Today's class ?

→ TC & SC ✗

→ Asymptotic analysis ✗

→ Big O notation ✗

→ TLE ✗

general patterns & # of it around them !

Quiz 1 :

Sum of first N natural no's

$$1 + 2 + 3 + \ldots + N = \frac{N(N+1)}{2}$$

Quiz 2 :

[3, 10] → 3, 4, 5, 6, 7, 8, 9, 10

[ → Inclusive

( → Exclusive

| [a, b] | [a, b) | (a, b) |
| --- | --- | --- |
| b - a + 1 $\xrightarrow{-1}$ | b - a $\xrightarrow{-1}$ | b - a - 1 |
| a — b | a — b-1 | a+1 — b-1 |

Quiz 3 : $N \to N/2 \to N/4 \to N/8 \cdots \to 1$

$$\boxed{\log_2 N}$$

## A.P : Arithmetic Progression

Series : 4 7 10 13 16 19 22

3 3 3 3 3 3

In general

$$\overset{1}{a} \quad \overset{2}{a+d} \quad \overset{3}{a+2d} \quad \overset{4}{a+3d} \quad --- \quad \overset{N}{a+(N-1)d}$$

$$\text{Sum of AP} = \frac{n}{2}\left[2a+(n-1)d\right]$$

a: first term
d: common diff
n: no. of terms

$$\boxed{\log_a a^x = x}$$

## GP : Geometric Progression

Eg 3 6 12 24 48 96

2 2 2 → r

General

$$\overset{1}{a} \quad \overset{2}{a.r} \quad \overset{3}{ar^2} \quad --- \quad \overset{N}{a.r^{N-1}}$$

$$\text{Sum of first } N \text{ terms of a GP} = a\left[\frac{r^n-1}{r-1}\right] \quad r>1$$

$r \neq 1$

$$a\left[\frac{1-r^n}{1-r}\right] \quad r<1$$

a = first term
r = common ratio
n = no. of terms

Q

```
void fun (int N) {
    s = 0
    f (i = 1; i <= N; i++) {
        s = s + i;
    }
    return s;
}
```

i = 1  2  3  4  5 ... N

i : [1 - N]

# it → N

Q

```
void func (int N  int M) {
    f (i = 1; i <= N; i++) {
        if ( i % 2 == 0) {
            print(i);
        }
    }
    f (i = 1 ; i <= M; i++) {
        if( i % 2 == 0) {
            print (i);
        }
    }
}
```

i : 1, 2 --- N

i : [1, N]

# it → N

i : [1, M]

# it → M

$\boxed{\text{Total it} \rightarrow N + M}$

Q

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

$i = 1, 3, 5, 7, \ldots N$

$N = 10 \rightarrow 1, 3, 5, 7, 9 \rightarrow 5 \leftarrow 10/2$ ✓

$N = 7 \rightarrow 1, 3, 5, 7 \rightarrow 4 \leftarrow 7/2$ ✗

$\# \text{it} \rightarrow (N+1)/2$

\# odd nos [1, N]

Q

```
int func ( int N) {
    s = 0
    f(i= 0; i<=100; i++) {
        s= s + i + i^2
    }
    return s;
}
```

i= 0, 1, 2, ..... 100

#it → 101

Q

```
void func (N) {
    f(i=1; i*i <=N; i++) {
        s= s+i^2
    }
    return s;
}
```

$i <= \sqrt{N}$

$i^2 <= N$

$i <= \sqrt{N}$

$i = [1 .. \sqrt{N}]$

$\boxed{\#it \to \sqrt{N}}$

Q

```
void func(N) {
    i = N;
    while (i > 1) {
        i = i/2;
    }
}
```

initially $i = N$

| iteration | i |
|---|---|
| 1 | $N/2 \to N/2^1$ |
| 2 | $N/4 \to N/2^2$ |
| 3 | $N/8 \to N/2^3$ |
| 4 | $N/16 \to N/2^4$ |
| ⋮ | |
| K | $1 \to N/2^K$ |

$$\boxed{N/2^K = 1}$$

$N = 2^K$ take $\log_2$ on both sides

$$\boxed{\log_a a^x = x}$$

$\log_2 N = \log_2 2^K$

$$\boxed{K = \log_2 N}$$

$\boxed{\# it \to \log N}$

Q

```
void func (N)
    S = 0
    f (i = 0; i < N; i = i * 2){
        S = S + i;
    }
}
```

$\boxed{\#it \rightarrow \infty}$

$i = 0$

| it | i |
|---|---|
| 1 | 0 |
| 2 | 0 |
| 3 | 0 |
| ⋮ | |
| $\infty$ | |

Q

```
void func (N) {
    S = 0
    f (i = 1; i <= N; i = i * 2){
        S = S + i;
    }
}
```

$i \rightarrow 1$

| it | i | |
|---|---|---|
| 1 | 2 | $2^1$ |
| 2 | 4 | $2^2$ |
| 3 | 8 | $2^3$ |
| 4 | 16 | $2^4$ |
| ⋮ | | |
| K | N | $2^K$ |

$N$

$N/3$

$N/3$

⋮

$1$

$\log_3 N$

$N = 2^K$

$\log_2 N = \log_2 2^K$

$\boxed{K = \log_2 N}$

$\boxed{\#it \rightarrow \log_2 N}$

```
void func(N) {
    f(i=1; i<=10; i++) {
        f(j=1; j<=N; j++) {
            print(..);
        }
    }
}
```

| i | j: [1-N] | # it |
|---|---|---|
| 1 | [1-N] | N |
| 2 | [1-N] | + N |
| 3 | [1-N] | + N |
| ⋮ | | + ⋮ |
| 10 | [1-N] | + N |

Total it → 10.N

$\boxed{\#it \rightarrow 10N}$

```
void func(N) {
    f(i=1; i<=N; i++) {
        f(j=1; j<=N; j++) {
            print(i*j);
        }
    }
}
```

| i | j: [1-N] | # it |
|---|---|---|
| 1 | [1-N] | N |
| 2 | [1-N] | + N |
| 3 | [1-N] | + N |
| ⋮ | ⋮ | + ⋮ |
| N | [1-N] | + N |

Total #it → N.N

= $N^2$

```
void func(N) {
  f(i=0; i<N; i++) {
    f(j=0; j<=i; j++) {
      print(i*j);
    }
  }
}
```

| i | j:[0-i] | #it |
|---|---|---|
| 0 | [0-0] | 1 |
| 1 | [0-1] | + 2 |
| 2 | [0-2] | + 3 |
| 3 | [0-3] | + 4 |
| ⋮ | | ⋮ |
| N-1 | [0 - N-1] | + N |

$$\boxed{\text{total \#it} = \frac{N(N+1)}{2}}$$

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

| i | j:[1-N] | #it |
|---|---|---|
| 1 | 1.2.4..N | $\log_2 N$ |
| 2 | " | + $\log N$ |
| 3 | ⋮ | + " |
| 4 | ⋮ | ⋮ |
| 5 | ⋮ | ⋮ |
| ⋮ | ⋮ | ⋮ |
| N | " | + ⋮ |

$$\boxed{\text{total \#it} \rightarrow N \log N}$$

Q

```
void func(N) {
    f(i=1; i<=2^N; i++){
        print(i);
    }
}
```

$i = 1 \quad 2 \quad 3 \quad \ldots \quad 2^N$

$[1, 2^N]$

$\boxed{\# it = 2^N}$

Q

```
void func(N){
  f(i=1; i<=N; i++){
     f(j=1; j<=2^i; j++){
         print(i*j);
     }
  }
}
```

| i | j : $[1-2^i]$ | # it |
|---|---|---|
| 1 | $[1-2^1]$ | $\to 2^1$ |
| 2 | $[1-2^2]$ | $\to + 2^2$ |
| 3 | $[1-2^3]$ | $\to + 2^3$ |
| ⋮ | | + ⋮ |
| N | $[1-2^N]$ | $2^N$ |

Total #it $= \underbrace{2^1 + 2^2 + 2^3 + \ldots + 2^N}$ GP

$a = 2$

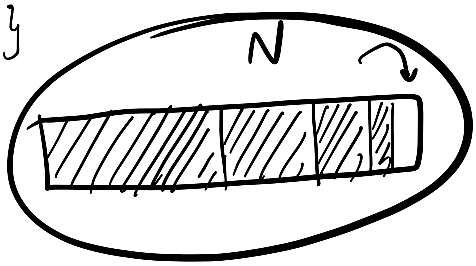
$r = 2$

$n = N$

Sum of GP $= a\left[\frac{r^n - 1}{r-1}\right]$

$= 2\left[\frac{2^N - 1}{2-1}\right] = \underline{2\left[2^N - 1\right]}$

```
for(i=N; i>0; i=i/2){
    for(j=1; j<=i; j++){
        print(i*j);
    }
}
```




| i | j : [1...i] | #it |
|---|---|---|
| 1 N | [1 – N] | $N \rightarrow N/2^0$ |
| 2 N/2 | [1 – N/2] | $+ N/2 \rightarrow N/2^1$ |
| 3 N/4 | [1 – N/4] | $+ N/4 \rightarrow N/2^2$ |
| 4 N/8 | [1 – N/8] | $N/8 \rightarrow N/2^3$ |
| ⋮ | ⋮ | ⋮ |
| $\log_2 N$ 1 | [1 – 1] | $1 \rightarrow N/2^{\log_2 N}$ |

total #it

$$= N + \boxed{N/2 + N/4 + N/8 + \cdots + 1}$$

$$= N + \left[N/2^1 + N/2^2 + N/2^3 + \cdots + N/2^{\log_2 N}\right]$$

$$= N + N\left[\frac{1}{2^1} + \frac{1}{2^2} + \frac{1}{2^3} + \cdots + \frac{1}{2^{\log_2 N}}\right]$$

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

$a = 1/2$

$r = 1/2$

$n = \log_2 N$

$$\cancel{1/2}\,\frac{\left[1-(1/2)^{\log_2 N}\right]}{\cancel{1-(1/2)}}$$

$$1 - \left(\frac{1}{2}\right)^{\log_2 N}$$

$$\boxed{\left(\frac{a}{b}\right)^n = \frac{a^n}{b^n}}$$

$$\boxed{2^{\log_2 N} = N}$$

$$\log_2\left(2^{\log_2 N}\right) = \log_2 N$$

$$1 - \frac{1^{\log_2 N}}{2^{\log_2 N}}$$

$$= 1 - \frac{1}{2^{\log_2 N}}$$

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

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

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

---

Compare

$$\boxed{\log_2 N < \sqrt{N} < N < N\log N < N\sqrt{N} < N^2 < N^3 \ldots < 2^N}$$

$N = 10^{18}$

| $\log_2 N$ | $\sqrt{N}$ |
|---|---|
| $\log_2(10^{18})$ | $\sqrt{10^{18}}$ |
| 60 | $10^9$ |

How to write Big O ? $\longrightarrow$ Next