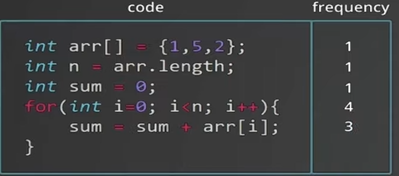
Time Complexity

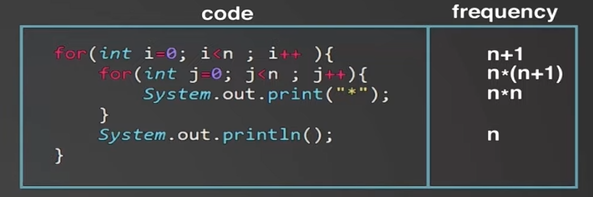
1. For loop executes one time more than the body because it checks the termination point which is last part.

for loop - (n+1) times

loop body - n times



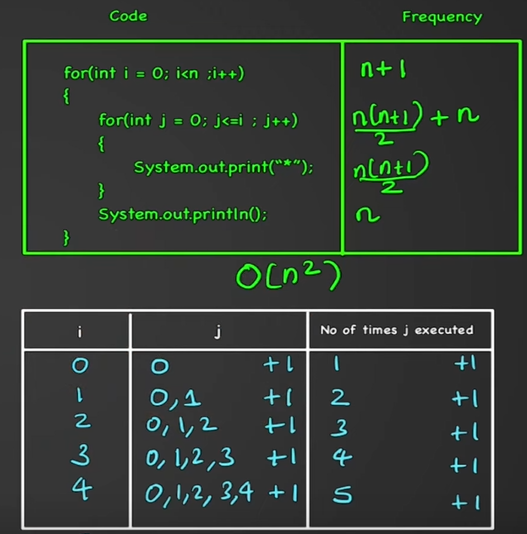
1. Big O notation will tell how much time our code will take.
2. 2n+4 for the above code snippet
3. Rule for writing the time complexity
4. Remove the constants
5. Retain the highest order term (n)
6. So it can be written as O(n)
7. Here time increases linearly
8. Program for nested for loop



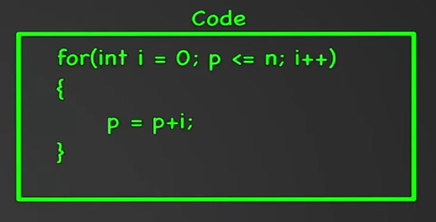
Here after adding all terms and follow the above rule we get n2 as the highest order term.

So it’s time complexity is O(n2)

1. Inner loop taking outer loop parameter for it’s break condition



1. Time complexity of code which depends on input



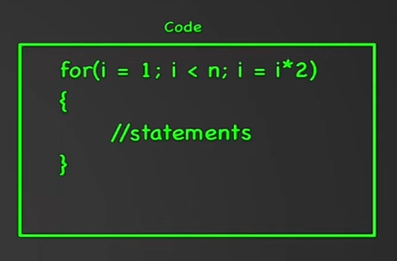
Here n is input value

Here Time complexity is O(wps)

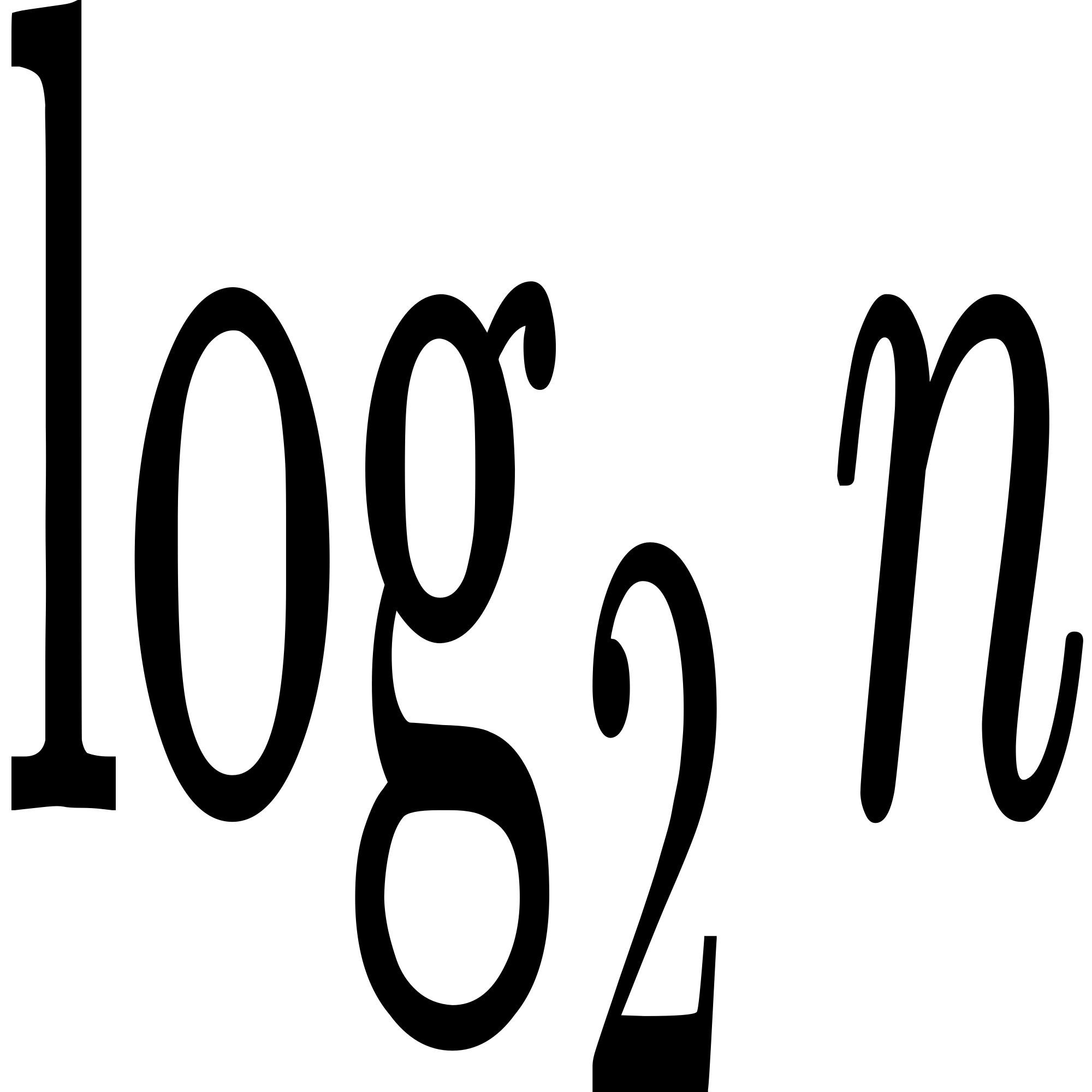
1. Efficiency of code with Time complexity

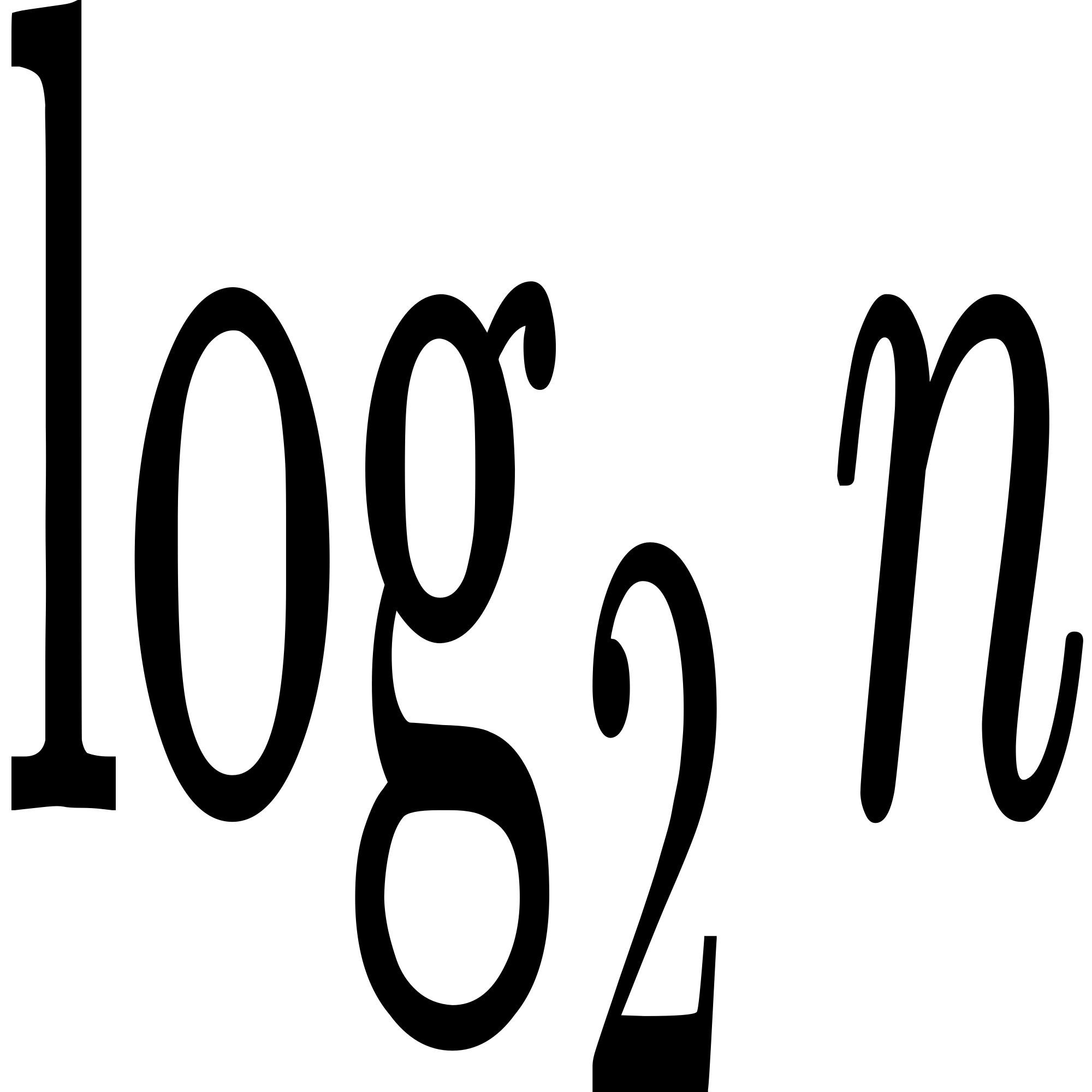
n3 > n2 > n > wps

1. Another Condition

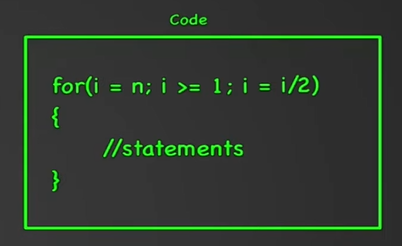


Here i is increasing with i2 so it’s Time complexity is O(wps)

or we can say that it’s Time complexity is O()

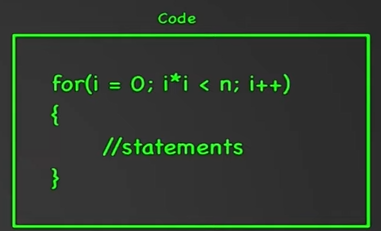
O() is generalized for all terms like i3 .

1. Another Conditions



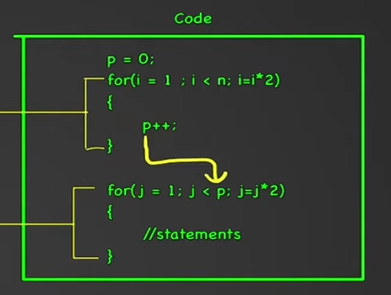
Here Time complexity is O(wps)

1. Middle Conditions



Time complexity is O(wps)

1. Time complexity for two independent loops is O(n)
2. Time complexity for below condition is O(wps)



1. Time complexity for another condition is O(wps)

