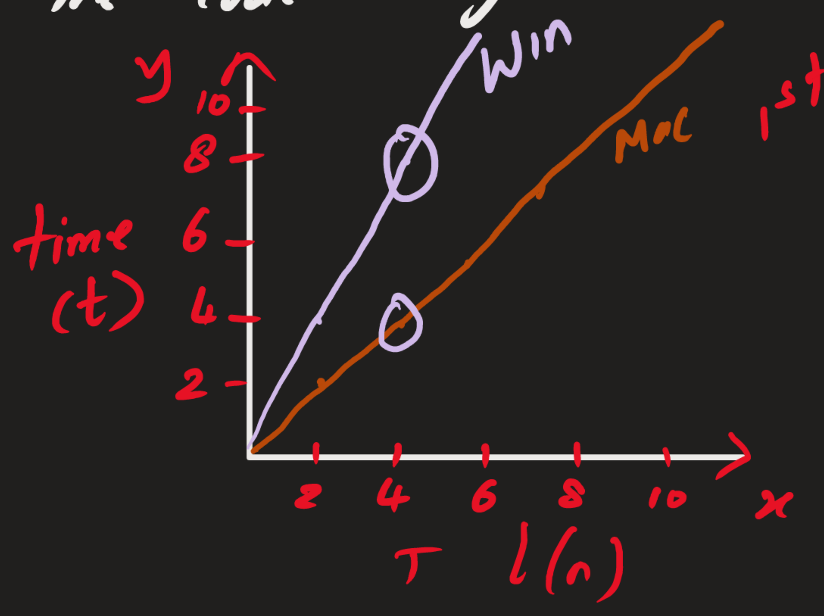
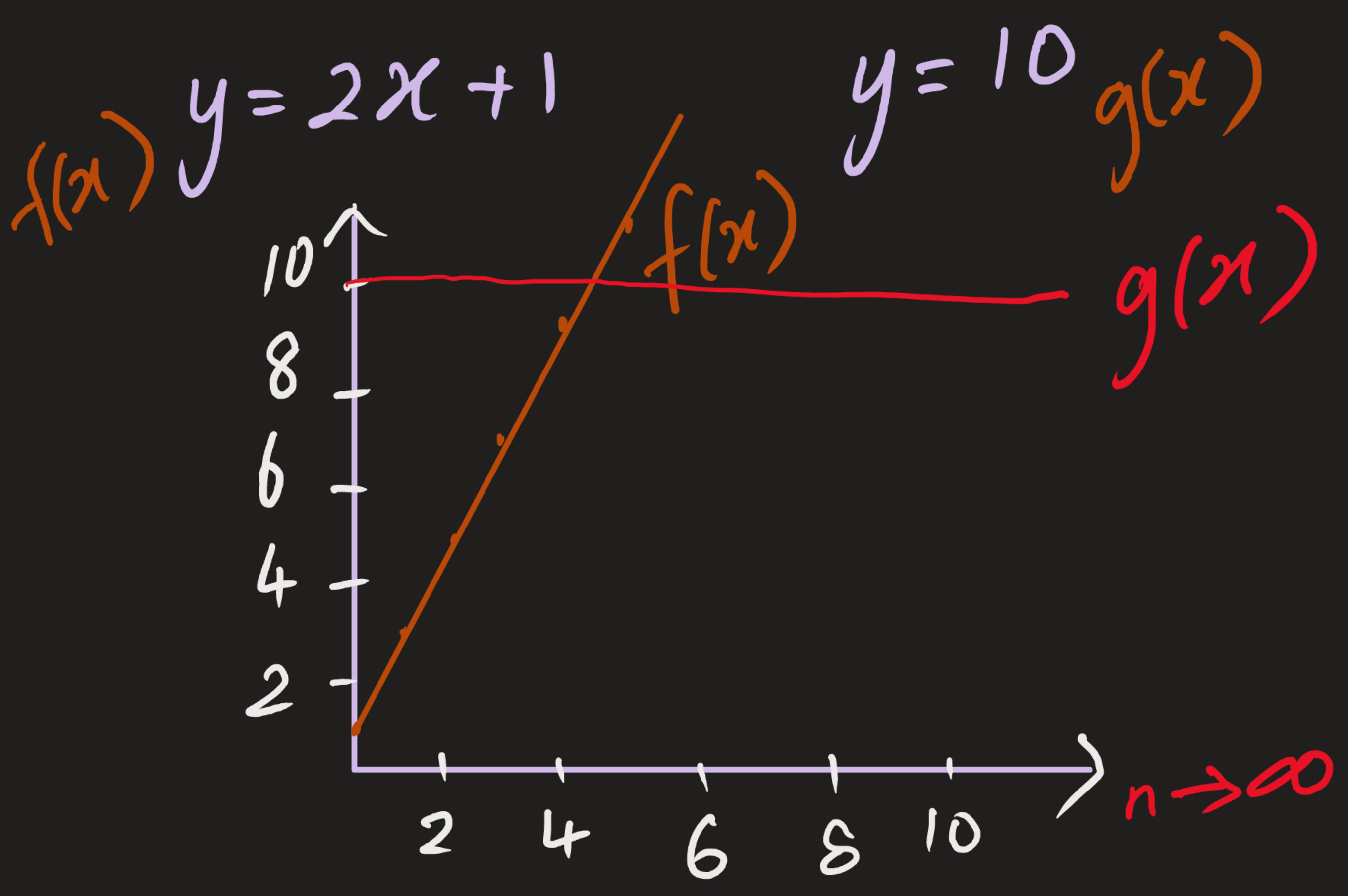
→ It is a rate at which time required to run the code changes w.r.t Input size.



$y = mx + C$

Comparing Functions



f(x) grows faster than g(x) as $n \to \infty$

If f(x) grows faster than g(x), then

$$\lim_{x\to\infty} \frac{f(x)}{g(x)} = \infty \quad \text{or} \quad \lim_{x\to\infty} \frac{g(x)}{f(x)} = 0$$

$$\lim_{x\to\infty} \frac{10}{2x+1} = \lim_{x\to\infty} \frac{10}{x\left(2+\frac{1}{x}\right)}$$

$$= \lim_{x\to\infty} \frac{\cancel{10}\,5}{x(\cancel{2})}$$

$$= \lim_{x\to\infty} \frac{5}{x} = 0$$

$f(x) = x^3 + 2x + 1 \qquad g(x) = x^2 + 3$

$$\lim_{x\to\infty} \frac{x^2+3}{x^3+2x+1} = \frac{\cancel{x^3}\left(\frac{1}{x} + \frac{3}{x^3}\right)}{\cancel{x^3}\left(1 + \frac{2}{x^2} + \frac{1}{x^3}\right)}$$

$$= \lim_{x\to\infty} \left(\frac{0+0}{1+0+0}\right)$$

$$= 0$$

f(x) grows faster than g(x)

Order of Growth:-

$O(1), O(n\log n), O(n), O(\log n)$

Direct Method:-

* Ignore lower Order terms

  $k < \log n < \sqrt{n} < n < n\log n < n^2 < 2^n < n!$

* Ignore leading Constants

1. $y = \cancel{2}n + \cancel{1}$ $\qquad y = 10$

   $O(n) > O(1)$

2. $n^3 + \cancel{2n} + \cancel{1}$ $\qquad n^2 + \cancel{3}$

   $O(n^3) > O(n^2)$

3. $n + 3\log n + \sqrt{n}$ $\qquad n\log n + 2^n$