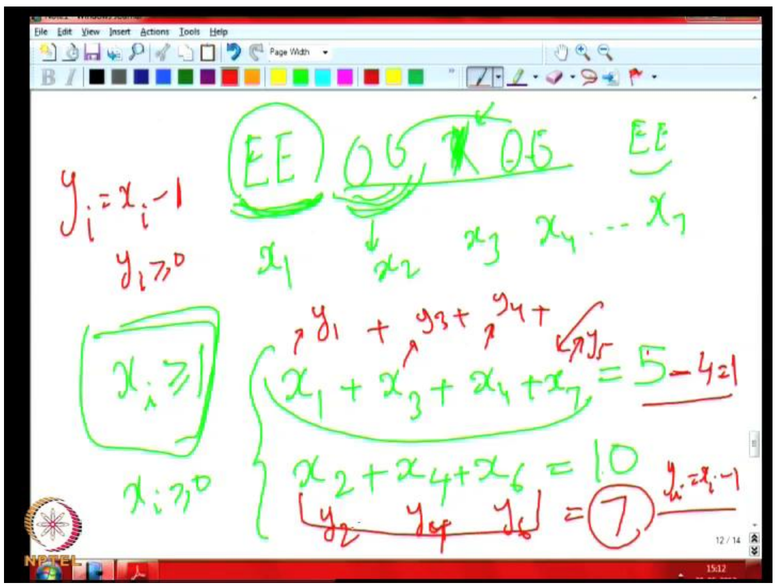
Either we can start by E’s; if we can start with E’s, write a run of E’s but then how many runs of E’s should be there because total is seven and after one run of E we should get a run of O, then there should be a run of E’s, then there should be a run of O’s, then there should be a run of E’s, right, total seven. So, we start with one E, so by the time we finish six runs then the last is O, the next is seven, right, seventh on should be E, right. So, if we start with E’s then there are four E runs, right, and three O runs.

(Refer Slide Time: 20:39)



Similarly if we had start with O it will reverse; if we start with O then what will happen? So, you start with O then you will have to start with E's, next will be E, next will be O, next will be E. So one, two, three, four, by the time the sixth run is will be E, because the second is E, forth is E, sixth is E, then the seventh will be O we will be ending with. So, there will be an extra O than the E's; so that means they will be four O runs and three E runs and these are the disjoint sets. So, we can count the number of patterns which satisfy all our property but starting with any run and then I add the number of patterns which satisfy all our condition plus the first run is an O run, right, because these are disjoint sets we can just add them together, right, now the question. The question is that what is the question?

(Refer Slide Time: 21:49)

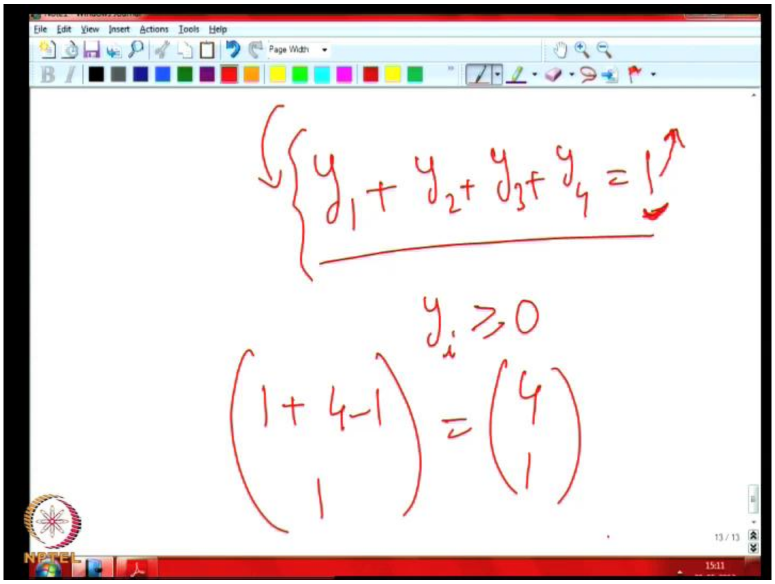


So, how many patterns are there with starting with an E run? EE OO then E something like this, right, and ending with an E, right. Now you know suppose the number of E's here, the first run let us call x 1 number of E's, right. This a variable; we can we can fix a value for x 1 integer value positive integer value and this should be x 2, say, number of O's, right; that is x 2 and then in the third one; that means any run third E run that x 2 will give you the number of E's there and x 4 will give the number of O's in the fourth run and x 7 will give you the number of E's in the last run. So, we know that x 1 plus x 3 plus x 4 plus x 7; that means the total number of E's is equal to 5; that is the power part of our condition and similarly x 2 plus x 4 plus x 6, the total number of O's is 10, right.

Now you see how many ways we can do this thing but you see this each x i has to be greater than or equal to 1 here, why? Because you know if it is an empty one the run we would not have 7 runs because if this was an empty then what will happen is this is empty right. So this will be continuous run of O’s that will be counted as one only and this was also this empty run is not counted at all, right. So, therefore every run should contain at least one E, similarly at least on O if it as an O run, right. So, therefore x i should be greater than equal to one. Now if x i was just greater than equal to zero, how many possible ways we can assign values to integer values to on x 1, x 2, x 3, x 7 is easy because you know we have seen that this is 5 is the value r there, the number of types is 4 here.

So, 5 plus 4 minus 1 choose 5 will be the value, right, but then now we know that this is just like everything should get a value of at least one. So, it is like filling and putting the balls in at the container such that no box is empty; that is the equivalent problem in the box and bin type but here we are saying that every x i's should get at least one. So, we can convert the problem to, say, we will defined y 1, y 3, y 4 and y 5 correspond to this thing and they should add up to now they should add up to, so what is this y i? y i is equal to x i minus 1, right because you know each x i has to be at least 1, so y i is x i minus 1 it should be at least zero. So, y i has to be at least zero because x i minus 1 is put, now the thing is when you add up these things x 1 plus y 3 plus y 4 plus y 5, we have to also minus from the other side, right, so 4. This will become 1, right, 5 minus 4 is equal to 1, right.

(Refer Slide Time: 25:22)

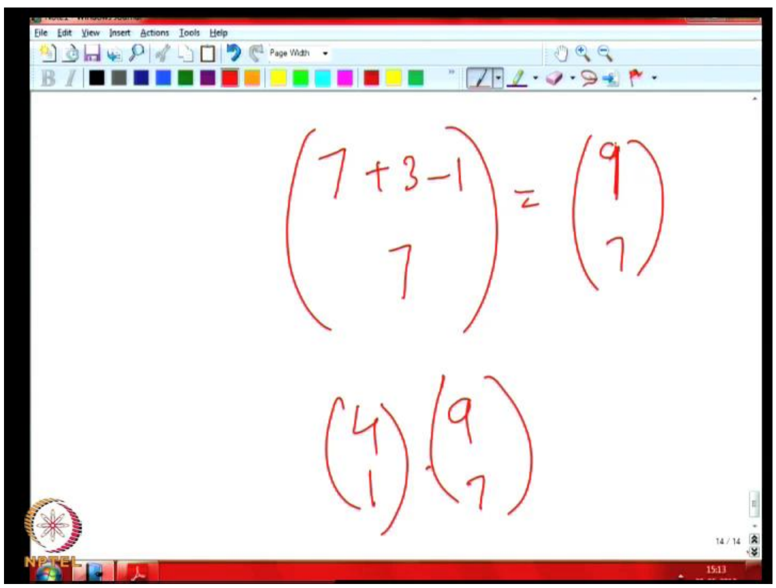


So, our question is like how many solutions are there? Therefore this equation y 1 plus y 2 plus y 3 plus y 4 is equal 1; you see any solution with the condition that y i is greater than equal to zero. you see any solution for this thing will correspond to a solution to the earlier problem if we just add one to each of the values, I mean whatever y 1 gets we will add one more, whatever y 2 get we will add one more. So, that will give a solution to that x 1 plus x 2 plus x 3 plus x 4 equal to five problem because we are adding one to each of them. This answer also will be 1 plus 4, right. So, reversely or conversely if we get a solution to that problem we can take a solution each of them are x i greater than equal to one, so therefore we can minus one from each of them.

So, now we will get a corresponding y i which is greater than equal to zero. And they will sum up to four less than what the x i's were summing up to so because the x i's were summing up to, the sum will correspond to one here. So, therefore it is enough to count this and this we know very well how to count because r equal to one here and then number of types is four. So, our formula say it is 1 plus 4 minus 1 choose 1. So, this is 4 choose 1, right. There are these many ways of selecting your E's and then that is not enough. What are our assumptions we are starting with E's; therefore there are four E runs and then how many O runs are there, four E runs are there and this four E runs can be selected in four ways as we have seen, right.

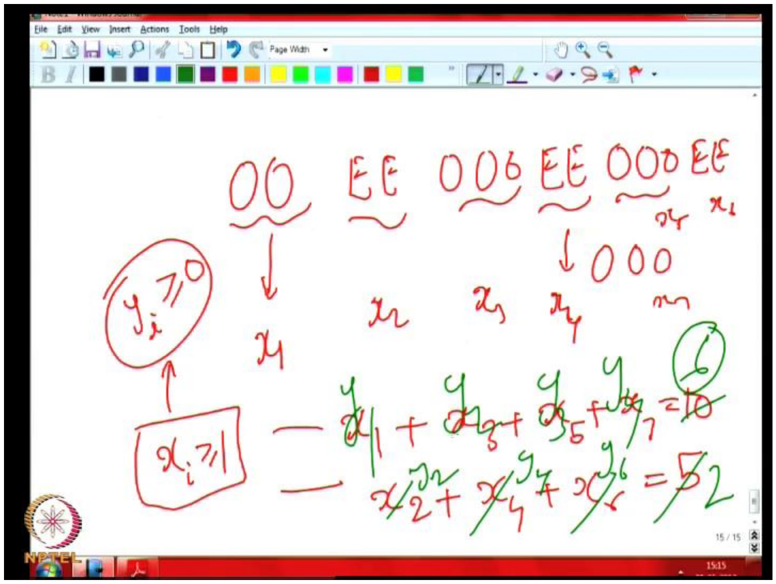
Now the other one, so the other question was how many O runs can be there? That is equivalent to the number of ways we can assign values positive integer values to x 2, x 4, x 6 such that they sum up to equal. So, remember each x 2, x 4 and x 6 has to get at least one as you are telling because their runs of O's and they should have at least one O there, right. But to convert it to the familiar problem what we do? We will convert it to y 2, y 4 and y 6 as we always do and then we will say that they should sum up to 7 because each y i is defined now as x i minus 1 and then so the condition on y i is only that y i greater than is equal to zero. So, therefore we can apply our earlier this thing with r equal to 7 and there are three types of things.

(Refer Slide Time: 28:19)



So, the answer will be 7 plus 3 minus 1 choose 7. So, this will be 7 plus 2, 9 choose 7, right. So, now we can multiply by 4, choose 1 into 9 choose 7. This will be the number of patterns where we start with any run, right.

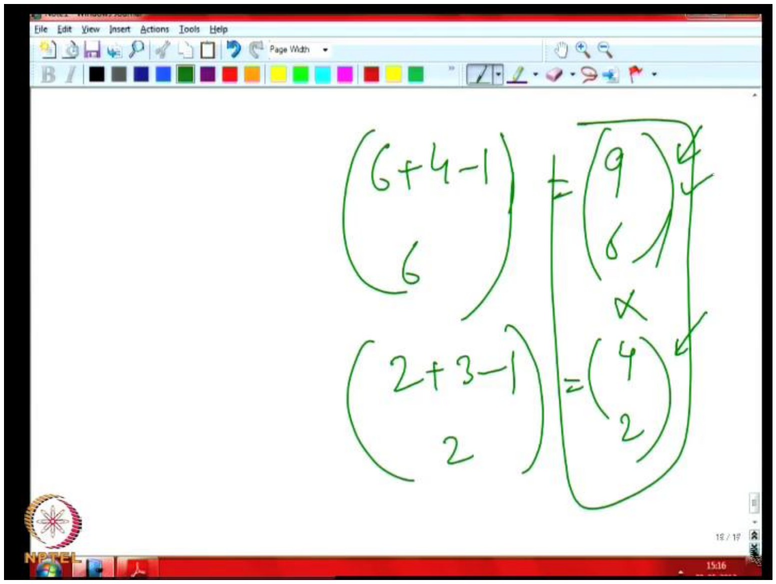
(Refer Slide Time: 28:49)



Now we consider the O runs, right. So, if we start with an O run then there are four O patterns O runs and then three E runs; that is a error because we start with O runs, then middles will be E run and then O, then E, then O, then E, right; one, then one more O, right. So, let us say the number of O’s is x 1 now, the number of E’s is x 2, number of O’s is x 3 here; same way this is x 4 and this is x 5, this is x 6 and this is x 7. So, we also know that x 1 plus x 3 plus x 5 plus x 7 has to be equal to 10 because then total number of O’s in the pattern is 10. Similarly x 2 plus x 4 plus x 6 has to be equal to 5 because the total number of E’s in the pattern is 5, right.

Now similarly because the condition here is each x i is greater than equal to one as we were telling. So, we converted to y i in both cases; y i has to be greater than equal to zero. So, for instance when we substitute by y 1, y 2, y 3, y 4 here, this will become six instead. Similarly when you substitute by y 2, y 4 and y 6 here, so this will become 5 minus 3 that is 2, right. So, now we can see that the total number of possible ways to give values to y 1, y 2, y 3, y 4 is 6 plus 4 minus 1 choose 6, right.

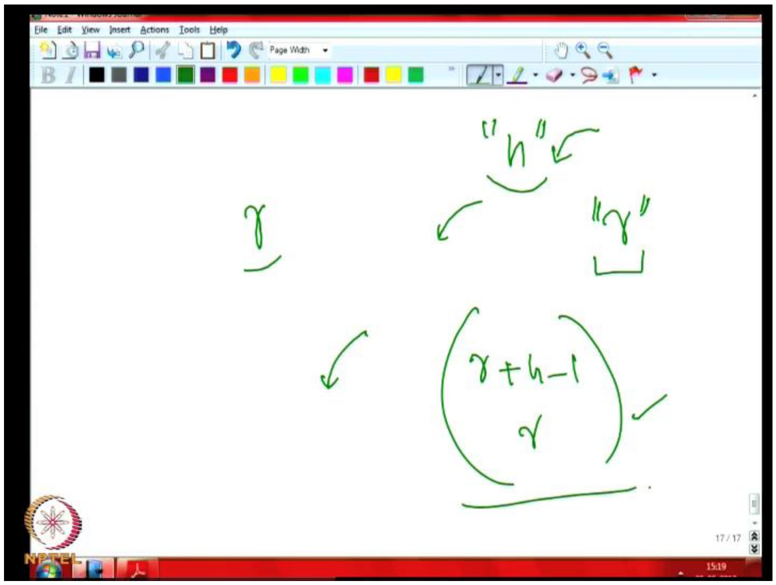
(Refer Slide Time: 30:38)



6 plus 4 minus 1 choose 6; that is 9 choose 6. Similarly the number ways to assign values to this thing is 2 plus 3 minus 1 choose 2, this is 4 choose 2. Now multiply them together because these are the number of ways we can decide the E's, the number of ways we can decide. So, these are the number of ways we can decide the O's and these are the number of ways we can decide the E's. Now we multiply them together, we get the number of patterns such that it start with an O run and now after finding this product we have to sum this with the earlier one we have found out namely this, right, because this was the number of patterns which starts with an E run, right. So, therefore if you add these two things together we will get the final answer, right.

So, this was this thing to the question. So, like that we found out the total number of patterns such that it is composed only two types of letters E's and O's and total number of runs is equal to 7 and number of E's is equal to 5 and number of O's is equal to 10. The strategy was to split them into two disjoints sets namely the patterns which start with an O run, the patterns which starts an E run and the calculation for computation how we found out the cardinality each of these sets was more or less similar; we just wrote we figured out how many E's should be then there in each of the E run and then how many O's should be there in each of the O run by writing those equation and evaluating the number of solutions, right. So, that was the thing and then yeah, now I think we complete the discussion and some examples about the multisets there.

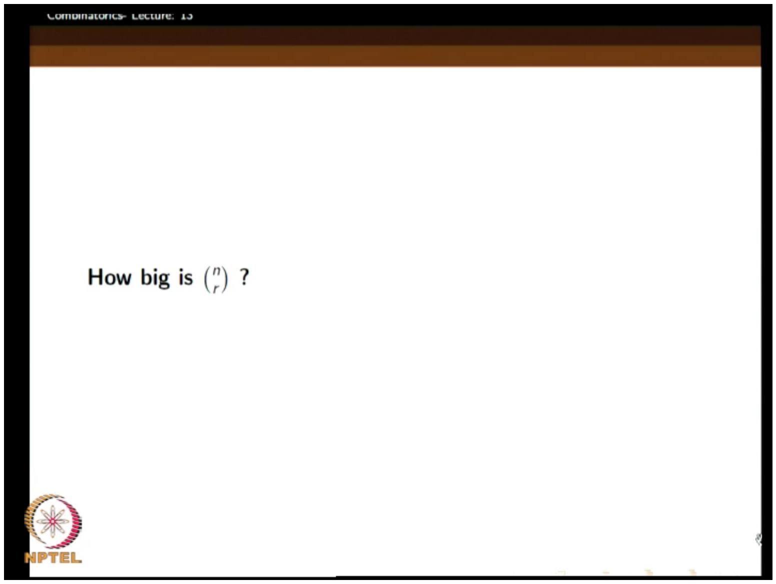
(Refer Slide Time: 33:15)



So, remember the question we were considering is that there are n objects, so we can say n type of objects and then we were making r set of r an r set from this thing but in this r set everything is not distinct, it is a multiset; that means these objects can repeat, means because if we are considering type of objects then a particular type of object can repeat several times that is what it says, right. But then we assume that it can repeat any number of times in it. So, infinite number of times is what we see but actually we do not need infinite as long as each thing can repeat r times that is enough for the formula we develop; that means r plus n minus 1 choose r.

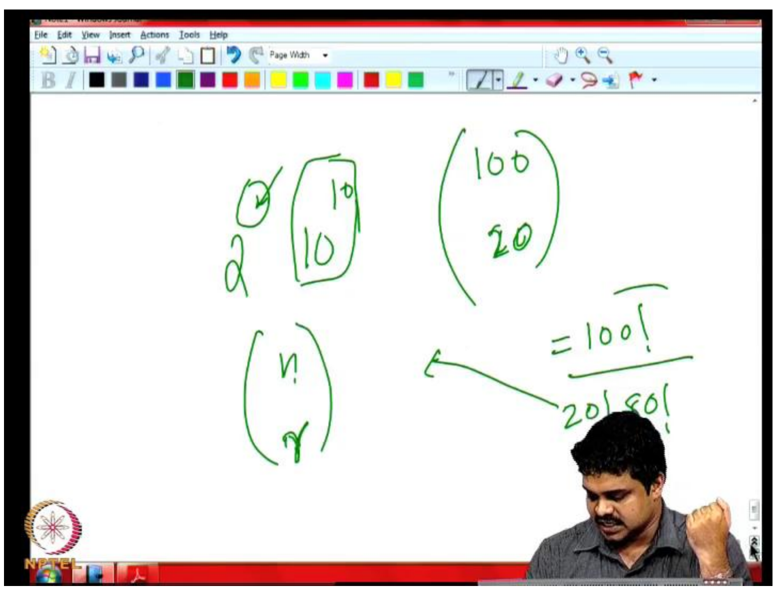
This is the formula to work because infinite number or what infinite number or finite number we cannot take any objects more than r times because any ways we are selecting only r things, so that is as good as infinite for us, right. So, therefore as long as there is r copies of each object each type of object that is more than enough. So, therefore this formula will work but on the other hand if the number of object of certain type are less than r, then this formula will not work and we will discuss this problem later and we will come back to that but it is not easy to get a general formula directly. We will try to find out some ways of dealing with that problem later.

(Refer Slide Time: 35:07)



So, now the next issue is this is something like which we have to be bothered about. How big is this n choose r? Of course we can prove identities.

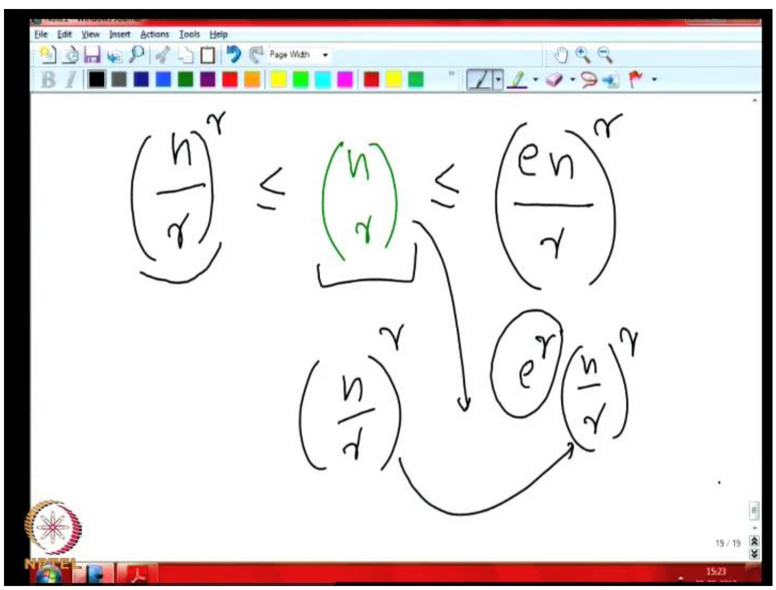
(Refer Slide Time: 35:30)



We can take a problem and show that the total number of things is some 100, choose 20 but then these numbers are not the usual numbers we use. So, this is something like we know that this is 100 factorial divided by 20 factorial into 80 factorial but then it does not make any sense, right, how big will be this? Of course I cannot get a feel of what is 100 factorial but still that number is just a product; rather we are more familiar seeing certain numbers of the form 2 to the power or may be 10 to the power and things like that, right.

So, it would be nice to compare with more familiar forms, so once n choose r i is given it would be nicer to known how big it is to get some upper bounds; I mean at least I know that this not more than this and this is not less than this and things like that upper bound lower bound, right. This is what our next interest is, so is this what we are going to do. So, let us say n, see of course most of the time when we are just studying for examinations it may not be like very relevant probably, but then if it is not in your syllabus of course but when you start doing research in many cases and many practical cases we will have to know a little more about how big this stuff is.

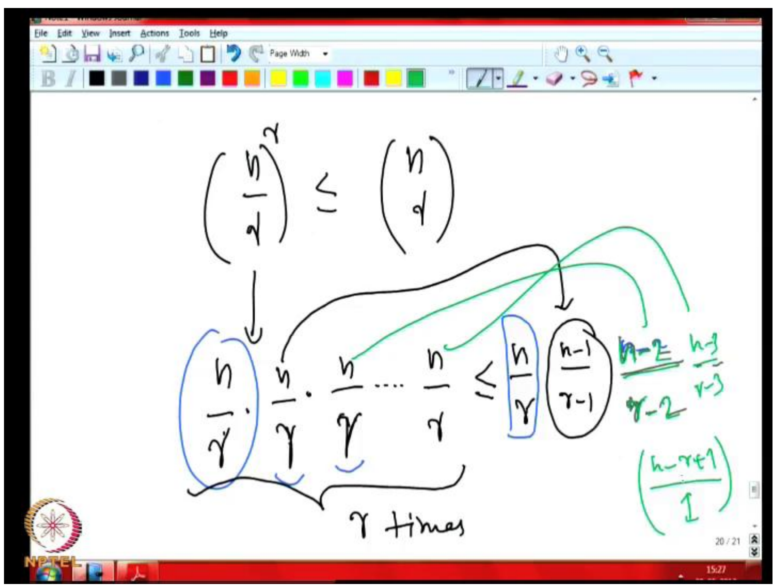
(Refer Slide Time: 36:57)



This n choose r is good enough for some person as long as he is not bothered about how does it compare with more familiar forms. So, here especially we want to compare with n raise to r kind of forms or something raise to the r; that seems to be more familiar, one can argue why is it more familiar. For instance if I always work with n choose r will I get more family with n choose r? Probably but still like intuit it looks like the n raise to r kind of forms are more familiar. Therefore we will try to come up with some upper bounds and lower bounds for this thing in terms of such powers.

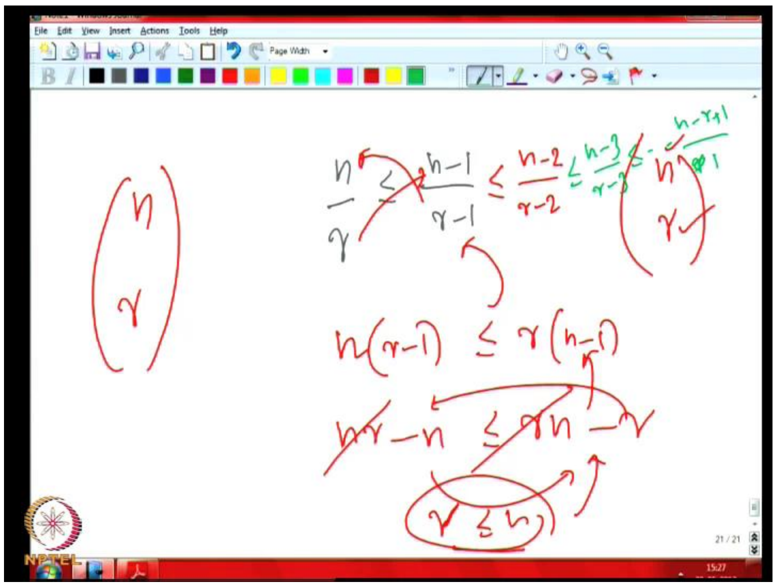
So, here the lower bound we are going to give is n pi r to the power r and the upper bound we are going to give is e n by r to the power r; e is the usual e, so the constant e and n by e to the power r and n by n choose r is less than equal to e n by r to the power r. So, here you see that the lower bound is n by r; see the lower bound is n by r to the power r, the upper bound is e n to the power r is e to the power r into n by r to the power r; that means it is e to the power r more, right, e to the power r more. So, we are seeing that this is sandwich in between these two quantities, this and this, so in this gap of; see this lower bound is multiplied by e to the power r that much big. So, in between somewhere this stays. So, many times this is very handy to use; therefore we will prove it.

(Refer Slide Time: 39:02)



So, let us look at the lower bounds name first n r is greater than equal to n by r to the power r, how will you prove it? So, what is n by r to the power r? This is n by r into n by r into n by r into like that we have to write several times, how many times we have to write? The number of times we write this is r times and this is less than equal to n by r into n minus 1 by r minus 1; can i replace this number by this and say that it is getting bigger. So, what I am planning is every first term I am keeping it just like that, here I copied it; the second term I am replacing with n minus 1 by r minus 1. So, if this n minus 1 by r minus 1 is bigger than n by r then we are okay; we are putting this greater than equal to here. So, it is only becoming bigger; here this will be replaced by something like n minus 2 r minus 2 and so on, right. We have to show that at every stage we are replacing with every term is replaced with a bigger term.

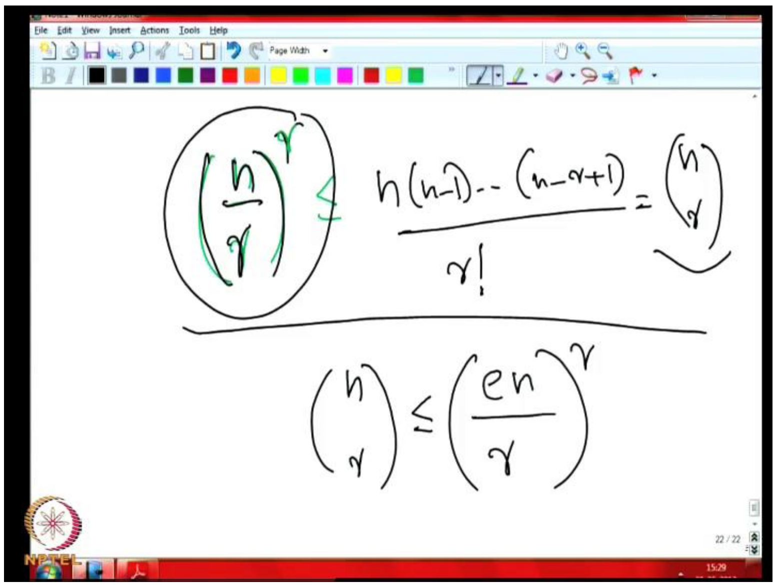
(Refer Slide Time: 40:38)



So, now for instance you see in general we want show that n by r is less than equal to n minus 1 by r minus 1, you just multiply like this, this and this is multiplied, you get n into r minus 1. If this is correct, the earlier one is also correct r into n minus 1. So, this is what n r minus n is less than equal to r n minus r, so cancel this off. So this here, this here; that means if r is less than equal to n, yes it is because whenever you say n choose r, so we mean r is less than equal to n, okay at least in the case after now we have discussed n and r both are positive integers and then when r is bigger than n; of course n choose r will be zero, we no need all these bounds. So, therefore we are only thinking about reasonable case namely n is bigger than r, right, n a positive integer and r is smaller than equal to n.

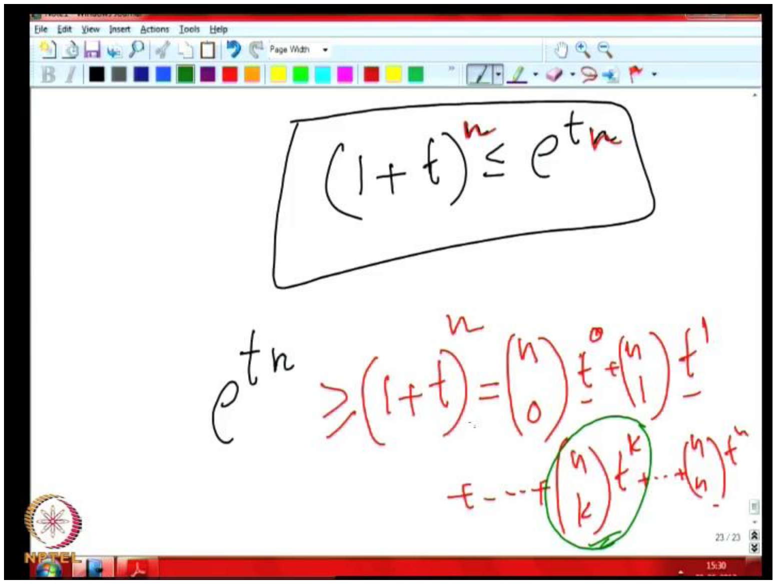
So, therefore we have this thing; therefore this is true, this is true and therefore this is true, right. So, therefore the next one of course we can instead of n if we can take n minus 1 this will be less than equal to n minus 2 by r minus 2 and then it will be less than equal to n minus 3 by r minus 3 and so on. We can keep on going, so n minus, right, how many times we have n minus r plus 1 by 1, right. So, r times we go this will keep on increasing; so that is what we are seeing. So, here we can do this n minus 2 n by r minus 2 and then next one will be this is this; the next one will be replaced by n minus 3 by r minus 3 and so on. How many terms are there? r terms are there, right. So, then the last term will be n minus r plus 1 divided by r minus already starting with that will be just 1, right, r minus 1 just 1. So, above what you get is n into n minus 1 into n minus 2 into n minus 3 into n minus r plus 1, then below what you get is r into r minus 1 up to 1; that is r factorial.

(Refer Slide Time: 43:14)



This is therefore we get n by r to the power r is less than equal to maybe you can use this one n by r to the power r is less than equal to n into n minus 1 into n minus r plus 1 divided by r factorial which is essentially n choose r; so that way lower bound is proved. Now this is the lower bound, right. Now to the upper bound; upper bound is n choose r is less than equal to e into n divided by r whole power r. Compared to the lower bound we proved this e to the power r more; that means when you multiply this lower bound by e to the power r we cross this n choose r as what we are seeing, we go beyond that n choose r, right. So, how do we prove this thing? It is a little trickier than the earlier one that was just direct substitution.

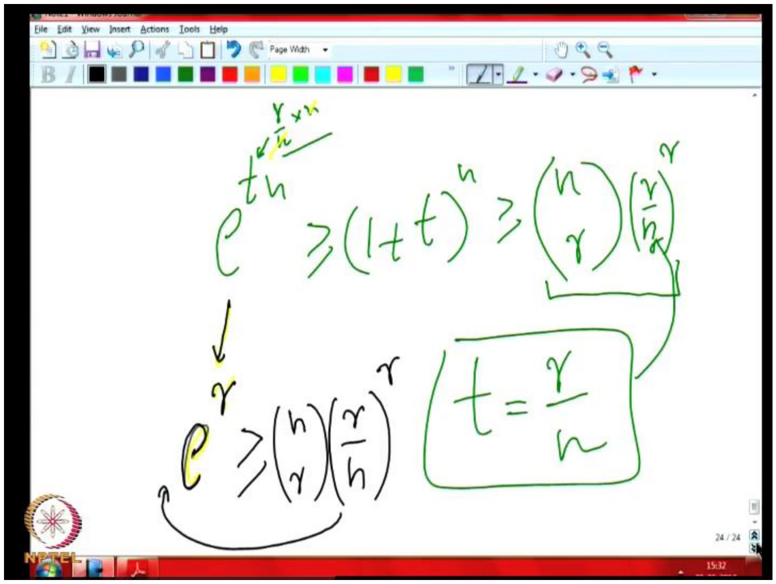
(Refer Slide Time: 44:27)



Here what we do is we start with something like we have an inequality. So, 1 plus t, this is not that taking from calculus, we will not prove this thing. This is very well known 1 plus t is less than equal to e to the power t. So, we start with e to the power t and power it by n, right. So, we can power it by n; that means here a n will come, right. So, that is what. We power by n here, then a n will come here. So, this is greater than or equal to 1 plus t to the power n, right, but then now this can be expanded using the binomial theorem which we have learn n choose 0 into t raise to 0 and then n choose 1 into t raise to 1 plus n choose k into t raise to k and all the way to n choose n into t raise to n, alright. So, this is the sum.

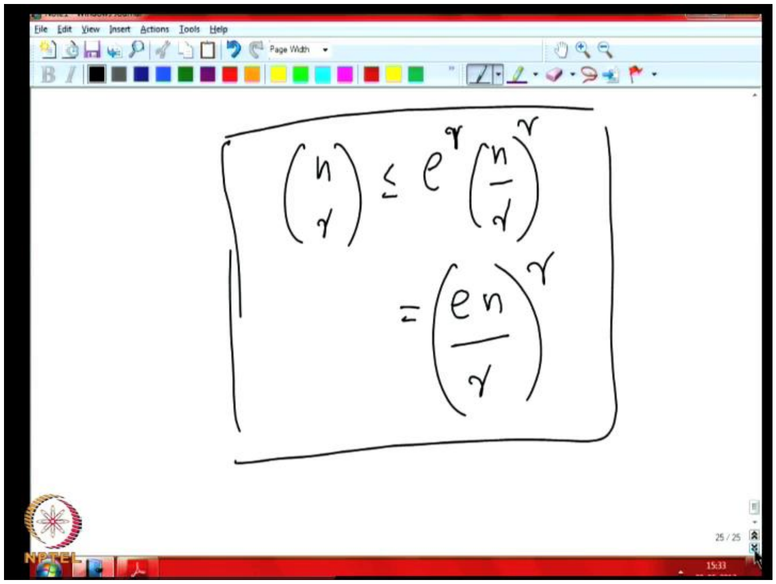
Now these are all positive terms n choose 0, n choose 1, n choose k, everything is positive term and then yeah, so if you give a positive value to suppose you are planning to give a positive value to t, so then these are all positive things. So, therefore we can say that any individual term I take from here; for instance I take this term that is going to be less than equal to this, so I can write like I just take this term and this is less than equal to 1 plus t raise to n that is less than equal to e raise to t n.

(Refer Slide Time: 46:18)



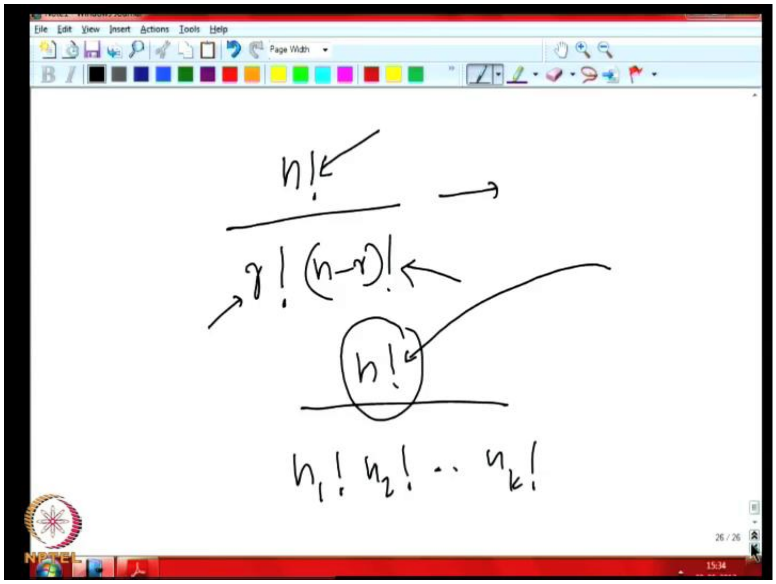
So, this is e raise to t n greater than or equal to 1 plus t raise to n is greater than equal to n choose k to the power t raise to k. Now take a positive value for t, t equal to, say, k by n; k is positive, n is positive. Put t equal to k by n, then what will happen? Sorry I just changed the variable, so here I will take r. So r, t raise to r, okay. Now put t equal to r by n. So, when you put t equal to r by n here we are replacing by r by n whole power r, then yeah, so this is n choose r into r by n all power r here what will happen? For this t we have to put r by n, so this entire thing will become t is equal to r by n, r by n into n this will cancel off, right, this will cancel off. So, this will turn out to be e to the power r. So, I can write it e to the power r; this is what is coming from here and now is greater than equal to, this we can discard here n choose r into r by n whole power r, right. Now we can bring this to this side.

(Refer Slide Time: 48:09)



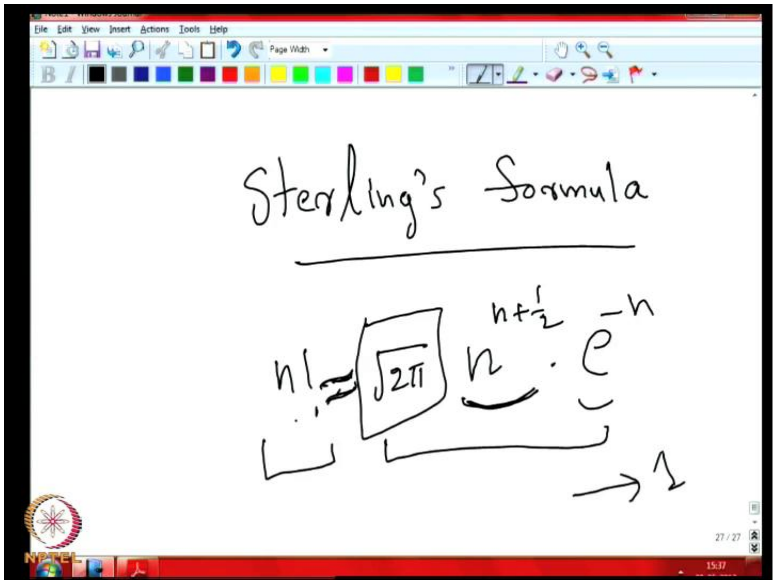
So, what we get? n choose r is less than equal to e power r into n by r whole power r, right. This means e into n by r whole power r; this is going to be the upper bound, right. So, we have proved lower bounds for n choose r and upper bound for n choose r. Of course for instance n choose r is very important which keeps appearing in various places. So, therefore this upper bound and lower bound comes very useful. So, it is good to remember it; for instance if you really work under search problems many times we will have to substitute this to makes sense of what we are doing. So, yeah it is very handy; it is very easy to use. So but on the other hand if you insist that okay, tell me something more precise.

(Refer Slide Time: 49:12)



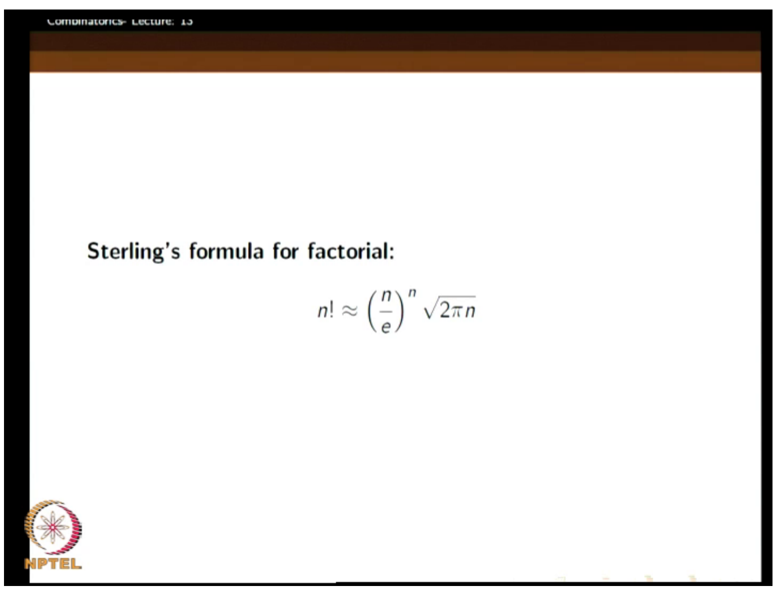
So, this is the n choose r was this, right. So, in principle I should substitute something for n factorial, substitute something for r factorial and substitute something for n minus r factorial and if you do the manipulation we will get something, right. For instance one may ask suppose sometimes I have a multinomial coefficients something like n factorial by n 1 factorial, n 2 factorial, so what will I do for this thing? So, you tell me something which I can use in all these situations; that means n factorial should be substituted by something, all this factorial should be substituted something.

(Refer Slide Time: 50:03)



So, in this case what we can use which is generally very useful to remember is something called Sterling's formula. So, I will show the formula so that it is easier than writing it here.

(Refer Slide Time: 50:18)

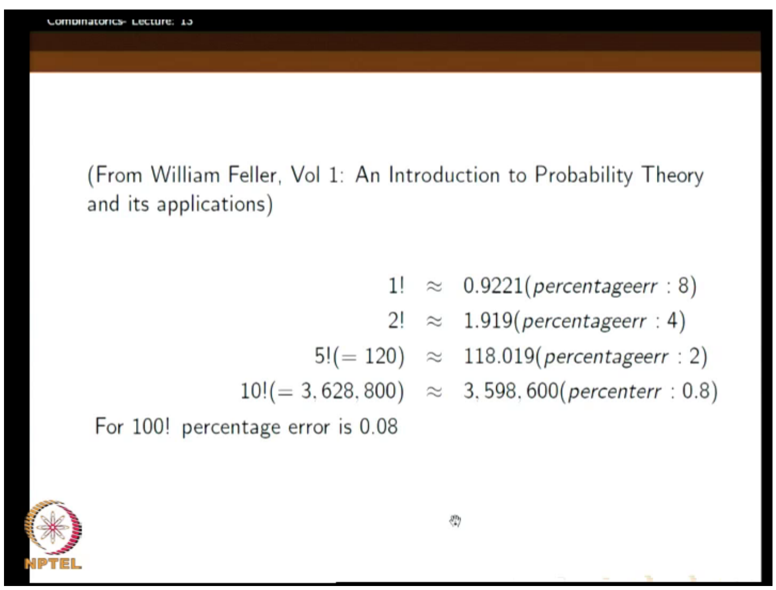


So, this n factorial is approximately equal to n by e whole power n into root of 2 pi n. So, you can also write it n by e whole power n into root of 2 pi n or you can write. So, n factorial is equal to root of 2 pi, this is the constant coming here and we have n to the power n plus half;

that is this n raise to half is coming as n here in the root in the earlier expression, e to the power minus n this is the another one. So, here this e was written n by e, right, n by e of whole power n. So, this half was taken inside the root, right, but here we separate the constant and the dependants on, so n to the power what and e to the power what, right.

But this is only approximately equal to, this is not equal; this is approximately equal to. What we say is this two sides if we consider this RHS and LHS if we consider and the ratio of these things these two quantities n factorial in one side, n factorial divided by this quantity root 2 pi into n to the power n plus half into e to the power minus n. This ratio will tend to 1, right, as n tends to infinity; that is what we say. That means as n becomes larger and larger; the ratio will tend to 1; that means they are almost equal. But on the other hand one should not think that they will become equal actually for some very large n; it will never become equal. In fact if you were considering the difference between the two sides; that means n factorial minus root 2 pi into n raise to this one n raise to may be the we can this formula, right.

(Refer Slide Time: 52:28)



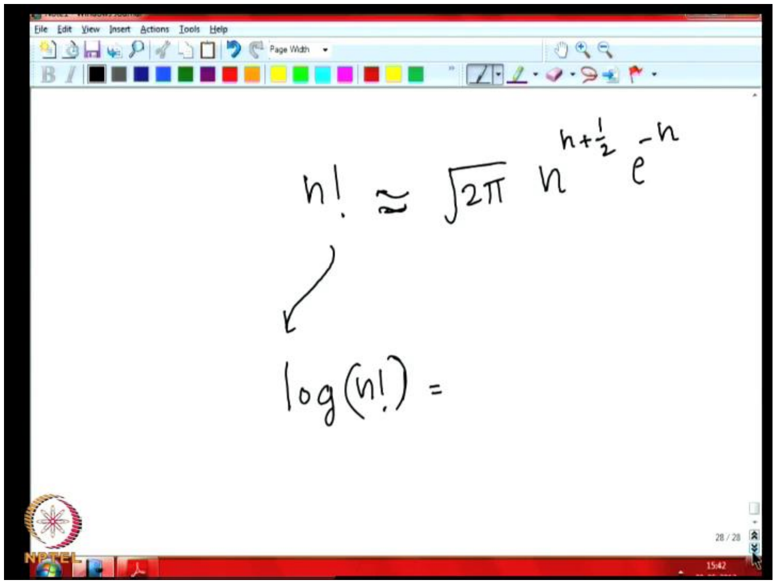
See n factorial minus n by e to the power n into root 2 pi n; that will become arbitrarily large as you go bigger and bigger, n becomes larger and larger. This gap will become bigger than any fixed number; for instance we can say that will it become bigger than one million? Yes, it will become bigger than one million. So, any large number fixed it will go beyond that; it will grow arbitrarily large but the ratio will tend to one is what it is saying. The difference may grow but the ratio will tend to one, right. So, many times it will be quite useful with that kind of approximation also.

You see this data I have taken from a book of William Feller Volume 1. He has given these values, right, for 1 factorial for instance if I evaluate the other side, so for instance I put n equal to 1 and I evaluate the other side; that means root 2 pi into 1 raise to 1 plus half into e raise to minus 1, then it is like 0.9221 approximately. So, here he says the percentage error; percentage error in the sense of this divided by this. For instance you take the difference between the two and divide it by what we are actually calculating; that will be 8 percentage but when you do 2 factorial the percentage error reduces to four 1.919 is the value which we get for this quantity when you put n equal to two to the power but actually we should have got what? 2 factorial is 2 only, right.

So, we are getting 1.919, right. There is a difference but the percentage error has become lesser compared to what it was in the first case but when you go to 5 factorial, value of 5 factorial is 120 but we will be getting 118.019 but something slightly smaller than the difference is slightly smaller than 2. So, if you find 2 by 120 and find that error it will be something like percentage that will something like 2 percentage approximately. So, now we go to 10 factorial so that percentage error is decreasing steadily and when you go to 10 factorial the value is this 36 lakh 28 thousand 8 hundred, right. So, if you evaluate the other side this side we will get something like 35 lakh 98 six hundred.

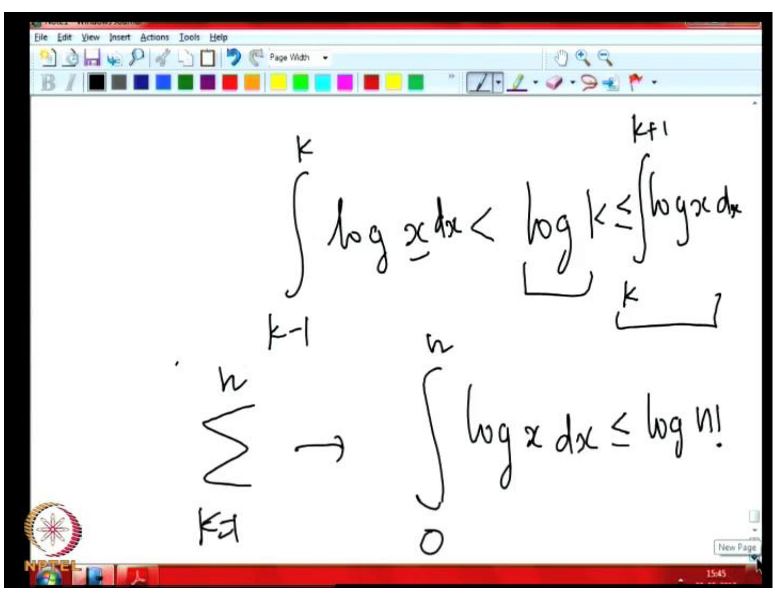
The percentage error has gone down to 0.8 here but on the other hand you see the difference increasing but the ratio wise is it is getting closer and closer to 1. When you go to 100 percentage 100 factorial the error is reducing to 0.08 percentage, right. So, it is better and better is what the claim is and now of course we have this question now how we are going to prove it, right, is it easy to prove. So, usually I do not have to prove it because see it is enough to remember this formula but again may be to give a feel of how the proof goes; may be I will indicate how it is.

(Refer Slide Time: 56:46)



So, for instance we can there are two sides here, right. So, one is, say, n factorial, the other side is our root 2 pi into n raise to n plus half and e raise to minus n, right. Sorry, these are two sides, we were saying approximately this is equal. Now what we do is we can work with say log of n factorial. We can work with log of n factorial and then yeah, with log of n factorial when we work so we can compare the quantity with how will you do this thing. So, we will say that so consider this integral, right.

(Refer Slide Time: 58:01)



So, for instance k minus 1 to k you can try to integrate this functional log x, right. So, k minus 1 to k; of course this log x keeps log of k minus 1, so when you keep increasing the value of k, it will slowly increase, right. So, therefore this is going to be less than equal to log k, right, and this is also less than equal to log. So, for instance for k minus 1 to k log x it will be less than, we do not have to put less than equal to less than log k and that will be less than equal to. For instance we can take it from k to k plus 1, right, this will be correct. So, we have to integrate d x, right; so therefore k minus 1 to k when you integrate log x so that will be definitely less than log k because this is the last k because this quantity is going to be bigger.

Now this can be done for instance when we put k equal to 1, this is going from 0 to 1 log x d x, right, and from here it is from 1 to 2 and so on, right. So, you can sum this thing for k equal to 1 to n, then what will you get? This will go from 0 top 1, 1 to 2, then 2 to 3 and all the way up to n. So, totally this will become an integral from 0 to 1 to n log x d x, right. So, now this will be less than equal to log of n factorial because this is log 1 into log 2 into, right. Of course it is almost time, so we will stop here.