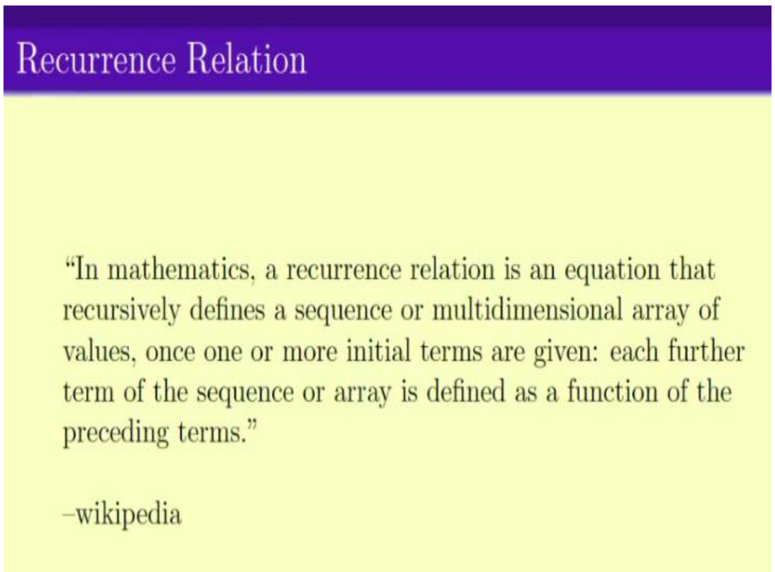
Discrete Mathematics
Prof. Sourav Chakraborty
Department of Mathematics
Indian Institute of Technology – Madras

Lecture No - 41
Asymptotic Notation (Part 2)

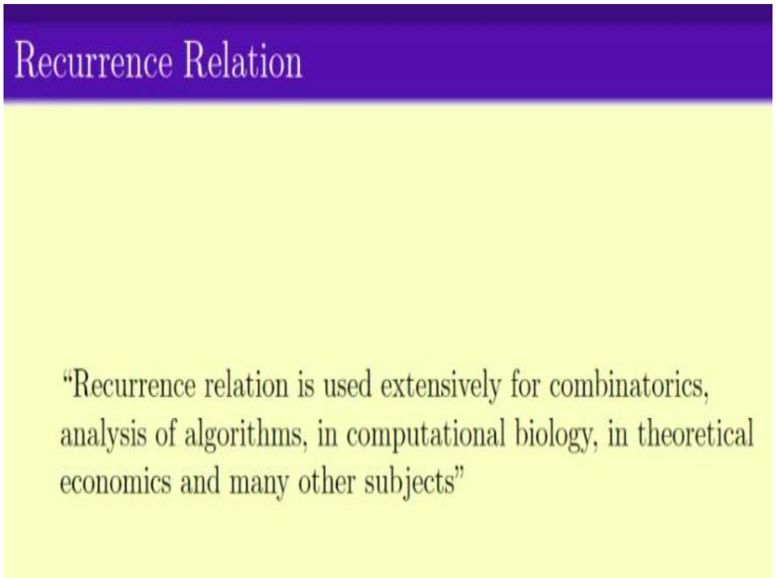
Welcome back.

(Refer Slide Time: 00:05)



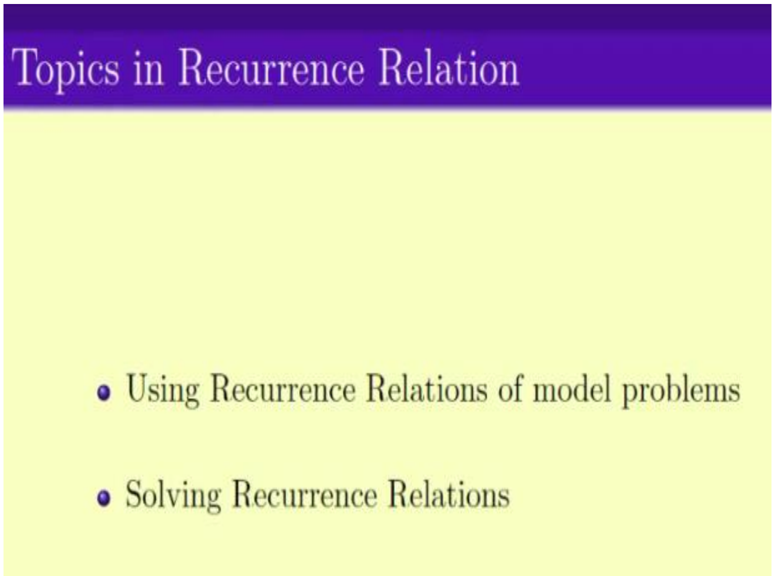
So we have been looking at recurrence relation. So we have seen recurrence relation for some time now. So, it is basically a sequence of numbers that have been defined recursively.

(Refer Slide Time: 00:15)



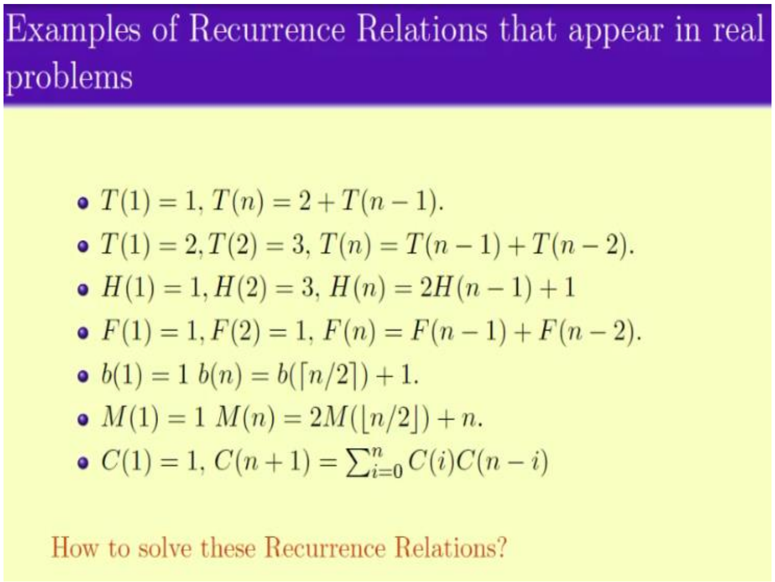
And recurrence relations are useful for various other topics in maths and computer science.

(Refer Slide Time: 00:26)



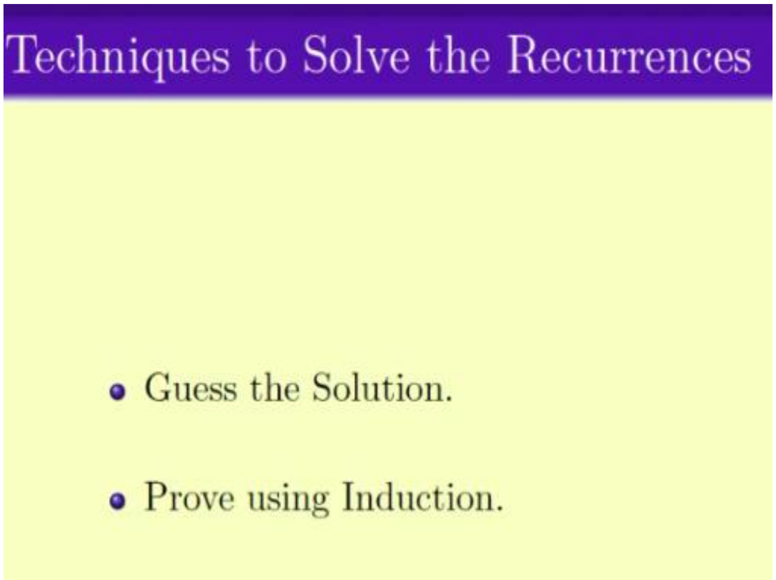
Now, in the topics related to recurrence relations, we have seen that how recurrence relations can be used to model problems and also we have seen how one can use to solve recurrence relations. We have seen some techniques for that.

(Refer Slide Time: 00:48)



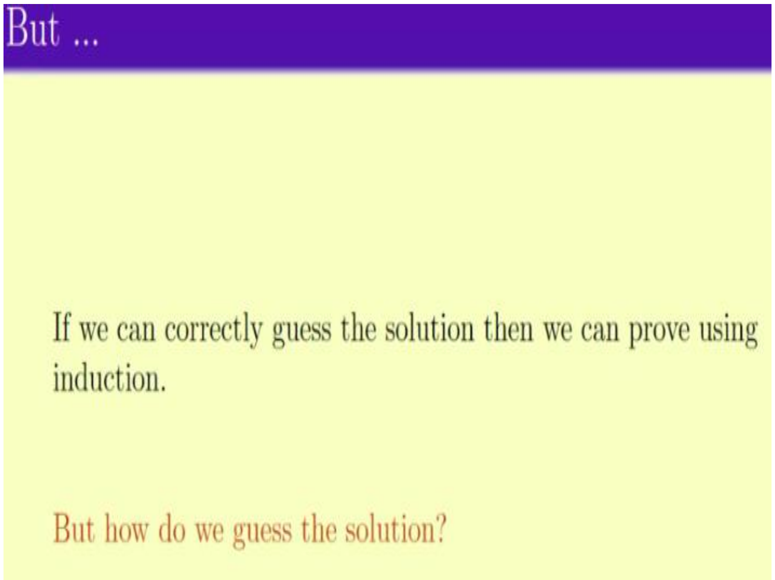
There were few examples that we have looked at already. I do not want to go about it all about again. And the question is, how can we solve this recurrence relation?

(Refer Slide Time: 01:02)



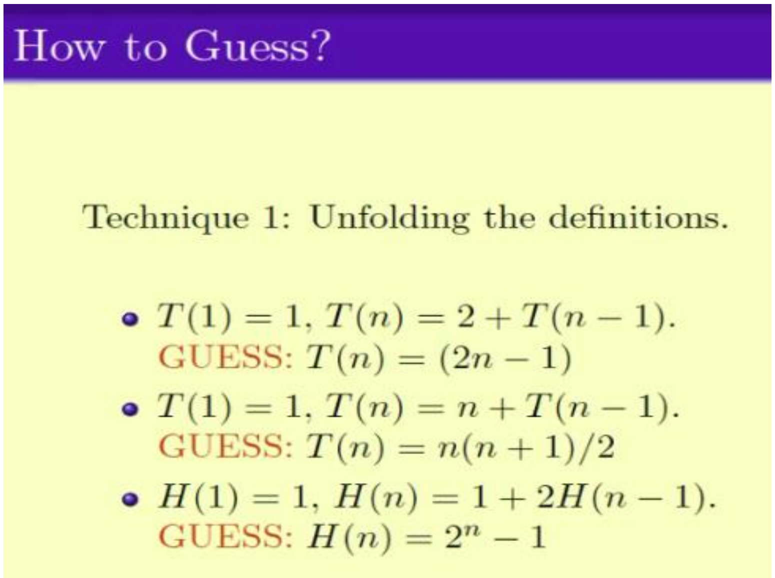
The thing that we saw is that one of the techniques of solving it is, first guess a solution and then prove by induction.

(Refer Slide Time: 01:14)



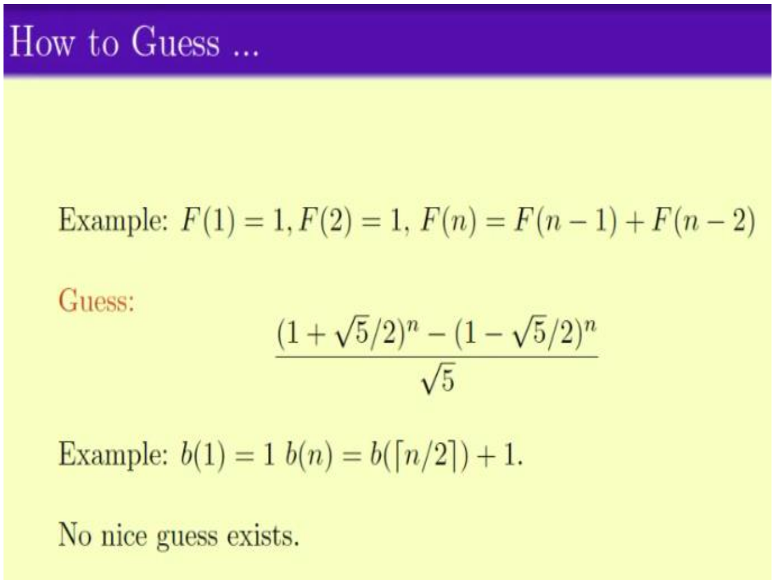
Now the idea is that, if we can correctly guess the solution, then looking by induction, either it is simply straight forward thing. But, how do we get the solution correctly.

(Refer Slide Time: 01:35)



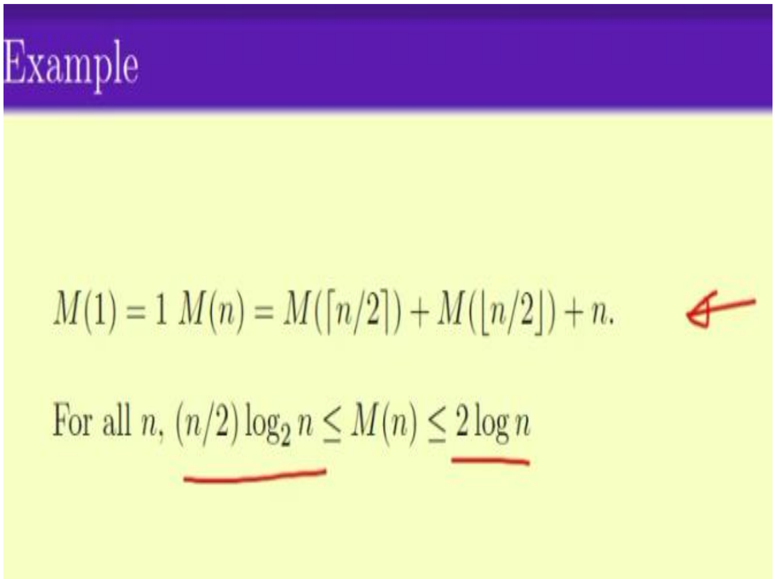
Here also we saw a few examples and we saw that one of the techniques to solve the recurrence relation or get recurrence relation is by unfolding the definition. So, for example we looked at recurrence relation and we say that 'T n' equals to 2n minus one. Similarly, here is another one and we will see how these calculations are solved.

(Refer Slide Time: 02:16)



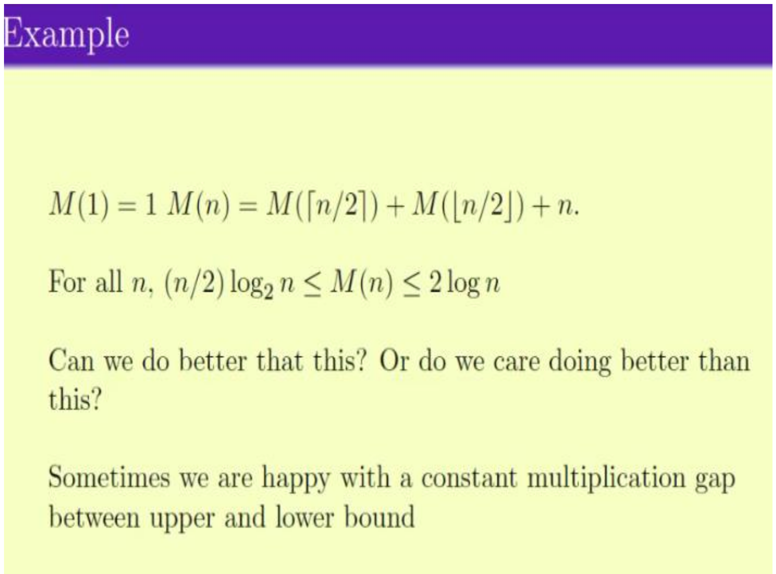
But then, there are some recurrence relations for which we cannot solve it. For example, Fibonacci number. The guessing of this solution is very hard. Similarly, if there were things like the, with floor and ceiling, guessing the recurrence relation is not easy because they does not exist any nice recurrence relation.

(Refer Slide Time: 02:47)



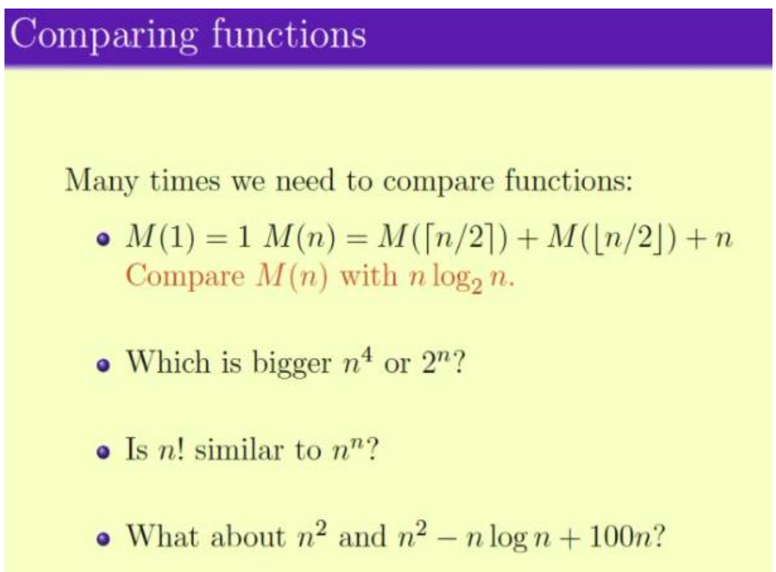
So in the last video lecture, what we saw was that maybe we can come up with some upper and lower bound, the upper and lower bound is something useful. For example, in this particular one, where we are looking at this marked spot, we came up with an upper bound and lower bound for M n, where the difference between upper bound and lower bound is not that much. It is a factor of four.

(Refer Slide Time: 03:17)



Now of course the question is that, can we do better or do even care to do better? So this brings us to the problem of comparing solutions. Okay. So, sometimes we are happy with just a constant multiplication gap between the upper and lower bounds.

(Refer Slide Time: 03:44)

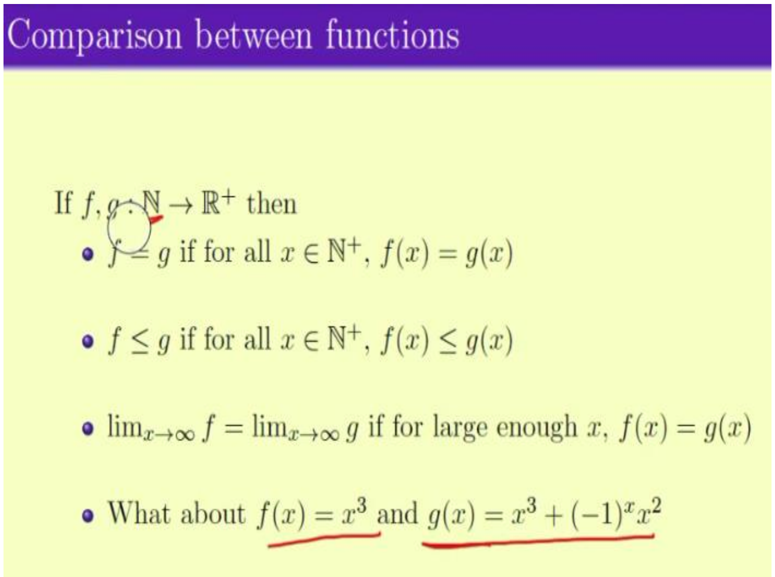


So when we compare functions, how we compare? Now, for example here is one example. So this was the case, where M n plus one and M n plus two is given by the recurrence relation. And question is that can we compare M n with n log n base 2, unless what we got was that it could upper bound M n as two times n log n and lower bound by n over two or half of n log n. That is one kind of compare.

We face with this kind of comparison many times in our various work. Another one is quicker, quick is bigger, n to the power four or two power n? Similarly, is n factorial and n power n similar? Or how about n square and n square minus n log n plus hundred n? So these are the kind of the problem that we face all the time. We have to compare functions. And comparing functions is not necessarily there is a standard way of doing it.

So, till now we have seen a few techniques of comparing functions.

(Refer Slide Time: 05:09)

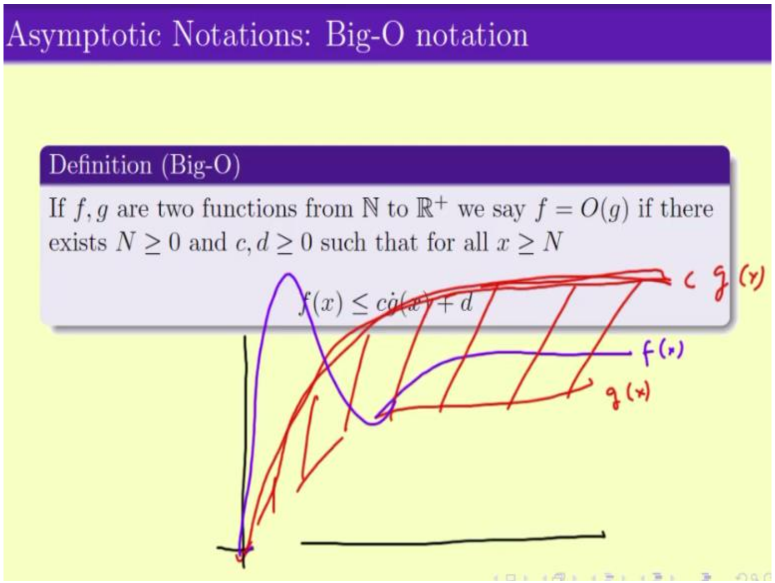


So, if we look at ‘f’ and ‘g’ as two functions from natural numbers to positive real numbers, then the first thing that comes in mind is, we say ‘f’ is equals to ‘g’, if for all ‘x’, ‘f of x equals to f of g’. This is very nice game, want to do it. Second one is what we did for the case of this M n, which is ‘f is less than or equal to g’ if for all ‘x’, ‘f of x’ is less than or equal to ‘g of x’. This can be written and told in other direction that ‘g is bigger than or equal to f’ if ‘g of x is greater than or equal to f of x’.

So what happens if you are not worried about what the functions are behaving in initial stages? So in that case, you should look at the limit of ‘f’ as x goes to infinity. (()) (06:14) the limit of x as it goes to infinity, or the limit of g, as x goes to infinity is said. Or in other words, if for large enough ‘x’, ‘f of x’ equals to ‘g of x’. That is another way of comparing the functions. We say that we do not care about what happens in the initial stages, but as they (()) (06:42) become same.

The other one is what about this kind of stuff. If ‘f of x’ is equal to x cube, and ‘g of x’ equals to x cube + (- 1) quadrate and x square. So, we can see here that since ‘x’ values over all the natural numbers, so that means that for some x, ‘g of x’ is less than power, some ‘x’ ‘g of x’ is more than of ‘f of x’. And the limit, they are not the same. But again, are they kind of similar? No. To formally put this one, we have to come with a formal set of definitions. And this is what we call as asymptotic notations.

(Refer Slide Time: 07:36)

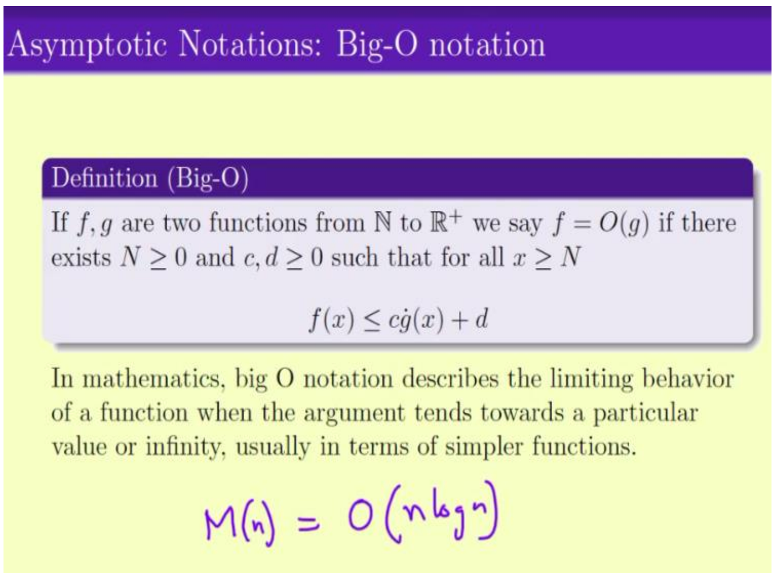


This is the first of the big asymptotic notations. And this is known as Big-O notation. So, if ‘f’ and ‘g’ are the two functions and if we say f is ‘O of g’, if there exists an N greater than or equal to zero and some constant ‘c’ and ‘d’ such that for all ‘x’ greater than or equal to N, ‘f of x’ is less than C times g of x plus d. So, it however what we are saying is that, basically for Large N of x, ‘f of x’ is upper bounded by some constant factor of ‘g’.

If you draw the graph here and say this is ‘g’ and this is ‘f, now ‘f’ can be, d can be small - but if you now consider this is ‘g of x’ and this is ‘f’ of x’ and if you think of some constant multiple, so this is the constant multiple, so, c times ‘g of x’ as long as ‘f of x’ is below this red line, this line, we say ‘f of x’ is bigger than ‘g of x’.

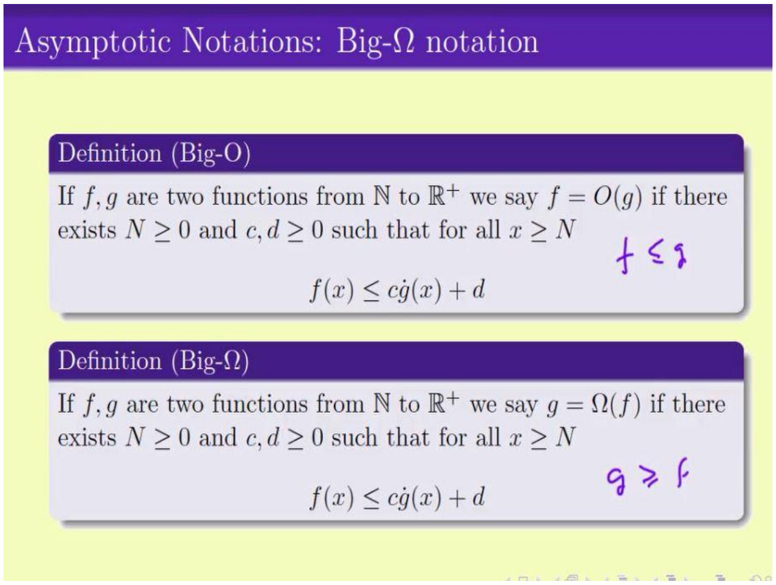
In fact, you do not care about what happens in the initial stage, the blue line can actually be like this and that is also good enough for us. So, we do not care some constant multiples of each other. But, f of x is bounded by some constants of g of x, for large N of ‘x’. And this is what we call as Big-O. We will say ‘f’ is equal to Big-O of g. That means ‘f is smaller than some constant multiple of g’. You should get familiar with this particular notation because this is supposed to be used all the time.

(Refer Slide Time: 10:13)



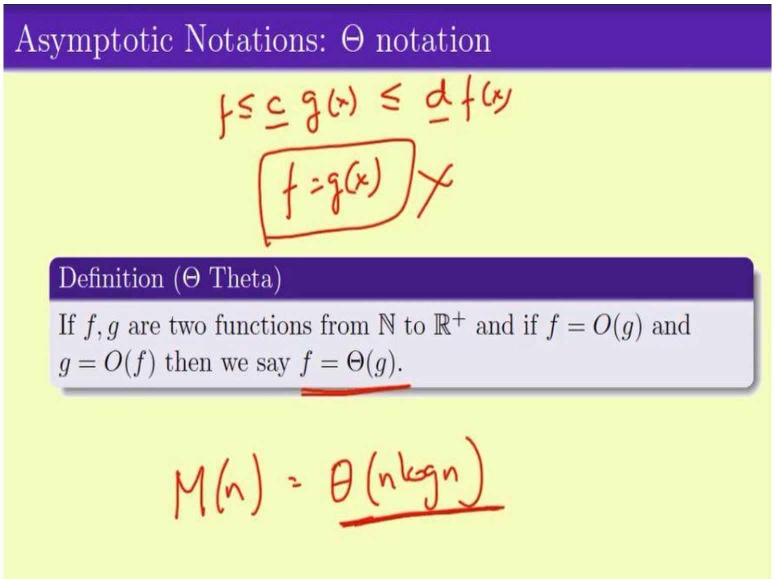
In fact, in mathematics, Big O describes the limiting behavior of a function where arguments tend towards a particular value in a simple term. So in fact, keeping this in mind, what we can say is that the M (n) that we were doing it already is a Big O of n log n. Because we have seen that M (n) is less than twice n log n for n greater than 5. So this is what we have.

(Refer Slide Time: 10:56)



Now just like Big O, we have the opposite like whatever in this relation with f is Big O of g, the other way of activity is g is Big Omega of f. So, this is the same thing just like once the case is less than g, so this is a thing of as f less than g and this is g is greater than f. Both of them are same term. In one case, we use Big O, and in one case, we use Big Omega.

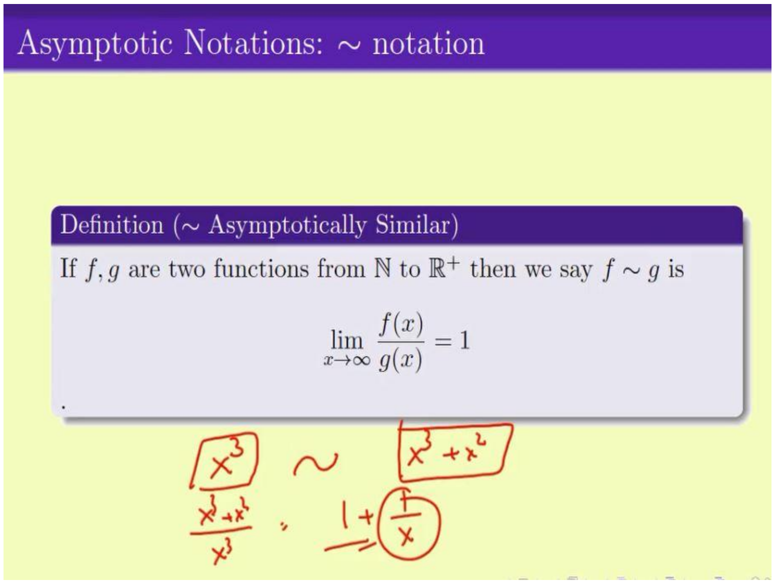
(Refer Slide Time: 11:30)



Now what happens if both of them are true that means if f is Big O of g and g is Big O of f? So in other words, in this term, f is something like less than constant some g which is less than some d times f. Or in other words, f is kind of sandwich between two constants of g and g is sandwich between two constants of x, then basically I would like to say something like f is equal to g, which is not true, in general because we have this constant, so instead of that we have f is Theta of g.

So in our notation, we have M (n), therefore it equals to Theta of n log n. So whenever if somebody says that some function is Theta of some object, it means that it is kind of it is upper bounded and lower bounded by a constant of this terms. And this is what we kind of help us to mathematically write down, what we mean by asymptotically this function is equal to this. Now there is two more, or actually three more things.

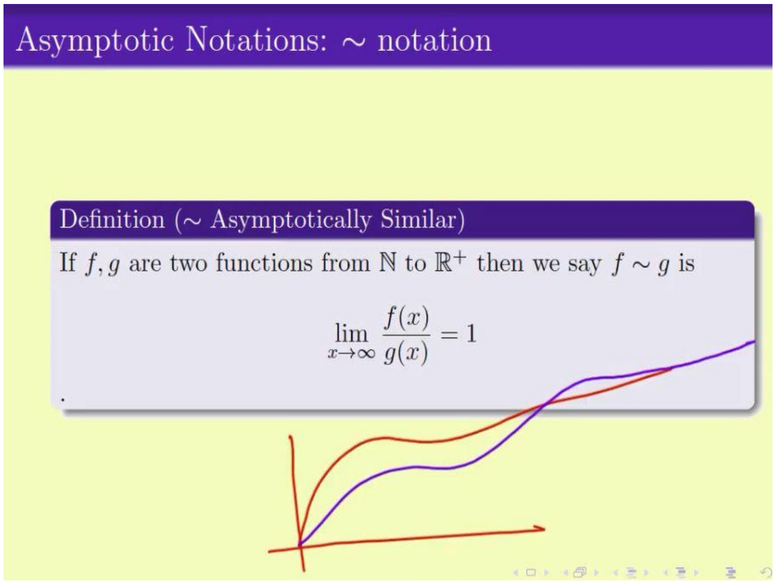
(Refer Slide Time: 13:00)



One thing much more stronger than this particular Theta is what is known as the sim notation and this called asymptotically similar. So f of x, g of x of two numbers are similar if limit x tends to infinity f of x / g of x is 1. Now, this is something useful, for example we looked at x cube and x cube + x square. Surprisingly enough, so these two are of course not equal, but they are similar. Why?

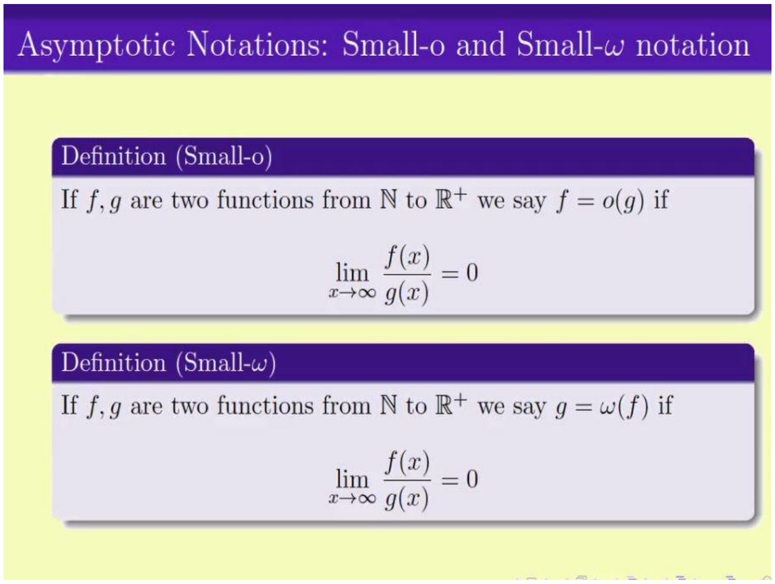
Let us look at this one this by x cube + x square / x cube is basically 1 + 1 / x and x goes to infinity this becomes 0 and get 1. So, in other words, here we say that this and this are similar.

(Refer Slide Time: 14:10)



The idea is that here if you draw the plot of f of x and g of x may be they are something, they are slightly they will be different for f and g. But, as and as x goes to infinity these two converse to each other.

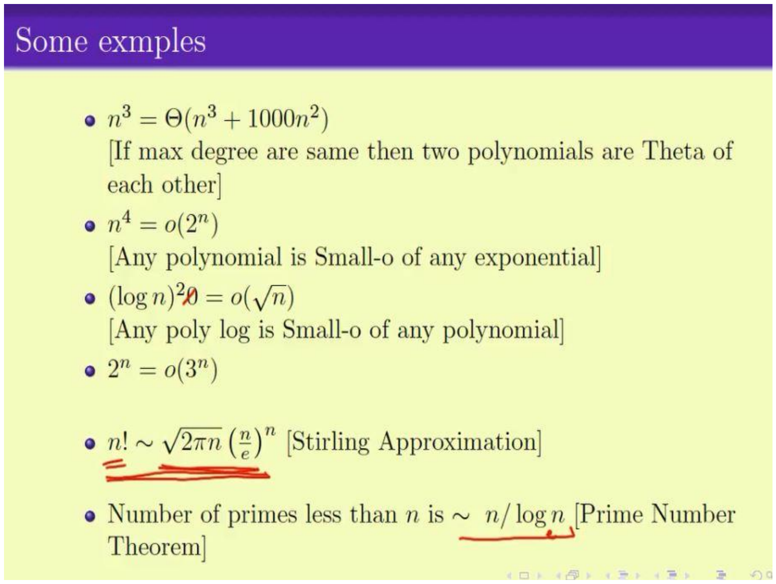
(Refer Slide Time: 14:26)



So this is the (()) (14:27) most the strongest that we can see, strongest relation between two functions other than becoming exactly equal to each other. So, we have Big O, we have Big Omega, we have Theta and we have (()) (14:50). There are two more things that we have first of all this Small o, Small o is basically same f (x) / g (x) goes to 0. If f (x) is significantly smaller than g of x, significantly smaller and third one is just opposite of this, which is same if f x is significantly smaller than g of x, we say g is Small omega of f.

Have you had to get familiar with these various notations, the idea is the simple, they help us to kind of get a handle on various functions help us to compare across functions which is bigger, which is smaller and it is useful if we marked it for simplifying our mathematics.

(Refer Slide Time: 15:50)



So, I will give you some examples and I will ask you guys to solve them or prove them for yourselves. First of all, n cube is Theta of n cube + 1000 n square. In fact, when you have two polynomials and if their maximum degrees are same, then they are Theta of each other. If (()) (16:16) more if the maximum, the coefficient of maximum degree is same then we have sim. Okay, so C n cube is sim of c n cube + d n square + (()) (16:33). So when you are looking at polynomials then looking at the maximum degree is what matters.

I will again ask you guys to prove them for yourselves. To prove and convince yourselves that it is true. So if somebody tells, somebody says that the running time of some function is n cube + 1000 n square, you can say it is Theta of n square. Similarly, n to the power 4 is Small o of 2 power of n, whereas n goes to infinity n to the power 4 is very, very, very smaller than 2 power n and in fact, any polynomial is small o of any exponential, to prove that.

So n to the power 1000 is Small o of 2 power square root n. Similarly, log n square is equals to Small o of square root n. Again any poly log is Small o of any polynomial. Now these are the kind of very useful things to keep in mind, when you are looking at this state of Small o, Big O, Theta notations. But then there are also other, couple of other formulas for which sim is very useful. And let me give you them, sorry there is one more here, namely 2 power n is Small o of 3 power n.

Now regarding sim, here are two very useful things. First of all, the n factorial. n factorial is a very weird of it. We do not know what we know, what it means is n into n minus 1 into n minus 2 and so on. But how does it compared with other functions? And we have what is known as Stirling's Approximation namely n factorial is asymptotically similar to square root 2 Pi n times n / e power n. So in other words, as n goes to infinity then n factorial basically behaves like this.

And this is very useful for us. So this helps us to control or understand or compare between functions. There is another one which is known as the prime number theorem. So, the number of primes less than n is asymptotically similar to n / log n. So, as n goes to infinity, number of primes that is less than n is equals to n / log n. So this is actually log n base. So, this is kind of language in mathematics that we have developed which helps us compare against or compare between functions, when the functions are not exactly same or they do not have a very clean representation.

To help us to compare between functions we use this set of notations. These notations come again and again and again to simplify our mathematics. In the next video, we will see how we can use these particular notations to solve some of the reoccurrence relations. Thank you.