Insertion Sort

Best Case → elements are already in sorted order. only one comparison in inner loop each time. total of (n-1) comparisons

Time complexity for best case is $O(n)$

$c_1 n$

Therefore, $c_1 n \le n-1$

$c_1 \le \frac{n-1}{n}$

$c_1 \le 1 - \frac{1}{n}$

As n increases ↑, $\frac{1}{n}$ decreases ↓, $1-\frac{1}{n}$ ↑

So $1-\frac{1}{n}$ would be min, when $n=2$

$\therefore$ $c_1 \le 1/2$

Worst case → Elements are present in reverse order

No. of comparisons

$(n-1)+(n-2)+ \text{-----} +(3)+(2)+(1)$

$\frac{n(n-1)}{2}$

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

Time complexity for worst case

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

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

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

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

$$\vdots$$

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

$$\because T(1) = 0$$

$$= \sum_{i=1}^{n} i$$

$$= \Theta(n^2)$$

complexity is $\Theta(n^2)$

$$\because \frac{n(n-1)}{2} \le c_2 n^2$$

$$\frac{n^2}{2} - \frac{n}{2} \le c_2 n^2$$

ignoring lower order terms we can take $c_2 \ge \frac{1}{2}$

Merge Sort

Best case - when largest element of one of the sublists is smaller than the first element of the other sublist.

Recurrence relation for merge sort is
$2T\left(\frac{n}{2}\right) + O(n)$

In best case - only one comparison from other list is done in each merge step
i.e. total $\frac{n}{2}$ comparisons are there.

Therefore,

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

$T(n) = 2T\left(2T\left(\frac{n}{4}\right) + \frac{n}{4}\right) + \frac{n}{2}$

$T(n) = 2\left(2\left(2T\left(\frac{n}{8}\right) + \frac{n}{8}\right) + \frac{n}{4}\right) + \frac{n}{2}$

$T(n) = 8T\left(\frac{n}{8}\right) + 4 \times \frac{n}{8} + n$

$T(n) = 8T\left(\frac{n}{8}\right) + \frac{n}{2} + n$

$T(n) = 8T\left(\frac{n}{8}\right) + \frac{3n}{2}$

⋮

$T(n) = 2^k T\left(\frac{n}{2^k}\right) + \frac{kn}{2}$

$2^k = N$
$k = \log_2 N$

$$T(n) = \frac{N}{2} \log N$$

Time complexity for Best case is $O(n \log n)$
$c_1 n \log n$

Therefore,

$$c_1 n \log n \leq \frac{n}{2} \log n$$

$$c_1 n \log n \leq \frac{1}{2} (n \log n)$$

for $c_1 \leq \frac{1}{2}$ inequality will be true.

$\boxed{\text{Worst Case}}$

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

where $(n-1)$ are total comparisons in each merge step

Total comparisons in worst case {Sorobesse}
$$T(n) \leq n \log n - n + 1$$

Time complexity for worst case is
$c_2 n \log n$

Therefore,

$$n \log n - n + 1 \leq c_2 n \log n$$

As $n \uparrow$, $\log n \uparrow$, $\frac{1}{\log n} \downarrow$

$n \uparrow$, $\log n \uparrow$, $n \log n \uparrow$, $\frac{1}{n \log n} \downarrow$

$$n\log n - n + 1 \le c_2 n\log n$$

$$\frac{n\log n - n + 1}{n\log n} \le c_2$$

$$1 - \frac{n}{n\log n} + \frac{1}{n\log n} \le c_2$$

$$1 - \frac{1}{\log n} + \frac{1}{n\log n} \le c_2$$

$\{ \log n < n\log n \Rightarrow 1/\log n > 1/n\log n \}$

It is clear that $\frac{1}{\log n}$ will always be < 1

$\frac{1}{n\log n}$ will also be < 1

then, $1 - (\text{anything} < 1) < 1$

therefore for $c_2 \ge 1$, inequality will be true

Quick Sort

$\boxed{\text{Best Case}}$ - When median of the list is chosen as Pivot, left & right splits both contain $n/2$ values.

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

$$\leq n\log n - n + 1$$

Recurrence relation for best case is

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

Solving by Master's theorem

$$a = 2, \quad b = 2, \quad f(n) = \Theta(n)$$

$$\therefore f(n) = \Theta(n^{\log_b a}) = \Theta(n)$$

$$\Theta(n^{\log_2 2}) = \Theta(n)$$

$$\Theta(n) = \Theta(n)$$

$$\therefore \quad T(n) = \Theta(n^{\log_b a} \log n)$$

$$= \Theta(n\log n)$$

$$c_1 n\log n$$

$$c_1 n\log n \leq n\log n - n + 1$$

$$c_1 \leq \frac{n\log n - n + 1}{n\log n}$$

$$c_1 \leq 1 - \frac{1}{\log n} + \frac{1}{n\log n}$$

for above inequality,

as $n \uparrow$

$\log n \uparrow$, $n \log n \uparrow$

$\frac{1}{\log n} \downarrow$, $\frac{1}{n \log n} \downarrow$

therefore $1 - \frac{1}{\log n} + \frac{1}{n \log n}$ also increases

which means as

$n \uparrow$, $1 - \frac{1}{\log n} + \frac{1}{n \log n} \uparrow$

$\therefore$ $1 - \frac{1}{\log n} + \frac{1}{n \log n}$ will be minimum ~~for~~ when n is minimum

$\therefore$ ~~for~~ for $c_1 \leq \frac{1}{2}$, inequality holds true

as minimum value of $n = 2$

$\Rightarrow 1 - \frac{1}{\log 2} + \frac{1}{2 \log 2}$

$\Rightarrow 1 - 1 + \frac{1}{2}$

$\Rightarrow \frac{1}{2}$

$\boxed{\text{Worst case}}$ - Pivot is either the largest or smallest element of list.

Recurrence relation is

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

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

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

$\vdots$

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

$\because T(1) = 0$

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

$= \frac{n(n-1)}{2}$

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

Time complexity for worst case is

$\Theta(n^2)$

$\leq c_2 n^2$

$\therefore \frac{n(n-1)}{2} \leq c_2 n^2$

$\frac{n^2 - n}{2} \leq c_2 n^2$

$\frac{n^2}{2} - \frac{n}{2} \leq c_2 n^2$

Ignoring lower order terms

$\frac{1}{2} \geq c_2$, inequality will be true.