Selection Sort

↳ Select an element and put it in its Correct Index

4 5 1 2 3

0 1 2 3 4

Ex:

(1) Targest element =5

4 5 1 2 3

0 1 2 3 4

→ Swapped with Correct Index

(2) largest element =4

4 3 1 2 5 (Initial)

0 1 2 3 4

4 3 1 2 5

0 1 2 3 4

(3) LE =3

2 3 1 4 5 (Initial)

0 1 2 3 4

2 3 1 4 5

(4) LE =2

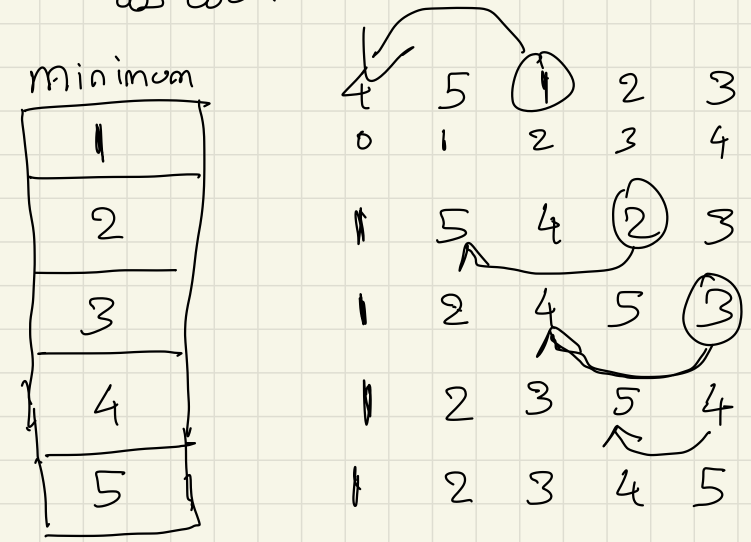
2 1 3 4 5

0 1 2 3 4

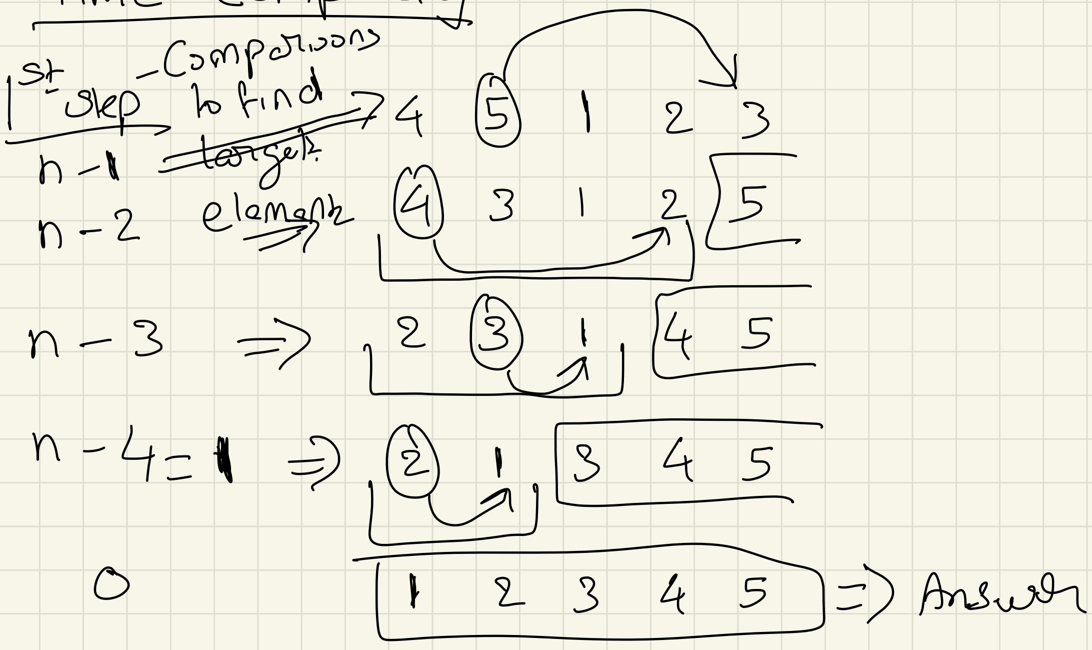
2 1 3 4 5

(5) 1 2 3 4 5 Answer

Ex2 Can be done using minimum element as well



Time Complexity



Total Comparison $= 0+1+2+\dots+(n-1)$

$= n(n-1)/2$

[Hint : Sum of n numbers = $n(n+1)/2$]

Here we need Sum of $(n-1)$ numbers,

So replace n with $(n-1)$

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

total Comparisons (Cont...) $= \frac{n^2 - n}{2} = O(n^2)$

(Constants (2) and less dominating terms can be Removed)

Worst Case $= O(N^2)$

Best Case $= O(N^2)$

Stability $= \boxed{\text{No}}$

Use Case : It performs well on Small lists / Arrays

Algorithm

For iteration of i, we are running the loop $(n-i-1)$ times

$$\boxed{i <= n-i-1}$$

Because after every iteration we can eliminate the end elements which are already sorted

⇒ And i is iterated $\boxed{N-1}$ times.