Tutorial-1

(1)

```
int a=0, b=0;
for(i=0;i<N;i++)
{
  a = a+rand();
}
for(j=0;j<M;j++)
{
 b = b+rand();
}
```

Ans :

$O(N+M)$ — T.C.

$O(1)$ — S.C.

The first loop is $O(N)$
The second loop is $O(M)$
Since we don't know whether
$N>M$ or $M \ngtr N$ and no extra space
so S.C. is $O(1)$ and T.C. is $O(N+M)$

(2)

```
int sum = 0, i;
for(i=0;i<n;i=i+2)
{
sum += i;
}
```

i

0 $\quad 0+2+4+6+ \ldots n$

2 $\quad a=0$

4 $\quad d=2$

6 $\quad S = 0 + \left(\left(\frac{n}{2}\right)-1\right)2$

$\quad = n$

n $\quad O(n)$

(3)

```
int sum = 0, i;
for (i = 0; i < n; i = i * 2)
sum + = i;
```

2^0

2^1

2^2

.

.

.

.

.

2^{K-1}

$2^{K-1} >= n$

Taking log.

$(K-1) \log_2 2 >= \log n$

$K-1 >= \log n$

$(K >= \log n + 1)$

✓ $O(\log n)$

(4)

```
int sum = 0, i;
for (i=0; i*i < n; i++)
{
sum + = i;
}
```

$K * K >= n$

$K^2 >= n$

$(K >= \sqrt{n})$

$= O(\sqrt{n})$

(5)
```
int j=1, i=0;
while (i<=n)
{
 i=i+j;
 j++;
}
```

At ith iteration we are getting sum of first i numbers

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

$$K^2 >= n$$

$$K >= \sqrt{n}$$

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

(6)

$$T(n) = T(n-1) + T(n-1) + 1$$

$$= 2T(n-1)$$

$$= 2 2 \cdot T(n-2)$$

$$= 2 \cdot 2 \cdot 2 T(n-3)$$

$$\vdots$$

$$= 2^n.$$

$$O(2^n)$$

(7) $O(\log n)$

$$T(n) = 2T\left(\frac{n}{2}\right) + 1$$

$a = b$ $\quad \log_2 2 = 1$

$a = b = 2$

$f(n) = 1$

$g(n) = 1$

$T.C. = O(n\log(n))$

(8)

(i) $T(n) = T(n-1) + 1$ $\quad T(n) = \begin{cases} 1 & n=0 \\ T(n-1)+1 & n>0 \end{cases}$

$T(n-1) = T(n-2) + 1$

$T(n-2) = T(n-3) + 1$

$T(n) = [T(n-2) + 1] + 1$

$T(n) = T(n-3) + 3$

$\vdots$

$T(n) = T(n-k) + K$

Let $n-k = 0$

$T(n) = T(0) + n$

$T(n) = 1 + n \longrightarrow O(n)$

if $T(1) = 1 + 1 = 2$

(ii) $T(n) = T(n-1)+n$

$T(n-1) = T(n-2) + n-1$

$T(n-2) = T(n-3) + n-2$

$T(n) = T(n-3) + (n-2) + (n-1) + n$

$\vdots$

$T(n) = T(n-k) + (n-(k-1)) + (n-(k-2))$

$\quad - - - - - - - (n-1) + n$

Assume $n-k=0$

$\therefore n = k$

$T(n) = T(0) + (\cancel{n}-\cancel{k}+1) + (\cancel{n}-\cancel{k}+2) + - - - (n-1)+n$

$T(n) = T(0) + 1 + 2 + 3 + - - + n-1 + n$

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

$T(1) = 1 + \frac{(1)\cancel{(2)}}{\cancel{2}} = 2.$

(iii) $T(n) = T(n/2) + 1$

$a = 1$
$b = 2$

$k = \log_2 1 = 0$

$n^0 = 1 \cong 1$

so,

$T(n) = O(1 \log n)$
$= O(\log n)$

(iv) $T(n) = 2T(n/2) + 1$

$T(n) = O(n \log n)$

(v) $O(2^n)$

(vi) $O(3^n)$

(vii) $T(n) = T(\sqrt{n}) + 1$

Let $n = 2^m$

$T(2^m) = T(2^{m/2}) + 1$

$S(m) = T(2^m)$

$S(m) = S\left(\frac{m}{2}\right) + 1$

$k = \log_2 1$

$k = 0, m^k = 1$

$T(n) = O(\log \log n)$

(viii) $T(n) = T(\sqrt{n}) + n$

Let $n = 2^m$

$T(2^m) = T(2^{m/2}) + 2^m$

$S(m) = T(2^m)$

$S(m) = S(\frac{m}{2}) + 2^m$

$k = \log_2 1 = 0$

$m^0 = 1 \neq 2^m \quad O(2^m) = O(n)$

9.
```
int sum = 0, i;
for (i = 0; i < n; i++)
sum += i;
```

$0 + 1 + 2 + \dots$ n times

$\Rightarrow O(n^2)$

10. $N + (N-1) + (N-2) + \dots 1 + 0$

$= N(N+1)/2 = O(N^2)$

11. For n, j runs $O(\log n)$ times

i runs $\frac{n}{2}$ times

$T.C. = O(\frac{n}{2} * \log(n))$

$= O(n \log n)$

12. x is asymtotically more efficient than y means x is better choice for large inputs.

(2) correct.

13. $i = N$

$N/2$

$N/4$

$\vdots$

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

$\frac{N}{2^{K-1}} >= 1$

$N >= 2^{K-1}$

$\log N = \log_2 2^{(K-1)}$

$K-1 = \log_2 N$

$K = (\log_2 N + 1)$

$T.C. = O(\log n)$

14. $T(n) = 7T(n/2) + \underbrace{3n^2 + 2}_{f(n)}$

$a = 7$, $b = 2$

$K = \log_2 7 = 2.81$

$n^K = n^{2.81}$

a) $O(n^{2.81})$ ✓

b) $O(n^3)$ ✓

c) $\Theta(n^{2.81})$ ✓

15. $f2 > f4 > f3 > f1$

$2^n > n \hat{} 10 * 2 \hat{} (n/2) > (1.0001)^n > n\sqrt{n}$

b) $2^{n/2} * 2^{n/2} > n^{10} * 2 \hat{} n/2 > f3 > f1$

16. $f(n) = 2 \hat{} (2n)$

$f(n) >= c * 2n$

b) $\Omega(2 \hat{} n)$

17. $b = 2 \quad a = 1 \qquad f(n) = n^2$

$\log_2 1 = 0 = k$

$n^0 = 1$

$n^2 > 1$

(a) $O(n^2)$

18. $O(\log n)$

19. $O(N^2)$

$O(N)$

$\Rightarrow O(N^2 + N) = O(N^2)$.