⇒ How your execution time increases if I increase the size of the input

	N	
A	10^4 (1 sec)	$\longrightarrow$ 10^5 (4 sec)
B	10^4 (1 sec)	$\longrightarrow$ 10^5 (300 sec)

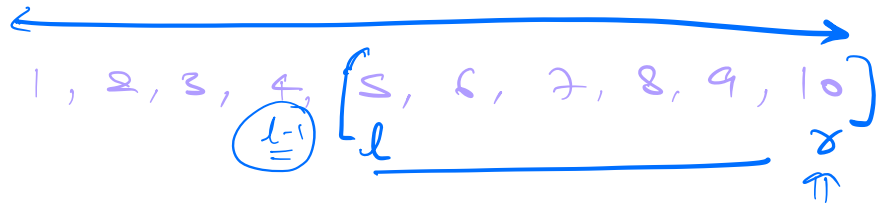
---

① $N \Rightarrow \log_2 N$ steps

② How many elements are there in the range $\Rightarrow [\overset{l}{3}, \overset{r}{10}]$

3, 4, 5, 6, 7, 8, 9, 10 ⇒ 8

elements $= r - l + 1$



$(1 \text{ to } r) - (1 \text{ to } (l-1))$

$r - (l-1) = r - l + 1$

$[l, r] \Rightarrow r - l + 1$

③ AP

3, 6, 9, 12, 15, 18, ...

3 3 3 3 3

$n \Rightarrow$ no. of terms in the sequence (6)
$a \Rightarrow$ first term (3)
$d \Rightarrow$ Common differen (3)

$$S_n = \frac{n}{2}\left(2 \times a + (n-1) \times d\right)$$

④ GP

2, 6, 18, 54 .....

×3 ×3 ×3

$a \Rightarrow$ first ten (2)
$r \Rightarrow$ Common ratio (3)
$n \Rightarrow$ no. of terms (4)

$$S_n = \frac{a\left(r^n - 1\right)}{(r-1)}$$

⑤ $\log_a a^x = x$

$\log_2 32 = \log_2 2^5 = 5$

Base

1) void func (int N) {

```
s = 0
for (i = 1; i <= N; i++) {
    [ s = s + i; ] ⇒
}
```

$i = 1, 2, 3, \ldots\ldots N$

$= [1, N] = N - 1 + 1 = N$

2) void fn (int N, int M) {

```
s = 0,
① [ for (i = 1; i <= N; i++) {
        s = s + i;
    }                                  ⇒ [1, N] ⇒ N
② [ for (j = 1; j <= M; j++) {
        s = s + j;                     ⇒ [1, M] ⇒ M
    }
```

iterations $= N + M$

③ void fn (int N) {

```
s = 0
for (i = 1; i <= N; i += 2) {
```

$s = s + i;$ ✗

}

$[1, N]$

$i \Rightarrow$ odd

$\left(\frac{N+1}{2}\right)$ ✓

$N = 7 \Rightarrow \frac{7+1}{2} = 7/2 = 3$ ✓

$N = 10 \Rightarrow \frac{10+1}{2} = 5$

$N/2 \Rightarrow \frac{N+1}{2}$ ✓

④
```
void fn(int N){
    s=0;
    for(i=0; i<=100; i++){
        s = s+i;
    }
}
```

$[0, 100] = 100 - 0 + 1 = 101$

⑤
```
void fn(N){
    s=0;
    for(i=1; i*i<=N; i++){
        s = s + i^2;
    }
}
```

2. trace

$i \Rightarrow [1, \sqrt{N}] \Rightarrow \sqrt{N}$ -times $= \sqrt{N}$

$i \times i <= N \Rightarrow i^2 \leq N$

$i \leq \sqrt{N}$

(6) void fn (N) {

int i = N

⇒ while (i > 1) {

i = i/2;

}

}

$i \Rightarrow [N \longrightarrow 2]$

T	i	
1	N	N/2 $N/2^1$
2	N/2	N/4 $N/2^2$
3	N/4	N/8 $N/2^3$
4	N/8	N/16 $N/2^4$
⋮		
K		$\frac{N}{2^K} = 1$

1/2

$\frac{N}{2^K} = 1 \Rightarrow N = 2^K$

Take $\log_2$ both sides

$\log_2 N = \log_2 2^K$

$\log_2 N = K$

(7)

```
void fn (int N) {
    s = 0;
    for (i = 0; i <= N; i = i*2) {
        s = s + i;
    }
}
```

$N \geq 0$

i = 0
= 0
= 0
= 0

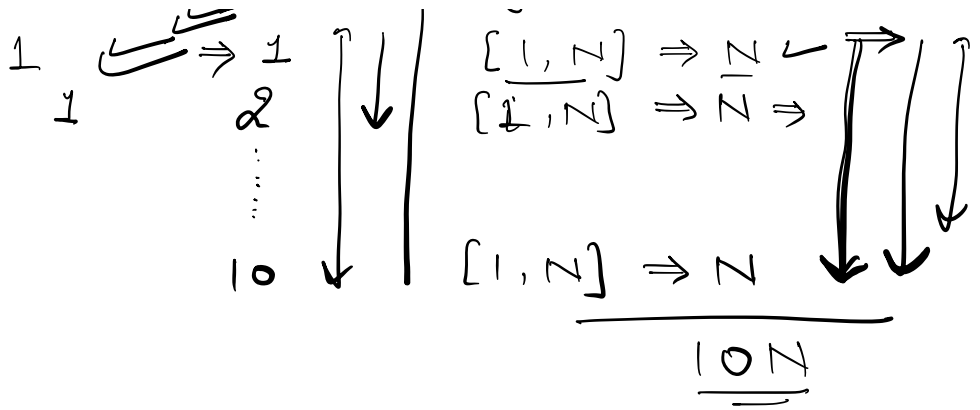
Infinite

(8)

```
void fn (N) {
    s = 0;
    (1) for (i = 1; i <= 10; i++) {
    ⇒       (2) for (j = 1; j <= N; j++) {
                s = s + i^j;
            }
    }
```

i | j

N+1



(9)

```
void fn(N) {
    s = 0;
    for (i=1; i<=N; i=i*2) {
        s = s+i;
    }
}
```

Iteration	before	after
1	1	$2 \Rightarrow 2^1$
2	2	$4 \Rightarrow 2^2$
3	4	$8 \Rightarrow 2^3$
4	8	$16 \Rightarrow 2^4$
⋮		
K		2^K

1, 2, 4, 8] 4

$2^K = N$

(16x) $K = \log_2 N + 1$

$N = 10$

$\log_2 10 = 3.\ldots$

1
2
4
8
16

(10) void fx (N)

s = 0;

① for (i = 1; i <= N; i++) {

⇒ ② for (j = 1; j <= N; j++)

s = s + i x j;

}

}

}

i
1
2
3
.....
N

[1 - N]

N
N
N
N

$N \times N$

iterations $= N^2$

(1) void for (N) {

s = 0;

(1) for (i = 0; i < N; i++) {

(2) for (j = 0; j <= i; j++) {

s = s + (i x j);

}

}

}

i	j	# iterations
0	[0, 0]	1
1	[0, 1]	2
2	[0, 2]	3
3		4
⋮		⋮
N-1	[0, N-1]	N

AP

$a = 1 + 2 + 3 + 4 \dots N$

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

$N = N$

$d = 1$

$$S_n = \frac{N}{2}\left(2 + (N-1)1\right)$$

$$\frac{N}{2}(N+1) = \frac{N^2+N}{2}$$

(12) void fn (N) {

for (i = 1; i <= N ; i = i+2) {

$\left(\frac{N+1}{2}\right)$

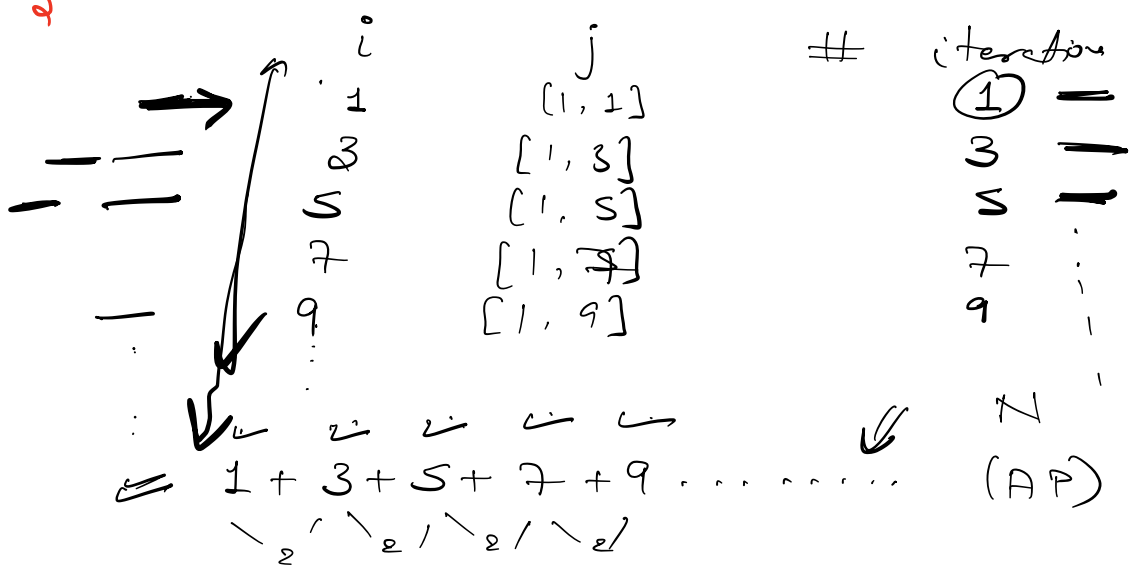
for (j = 1; j <= i ; j++) {

print (i+j);

}

}

}



$a = 1$

$d = 2$

elements in the sequence $= \left(\frac{N+1}{2}\right)$

$$S_n = \frac{n}{2}\left(2a + (n-1)d\right)$$

$$\frac{1}{2}\left(\frac{N+1}{2}\right)\left(2 + \left(\frac{N+1}{2} - 1\right)2\right)$$

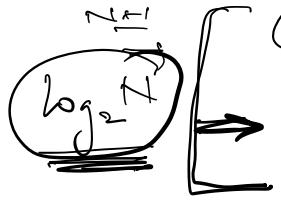
$$\frac{1}{2}\left(\frac{N+1}{2}\right)\left(2 + \frac{N+1-2}{2} \times 2\right)$$

$$= \frac{(N+1)(N+1)}{4}$$

$$= \frac{(N+1)^2}{4} = \left(\frac{N^2+1^2+2N}{4}\right)$$ (2+B)

(13)

```
void fn(N) {
    for (i=1; i<=N; i++) {
        for (j=1; j<=N; j=j*2) {
            print(i*j);
        }
    }
}
```



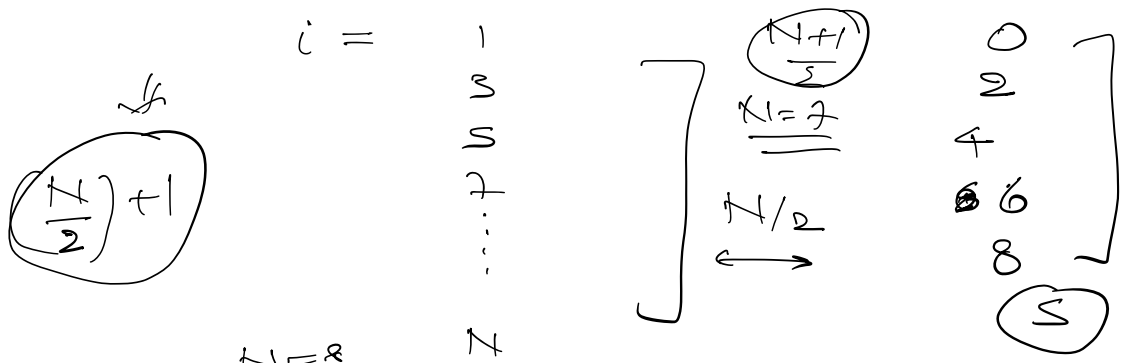
iteration = $N \times \log_2 N$

Doubts

```
for (i=1 ; i <= N ; i = i+2) {
```

//

5



N = 7

1, 3, 5, 7

← 4 times →

7/2 = 3

$\frac{N+1}{2}$

$N = 8$ $\quad (1, 3, 5, 7)$

$\leftarrow$ 4 times $\rightarrow$

$N/2$

$\frac{N+1}{2} = 9/2 = 4$