**NPTEL**

**NPTEL ONLINE CERTIFICATION COURSE**

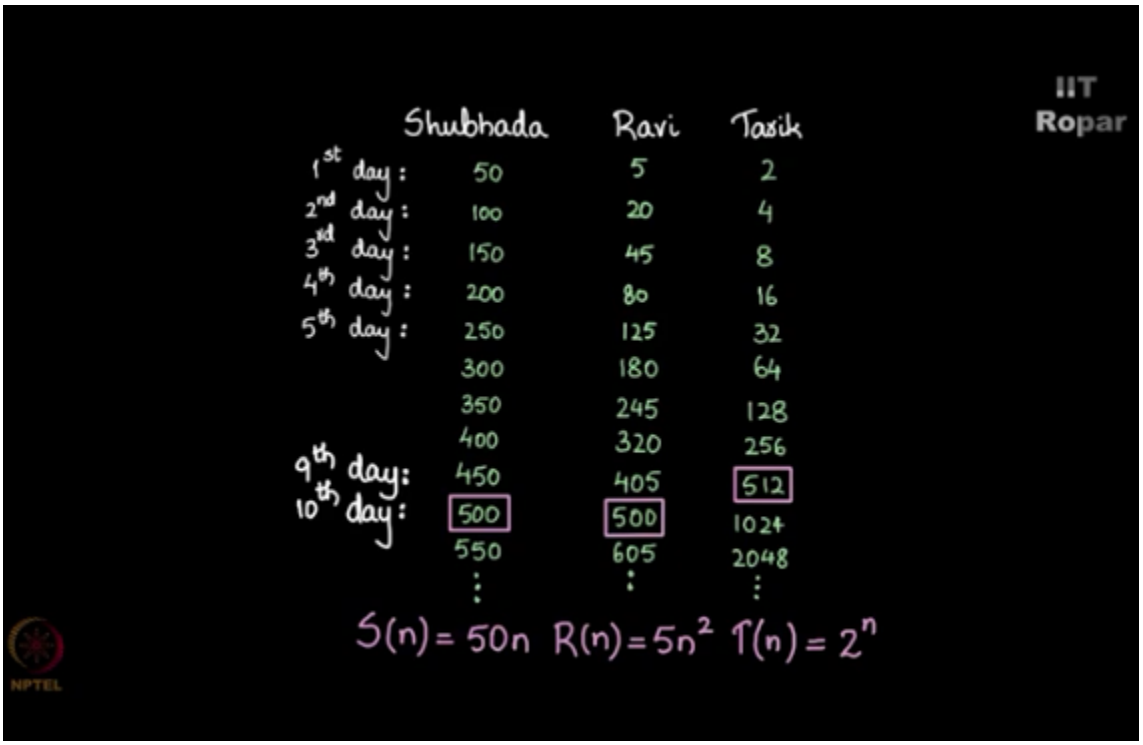
**Discrete Mathematics**
**Recurrence Relation**

**Intuition for 'complexity'**

**By**
**Prof. S.R.S Iyengar**
**Department of Computer Science**
**IIT Ropar**

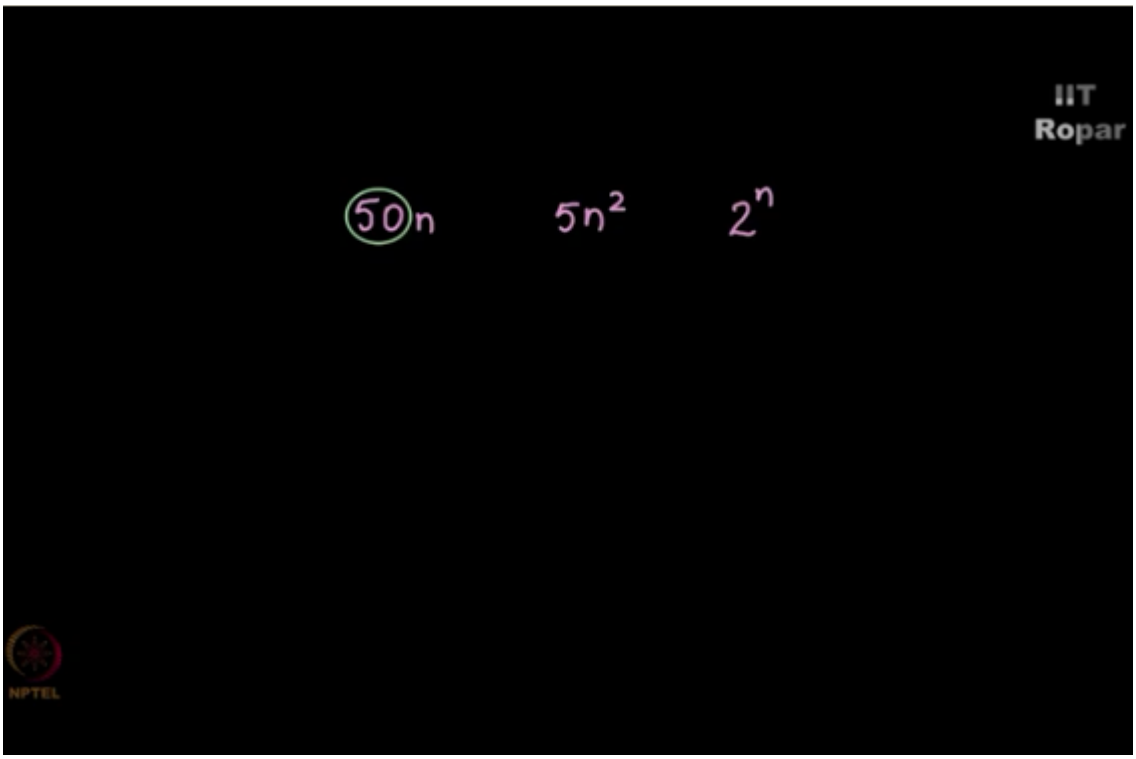
Imagine the situation, Shubhada, Ravi and Tarik, they're earning daily wages let's say, on the first day Shubhada earns 50 rupees, second day she earns 100 rupees, third day she earns 150 rupees, and fourth is 200, 250, 300, 350, 400 and so on, I would capture this function as S(n) = 50n on the nth day Shubhada earns 50n, although she starts very less, let's say 50 rupees per day on the $10^{th}$ day she starts earning 500 rupees per day, that's how Shubhada earns, S for Shubhada, S(n) is 50n, and Ravi earns based on this formula 5n square, first day it is 5 x 1 square 5, second day it is 5 x 2 square that is 20, and then 45, 80, 125, R(n) is 5n square, you see you can compare S with R, Shubhada with Ravi you will observe that Ravi's earning less than Shubhada, but if you observe on the $10^{th}$ both of them get 500 and after that forever Ravi earns more than Shubhada, correct, point to note how you start is not important, it's all about how you grow better with time, correct, Shubhada appear to be richer than Ravi to begin with but very soon Ravi took over.

Look at a third column here, Tarik, his function is simply 2 to the N, first day he gets 2 rupees, second day he gets 4, $3^{rd}$ $8^{th}$, $16^{th}$ 32, 64, 128, 256, 512, now do you see in the $9^{th}$ Tarik gets when N = 9, $9^{th}$ day Tarik gets more than Shubhada and Ravi, but he was faring much worse than Shubhada and Ravi to begin with, and after this Tariq is always having an upper hand over Shubhada and Ravi because his returns are going to be very high,
(Refer Slide Time: 02:33)



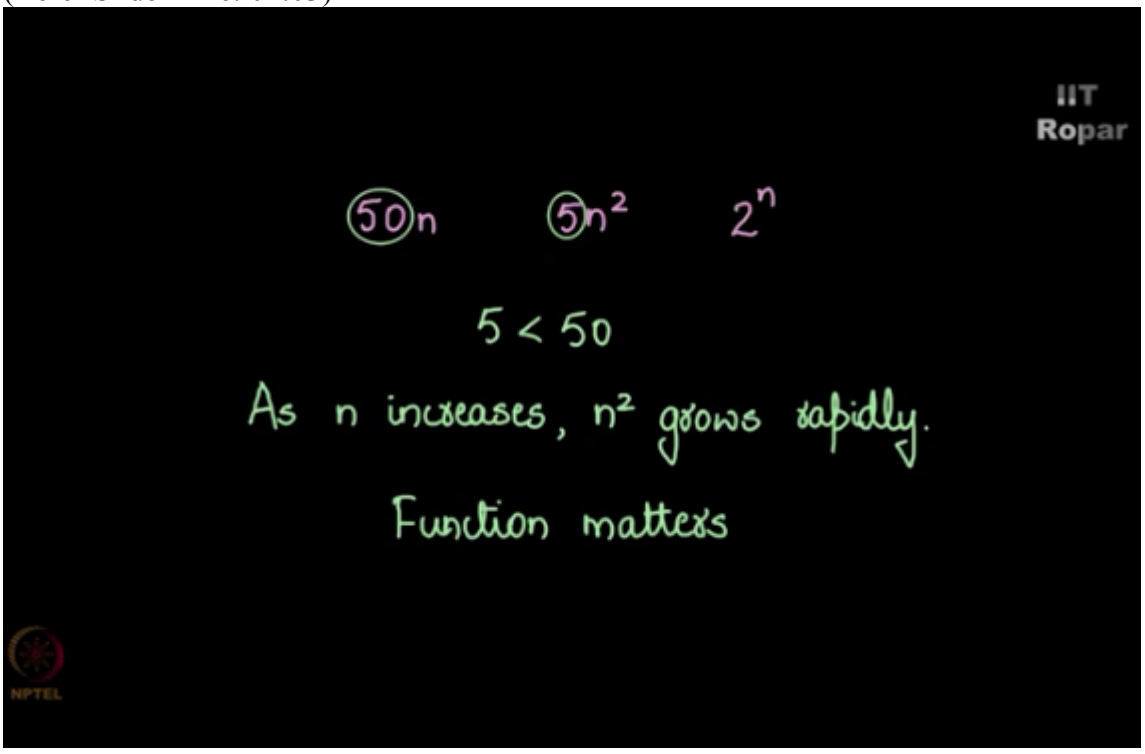

so this is like kangaroo and the cat example, they might appear to be very small when they are born but the growth of these two animals happen very differently, kangaroo appears to be a tiny biscuit, while a kitten looks like a puppy, but then a kitten grows very slowly and reaches 45 kgs well a kangaroo doubles the size and reaches 90 to 100 kgs right, so we cannot say anything about a function until we analyze how it grows, point to observe 50n, 5n square, 2 to the N.

Look at 50n here the reason why Shubhada appear to be rich in the beginning is because of the number 50 there, but then eventually what adds to this 50n is the N itself, the number 50 will not, it's a constant it will not have much to contribute why?
(Refer Slide Time: 03:36)



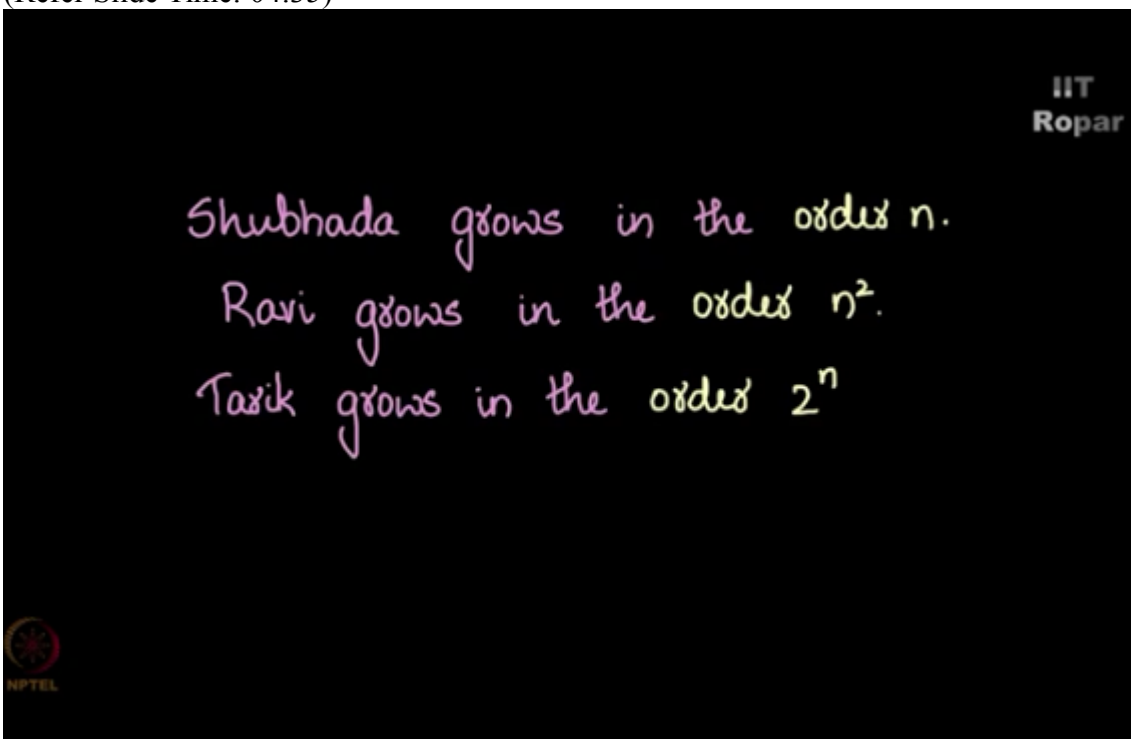

5n square, 5 was small that is why Ravi R(n) you observe that he was increasing slowly, because the constant was smaller 5 is smaller than 50 but then as N increases N square grows very big in comparison to N, and that is why Ravi had an upper hand, so it is not important what is a constant that is sitting in front of a function, what matters is the function itself,

(Refer Slide Time: 04:03)



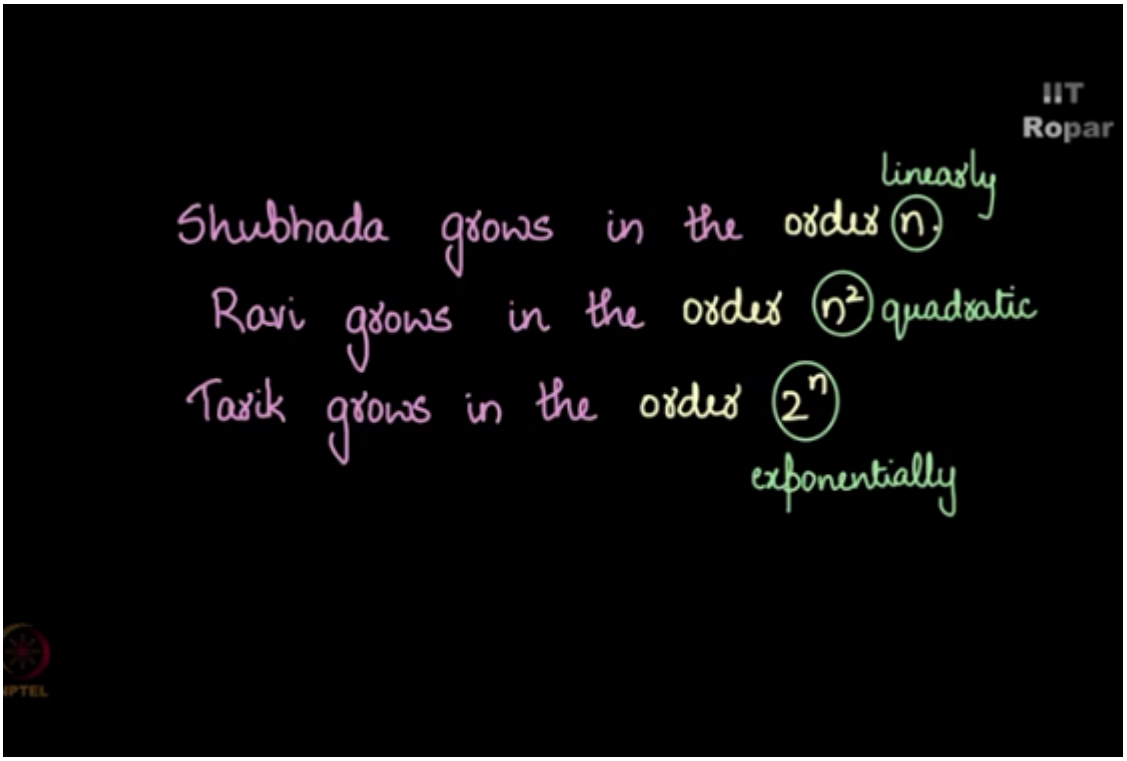
so what we'll do is we will use this language, concentrate carefully, we'll use this language, we will say Shubhada grows in the order of N, Ravi grows in the order N square, what did I do? I'm just dropping the constants because I realize that constants are not important here, and Tarik grows in the order of 2 to the N, correct,

(Refer Slide Time: 04:35)



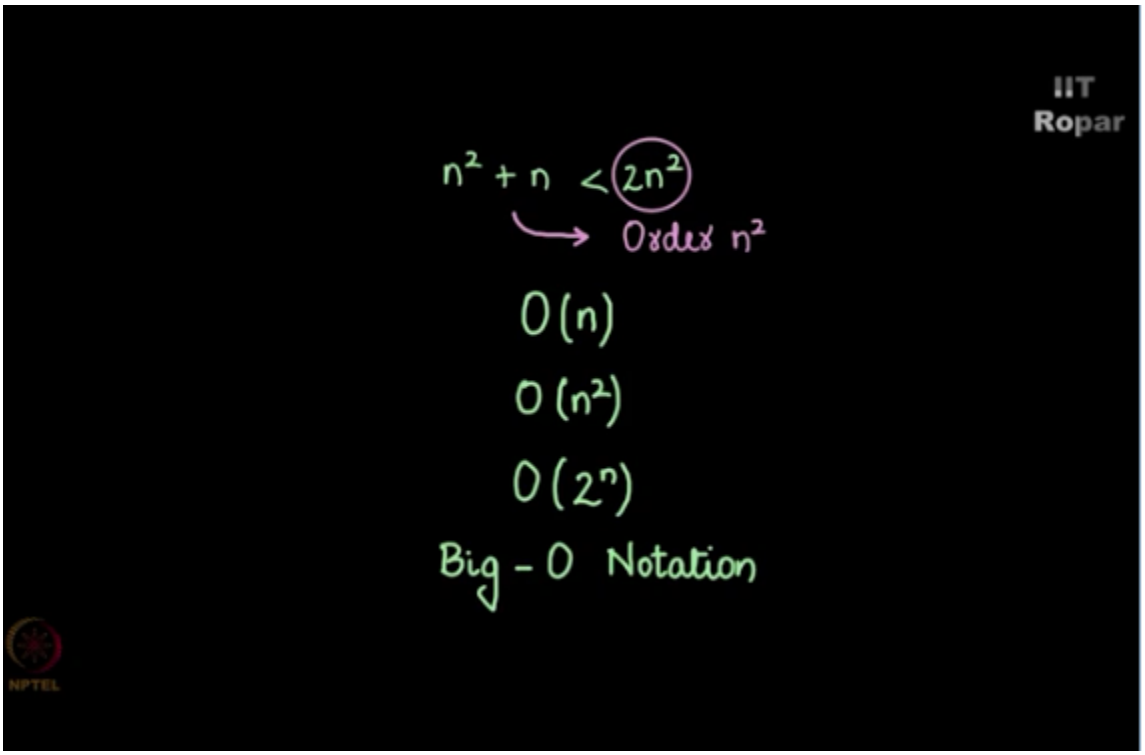

so look at Shubhada, don't look at 50 there Shubhada gross we say linearly, order N, N is linear you see, N square we call it quadratic, Ravi grows in a quadratic way, and Tarik 2 to the N is called an exponential function,

(Refer Slide Time: 04:56)



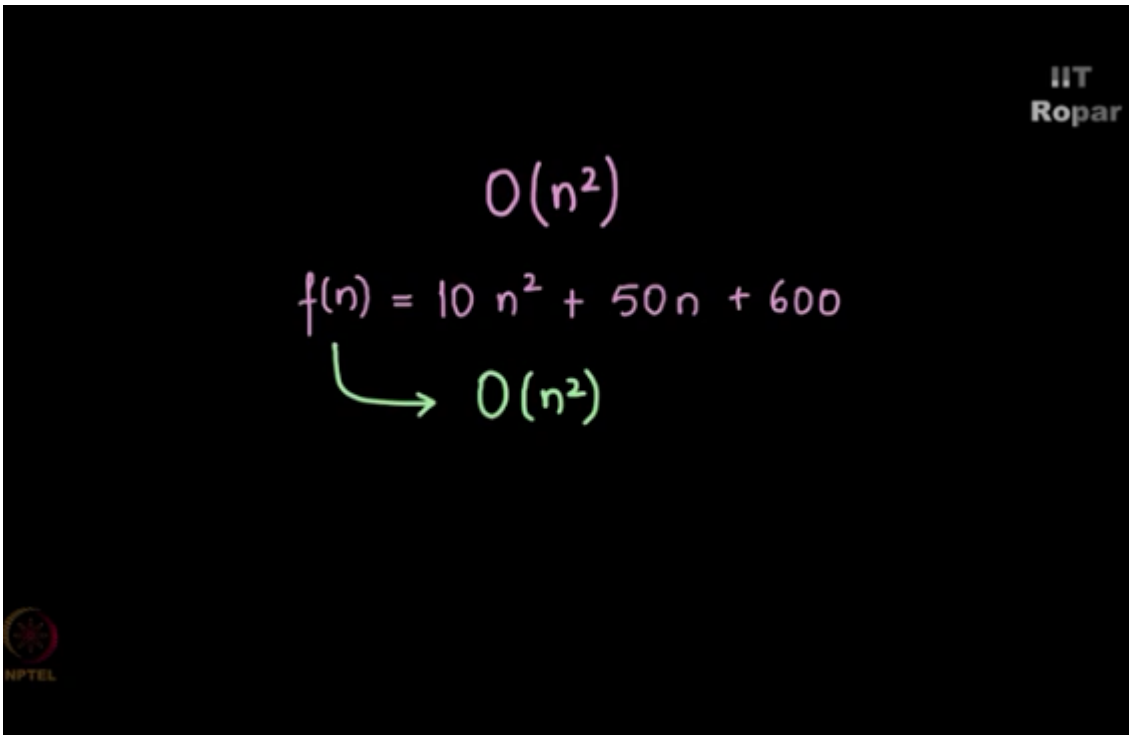

Tarik grows exponentially, exponentially means very quick things get doubled you see although appearing is slow in the beginning because a constant that sitting in front of Tarik which is 1, 5 in front of Ravi, 15 front of Shubhada, but then Tarik grows exponentially.

So order of Shubhada is N, order of Ravi is N square, order of Tariq is 2 to the N, underline this get this straight what matters the most is the function itself and not the constants, also note something if there was something like N square + M this is less than or equal to you see I can easily write N square + N less than N square + N square which is 2N square, but then 2N square is of order N square, so whenever you have N square + N you can consider that as order N square as well, right, so we are going to represent such a notation with a big O(n), big O(n square), big O(2 to the N), a word of warning, many different books have different ways of introducing what is called the order complexity, this is called the big O notation,
(Refer Slide Time: 06:19)




but we choose to only give you an intuition because it's a discrete math course and eventually when you study data structures algorithms in computer science this will be dealt very deeply, but then you will have the maturity to understand it there when it gets more mathematical, as of now let us only work at the level of intuition, order N square means my algorithm is growing the N square style, I am going to drop all the constants in front of N square and if it's a polynomial like 10n square + 50n + 600 if that is let's say F(n) I will simply call F(n) as order N square, big O(n square) right, the motivation is the kangaroo cat, the motivation is how Shubhada, Ravi and Tarik's money was growing day by day.

(Refer Slide Time: 07:11)




**IIT MADRAS PRODUCTION**

**Founded by**
**Department of Higher Education**
**Ministry of Human Resources Development**
**Government of India**

**www.nptel.iitm.ac.in**