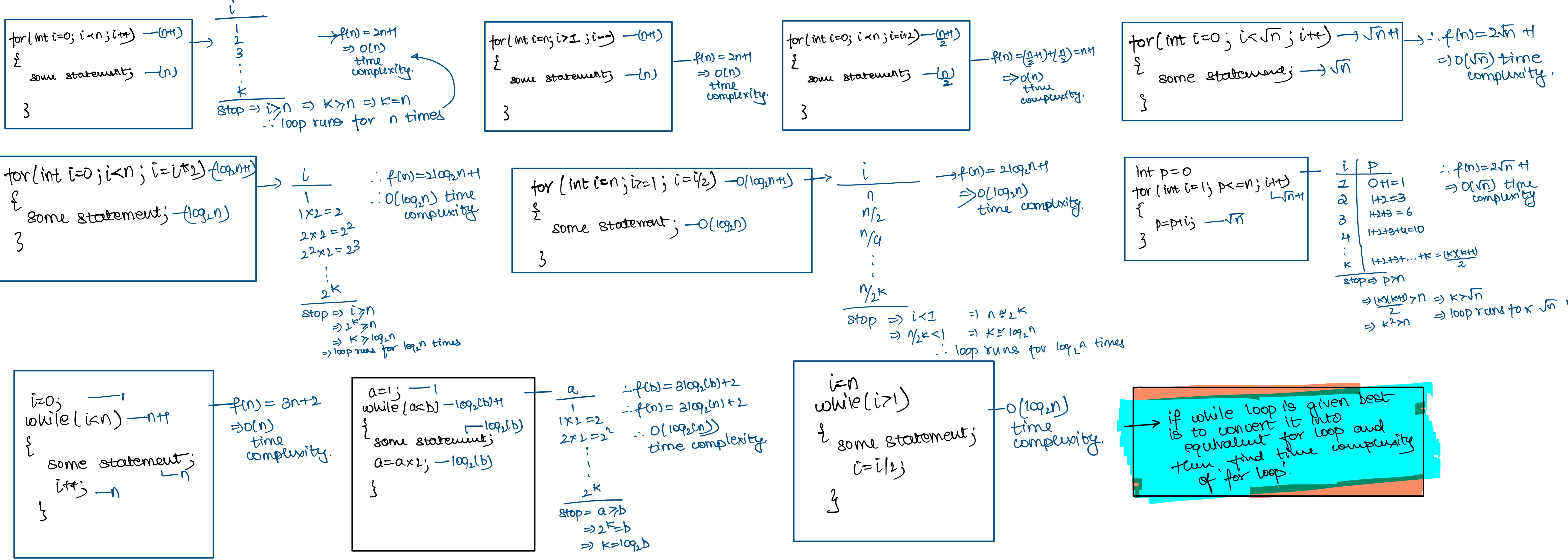
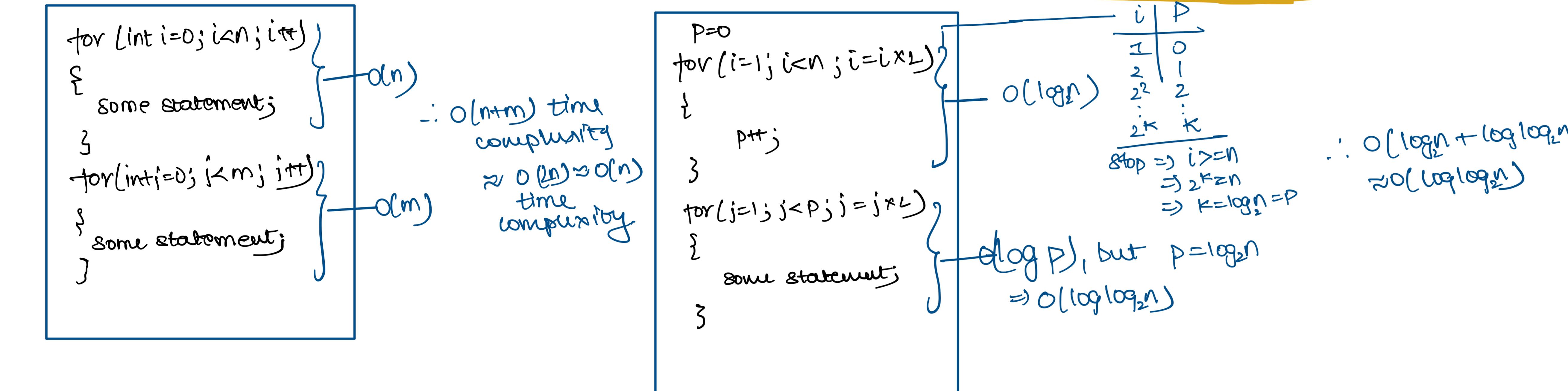


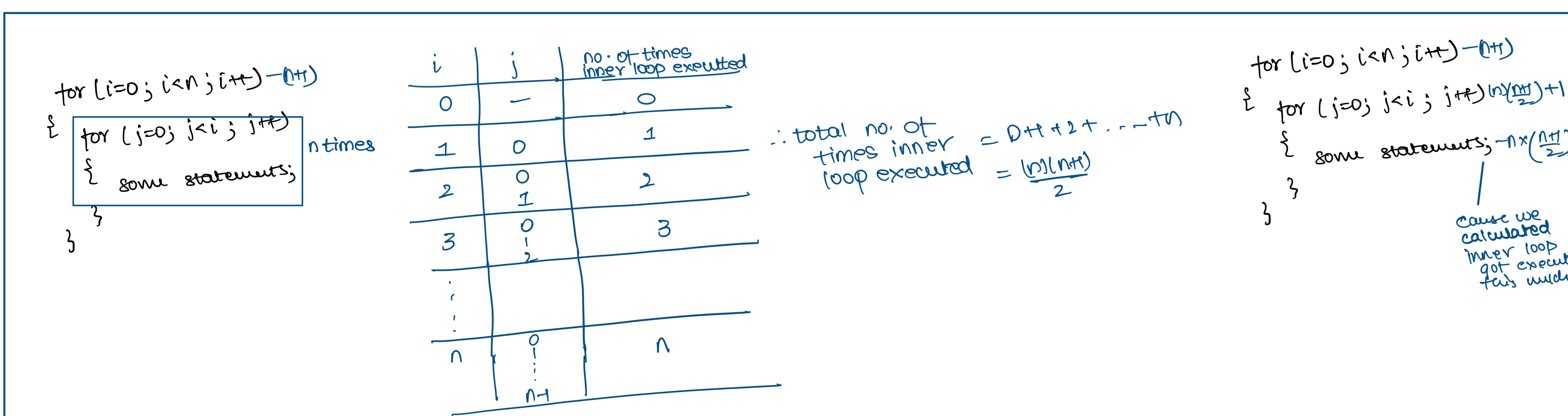
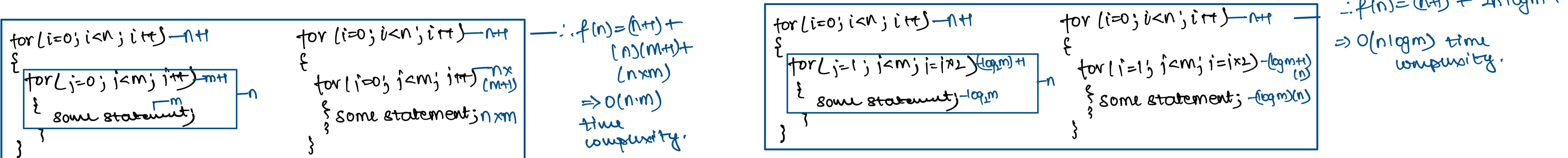
### single loops



### Two Unnested loops



### Two nested loops



$$O(1) < O(\log n) < O(\sqrt{n}) < O(n) < O(n \log n) < O(n^2) < O(n^3) < O(n^4) < O(2^n) < O(2^n \times n) < O(n!) = O(n^6) < \dots < 3^n < 4^n < \dots < n^n$$

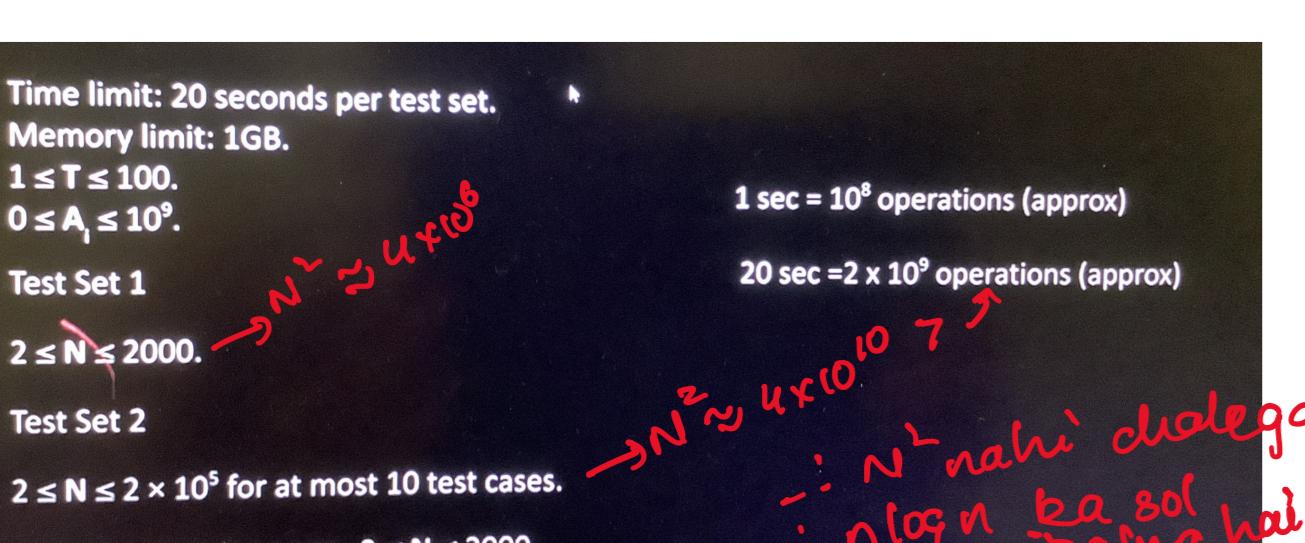
To decide which algorithm is better than other, we use two parameter, namely, time complexity and space complexity.

Space complexity deals with how much space does a program take in your memory. But now since space is not a concern anymore, we do not pay attention to it.

Time complexity deals with how much time does a program takes to run. The lesser time your program takes to run, the better it is. It has been represented in three form i.e.  $O(1)$ ,  $O(n)$  and  $O(n!)$ .  $O(1)$  is said to be constant while  $O(n!)$  is lower bound and  $O(n!)$  is upper bound.

M	$\log(n)$	n	$n \log n$	$n^2$	$n^3$	$2^n$
4	2	4	8	16	64	16
16	4	16	64	256	4096	65536
256	7	256	1024	65536	262144	$2^{256} \times 10^{38}$

Hence we can say following given below series should always be yours preference  
 $O(1) > O(\log n) > O(n) > O(n \log n) > O(n^2) > O(n^3) > \dots$   
 $\Leftrightarrow$  polynomial function  
 $\Leftrightarrow O(2^n) > O(3^n) > O(4^n) \dots > O(n^n)$   
 $\Leftrightarrow$  exponential functions



1 sec = 10<sup>9</sup> operations (approx)  
20 sec = 2 × 10<sup>10</sup> operations (approx)

$\rightarrow n^2 \approx 4 \times 10^{10} \rightarrow n \approx 2000$   
 $\therefore n^2 \text{ naah dialegg}$   
 $= n \log n \text{ ka sef mihina naal}$