Time Complexity :-

```
int fact=1;
for(int i=1; i<=n; i++)
{
    fact *= i;
}
Sopln(fact);
```

Program → Worst Case Time Complexity

1. Best Case Time Complexity (Ω) (Omega)
   $\Omega(1)$ || Ω(constant)
   $\Omega(1)$ or Ω(constant time)

2. Average Case Time Complexity (Θ) (Theta)
   $\Theta(n/2)$
   n = no. of el

3. Worst case Time Complexity (O) (Big Oh)
   $O(n)$; n = no. of element in an array.
   $O(n)$

Linear Search

x = 200

0	1	2	3	4	5
20	15	17	29	21	100

Last / Not found

(traverse)

$\simeq n/2$

n

n − 1

999999

$O(n^2) \to O(n)$

Asses

$f(n) = 2n^2 + 5n + 7$

S

$50 - 25 + 7$

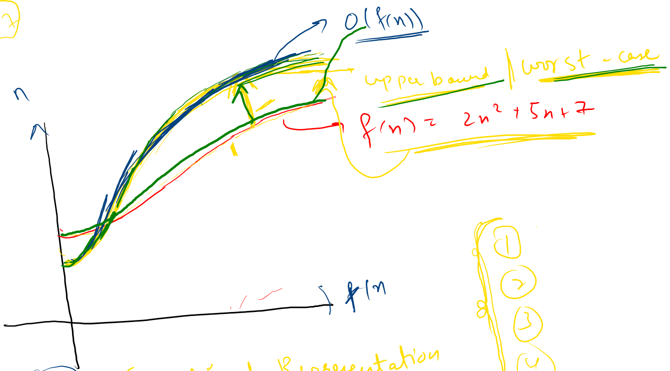
$f(n) = \cancel{2}n^2 + 5n + 7$

$$\boxed{O(f(n)) = cn^2}$$

50

$O(f(n)) = ?$

n^3



Graphical Representation

1) $f(n) = 2n^2 + \frac{5n^5}{n^3} + 7n^3$

2) $f(n) = (2n+5)(7\sqrt{n}+5) = n^{3/2}$

3) $f(n) = \frac{7n^3 + 5n^2}{n^4} = n^{-1}$

4) $f(n) = \sqrt[5]{n} + \sqrt{n}$ $\Rightarrow n^{5/2}$

1) $f(n) = 2n^2 + 5n^2 + 7n^3$
$= 7n^2 + 7n^3$

1
2
3
4

$f(n) = \underbrace{(2n+5)}(7\sqrt{n}+5$

a) $14n^{3/2} + 10n + 35n^{1/2} + 25$

$f(n) = n^{5/2} + n^{1/2}$

7) $f(n) = 7n^{-1} + 5n^{-2}$

T(n) = Total time calculate by array

Linear Search

1	2	15	19	75	80

Derivation

Linear search

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

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

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

$T(n-3) = K + T(n-4)$

$T(1) = K + T(0)$

$T(0) = 0$

$T(n) = \underbrace{K + K + K + K + \cdots + K}_{n\text{-times}}$

$\boxed{T(n) = T(n/2) + K}$ → Binary search

$T(n/2) = T(n/4) + K$

$n/2$

$n/2$

$n/2$

$O(T(n)) = ?$

$\boxed{T(n) = nK}$

$O(T(n)) = \boxed{O(n)}$

$T(n) = T(n/2) + K \rightarrow n/2$

$T(n) = T(\frac{n}{2^0})$

$T(n) = T(\frac{n}{2^0})$

$2^0 = 1$

$T(0) = 0$

$T(\frac{1}{2}) + K$

$\frac{1}{2} = 0.5 \approx 0$

$T(\frac{1}{2})$

$$
\left.
\begin{aligned}
T(n) &= T(\cancel{n/2}) + K \\
\cancel{T(n/2)} &= \cancel{T(n/4)} + K \\
\cancel{T(n/4)} &= \cancel{T(n/8)} + K \\
\cancel{T(n/8)} &= \cancel{T(n/16)} + K \\
&\vdots \\
T(1) &= \cancel{T(0)}^{0} + K
\end{aligned}
\right\} x \text{ terms}
$$

$T(n) = K + K + K + K + \dots K$

$\boxed{T(n) = xK}$

$\boxed{T(n) = K \log_2 n}$

$\boxed{O(T(n)) = O(\log_2 n)}$

n

$n/2,\ n/4,$

① $T(\frac{n}{2^0}) = T(n/2) + K$

② $T(\frac{n}{2^1}) = T(n/4) + K$

③ $T(\frac{n}{2^2}) = T(\frac{n}{8}) + K$

④ $T(\frac{n}{2^3}) = T(\frac{n}{16}) + K$

$\vdots$

x-term: $T(1)$

$\frac{n}{2^{x-1}}$

x-term $= T(\frac{n}{2^{x-1}}) = T(1)$

$x - 1 \approx x.$

$\frac{n}{2^{x-1}} = 1 \Rightarrow n = 2^{x-1}$

$\Rightarrow n = 2^x$

Taking log both side

$\Rightarrow \log_2 n = x$

Binary

$T(n) = T(n/2) + K$

$2^0 = 1$

$T(\frac{1}{2})$

$T(0)$

$n = \frac{n}{2^0}$

$$
\left.
\begin{aligned}
&① \quad T(\tfrac{n}{2^0}) = T(\cancel{n/2}) + K \\
&② \quad \cancel{T(\tfrac{n}{2^1})} = T(\cancel{n/4}) + K \\
&③ \quad T(\tfrac{n}{\cancel{4}\,2^2}) = T(\cancel{n/8}) + K \\
&④ \quad T(\tfrac{n}{\cancel{8}\,2^3}) = \cancel{T(n/16)} + K \\
&\quad \vdots
\end{aligned}
\right\}
$$

x terms $\rightarrow T(1) = \cancel{T(0)}^{0} + K$

$T(n) = (K + K + K + K + \dots K)$

$T(n) = xK$

$\boxed{T(n) = K \log_2 n}$

$\boxed{O(T(n)) = O(\log_2 n)}$ — Ans

$\boxed{\frac{1}{2} = 0}$ $\quad 0.5 = 0$

$n \longrightarrow 1$

$\frac{n}{2}, 1$

$\boxed{x\text{-terms}} \Rightarrow T(\frac{n}{2^{x-1}}) = T(1)$

$\frac{n}{2^{x-1}} = 1$

$x - 1 \approx x$

$n = 2^{x-1}$

$n = 2^x$

Taking log both sides

$\boxed{\log_2 n = x}$

$\log_2 n = x \log_2 2$ $\quad (2^x$

$\boxed{\log_2 n = x}$

$\log_2 2^x = x \cancel{\log_2 2}$

$= x$

$n = 5$

① for (int i = 1; i <= n; i++)

{

i = 1
i = 2
i = 3
i = 4
i = 5

= n times

$\boxed{O(n)}$

② for (int i = 1; i <= n; i++)

{

for (int j = 1; j <= m; j *= 2)

{

i = 1 → m
i = 2 → m
i = 3 → m

i = n → m

}

}

$\Rightarrow O(n * m)$

$n * m$

$n * n = n^2$

$n == m$

$\boxed{n \log n}$

③ for (int i = 1; i <= n; i *= 2)

{

G.P.

i = 1, 2, 4, 8, 16, 32, ... n

}

$\boxed{O(\log_2^n)}$

$n, \frac{n}{2}, \frac{n}{4}, \frac{n}{8}, \ldots 8, 4, 2, 1$

$a_n = a r^{n-1}$

$\boxed{n = 1 \cdot r^{n-1}}$

$n = r^{n-1}$

log

④ for (int i = n; i >= 1; i /= 2)

{

$n, \frac{n}{2}, \frac{n}{4}, \frac{n}{8}, \ldots$

}

$\boxed{O(\log_2^n)}$

$\log_2^n \Rightarrow \boxed{\frac{\log_e^n}{\log_e^2}}$

$n \quad O(n)$

for (int i = 1; i <= n;

$\log_e^n \Rightarrow \boxed{\log_2^n}$

$\boxed{\log_2 2 = \log_n^n = 1}$

$\frac{\log_e^n}{\log_e^n} = 1$

$\boxed{\log_3^{2^x} \Rightarrow x \log_3^2}$