* TC Analysis of if & while

a = 1;

```
while(a<b)
{
    Statements;
    a = a*2;
}
```

Iterations a
1
$1 \times 2 = 2$
$1 \times 2 \times 2 = 4$
1×2^k

When ~~condition~~ loop will break?

→ $a \geq b$ will hit.

We are finding here how many times the while loop will run and it k times

$$2^k \geq b$$

Taking log on both side

$$\log 2^k \geq \log b$$

$$k \log_2 2 = \log b$$

$$\boxed{k = \log b}$$ log b times loop will execute

For for loop it will

```
for(i=1 ; i<n ; i=i*2){
    Statements;
}
```

$O(\log n)$

PAGE:
DATE: / /

(2)

```
i = n;
while (i > 1) {
    statement;
    i = i/2;
}
```

loop breaking condition

iterations $i <= 1$

$i = n$

$\frac{n}{2}$ $\qquad \frac{n}{2^k} <= 1$

$\frac{n}{2 \times 2}$ $\qquad n <= 2^k$

$\frac{n}{2^k}$ Taking log on both side for finding the value of k because loop is running k times

$\log n = k \log 2 \qquad \log_2 2 = 1$

$\boxed{k = \log_2 n} \rightarrow TC\ O(\log_2 n)$

(3)

```
i=1; k=1;
while (k<n)
{
    k = k+i;
    i++;
}
```

loop breaking condition

iterations $k >= n$

i	k
1	k+1
2	k+2
3	k+2+3
⋮	k+2+3+4 ...... m
M	

(k = 1 given)

$\frac{m(m+1)}{2}$

$k >= n$

$\frac{m(m+1)}{2} >= n$

$\frac{m(m+1)}{2} = n$

$m^2 = n$

$\boxed{m = \sqrt{n} \qquad O(\sqrt{n})}$

(4)

```
while (m != n) {
  if (m>n)
      m = m-n;
  else
      n = n-m;
}
```

Iterations

let's take

iterations $n = 18$	$n = 2$
$16 = 18 - 2$	2
14	2
12	2
10	2
8	2
6	2
4	2
2	2
1	

→ iterations = 9
that means
→ Time complexity is $\frac{n}{2}$.
→ after solving
further it will $O(n)$

TC Analysis of For loops.

①

```
for(i=1; i<n; i=i*2){
        statements;
}
```

loop breaking condition

$i \geq n$

iterations
1
1
1×2
$1 \times 2 \times 2$
$1 \times 2 \times 2 \times 2$
$1 \times 2 \ldots\ldots 2^k$ till run

$2^k \geq n$

Taking log on both side
for finding the value of K

$\log 2^k \geq \log n$

$k \log 2 \geq \log n$

where $\log 2 = 1$

$K = \log_2 n$

$TC = O(\log n)$

loop iterations

(2) for(int i=n; i>=1; i=i/2)
{
statements;
}

loop breaking condition
$i<1$

i	
n	— 1st iteration
$\frac{n}{2}$	
$\frac{n}{2\times2}$	
$\frac{n}{2\times2\times2}$	
⋮	
$\frac{n}{2^k}$	

$i<1$

$\frac{n}{2^k}<1$

$n<2^k$

We have to find the value of k for ~~finding~~ determining the overall loop iterations.

$n<2^k$

Taking log on both sides

$\log n < k\log 2$ $\qquad \log 2 = 1$

$k = \log n$

$TC = O(\log n)$

(3) for(int i=0; i*i<n; i++)
{
statements;
}

loop breaking condition
$i*i \geq n$

loop iterations

$i=0$	0×0
1	1×1
2	2×2
⋮	⋮
k	k^2

$k^2 \geq n$

$k = \sqrt{n}$

$TC = O(\sqrt{n})$

(4)

```
for (int i=0; i<n; i++)
{
    statements;
}
```

	Iterations
	i=0
	1
	2
loop breaking condition	3
$i \geq n$	4
$k \geq n$	:
$\boxed{k = n}$	k

$TC = O(n)$

(5)

```
P=0
for (int i=1; i<n; i=i*2)
{
    P++;
}
for (j=1; j<P; j=j*2)
{
    statements;
}
```

For first loop

iterations		
i=1	2	
2	2x2	
3	2x2x2	
4	:	
5	:	
k	2^k	

breaking condition for 1st loop

$i \geq n$

$2^k \geq n$

Taking log on both sides

$k \log 2 \geq \log n$

$\boxed{k = \log n}$

Tc of first loop is logn

$\therefore$ $P = \log n$

for 2nd loop

loop breaking condition

$j \geq P$

$j = 1$	$j = j * 2$ 1×2
2	2×2
3	$2 \times 2 \times 2$
4	$2 \times 2 \times 2 \times 2$
:	:
k	2^k

$2^k \geq P$

Taking log on both side

$\log 2^k \geq \log P$

$k \log 2 \geq \log P$

$\boxed{k = \log P}$ $TC = O(\log P)$

Evaluate both loop

$P = \log n$

$\log P$

$\boxed{TC = O(\log(\log n))}$