waiting for some number of times so that we do n't know we are sure that it still not execute for n. 4 goes on how many times it "s going to execute it 's going on K times so what this will be 1 plus 2 Plus 3 plus 4 plus goes on to K time. Tracy: In the next video we 'll find more examples right so different type of course I will show you and I 'll show you how to analyze them so this is all by tracing you canonized n so in the other video next video also you can find few more algorithms that

1.5.2 Time Complexity Example #2

This is the time complexity so the statement will execute for order of log n base 2 times so from this we can observe that if you have a loop where the counter variable it 's not implementing but it is getting multiplied by something let us say 2 or 3 then it will get log base 2 time so you'll execute form operates two times. K that will be equal to n so K is how much log in these two so the other we have analyzed this one. If you have any look like this the value of I is getting multiplied then it is going to take log and time. If a rent of n value at 10 let us see what happens is initially 1 1 is less than 10 now n value is at 10 once less then 10 continued its multiplied by 2 so 2 is left then four four is less then eight multiplied with it is 8 8 is also connected then 16. If I find log eight log interval between how long eight is written as two for three days. The previous one in the first example I have shown you that it was multiplying time and starting for a month now this is dividing every time starting from N and and up - what up - 1 it is reaching so it 's part of login makes these fullest. The time complexity for this one is this is independent rule and it is situated for n times.
Order of n is order of n and whatever is they rookie will repeat for and times next the in the follow this is using j j j is less than n and j is every time x 2 right so this is a friend a login time and anything inside that also will take login time it remember initially I've shown you that this is mistakes n plus 1. Our loop is incrementing that is n or decrementing it is and whether it is in demanding by one hour two or even ten or even hundred also it 's sort of Emily 's same weight and my rules also order of N and by 200 is also order.

1.5.3 Time Complexity of While and if #3

In C language there are three loops while and do-while there is a difference though while will execute minimum one time but follow fine while loop. Before C language he has languages used to provide for loop in a different way let us see that follow for I assign one to M do some statements inside and this is : here this was the syntax in Pascal language. in old languages they used to be a loop called repeat some statement inside and under some condition now these loops were different repeat until loops are different this will repeat as long as a condition is false and once the condition is true it will stop so it is similar to do while how minimum one time the statement is but it is different compared to Dubai. for n time total how many 3 n plus 2 but when I was using for loop I was saying that let us ignore these two and just take it as n plus 1 so I was. saying that it was 2 n. plus 1 whatever the function may be we are not interested in exact formula or exact function we are. interested in the degree of a function so we say order of n now that 's it now you can see whatever I can write using while loop I can. write it using for loops also now our next piece of code here see