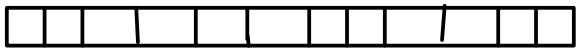
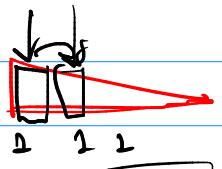
Sorting Revision

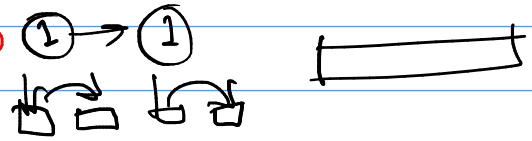


Sorting

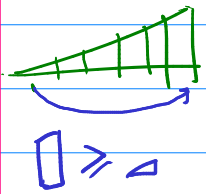
1. Find the biggest number among n numbers.
   → Do it for n times
2. 

$1 \times n = O(n)$

Best: Already Sorted



Bubble Sort : $A_i \geq A_{i+1}$



(n)

$n \times n = O(n^2)$

Rule of Thumb:

If avg case is unknown
→ then worst case is avg case

$$\lim_{n \to \infty} \frac{f(n)}{g(n)} \begin{cases} = 0 & (\Omega) \\ = \infty & (O) \\ = c \mid c \neq 0; c \in \mathbb{Z}^{\oplus} & (\Theta) \end{cases}$$

$\Theta \rightarrow O + \Omega$

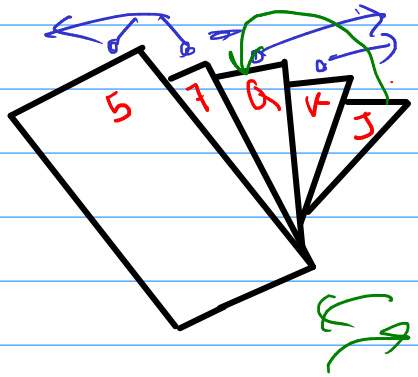
$f(n) \leq g(n) \rightarrow O \rightarrow f(n) < g(n)$ ✗

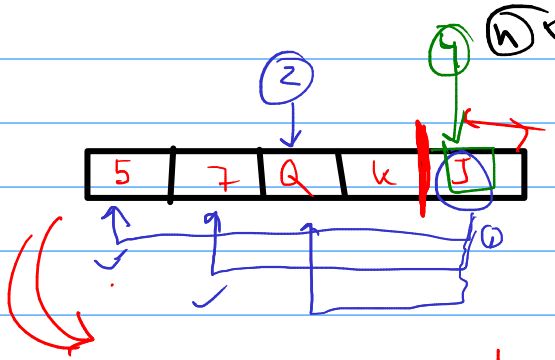
$f(n) \geq g(n) \rightarrow \Omega \rightarrow f(n) > g(n)$

$n \quad n^2 \quad n^2$

(n) k

Insertion Sort





$0 - (j-1)$

$j++;$

5 7 Q K J

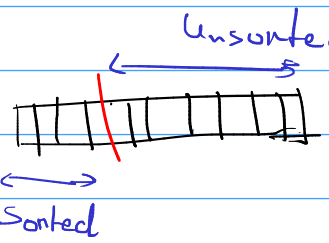
J

5 7 J Q K

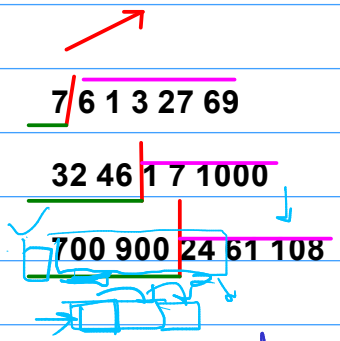
Insertion Sort Autopsy

① Dividing the array into two different segments by putting a seperator. (LTR)

Ⓐ Sorted Ⓑ Unsorted



Q) Is it possible?



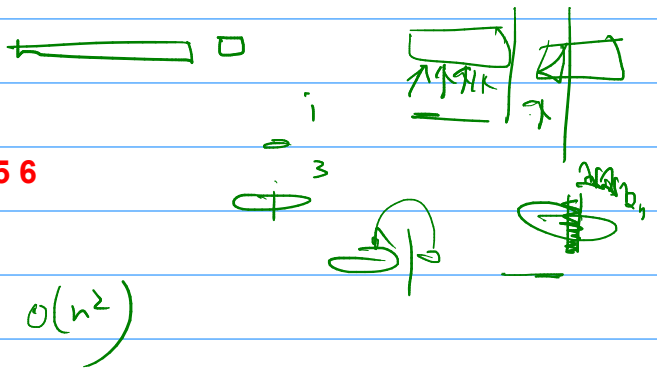
Insertion Sort is an optimistic sort
↳ A fragment of the input is already sorted.

↳ An one element array is always sorted.

② Take elements from the unsorted section and put them into their rightful places in the sorted section, one by one.

↳ ① Find the place for the new element
② Expand the size of the sorted section by 1
③ Vacate the place found in ①.

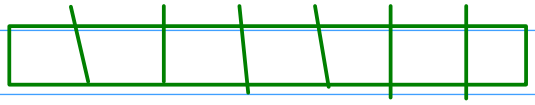
Best Case:



n × n
$O(n^2)$
↳ $O(n^2)$

$O(n^2)$

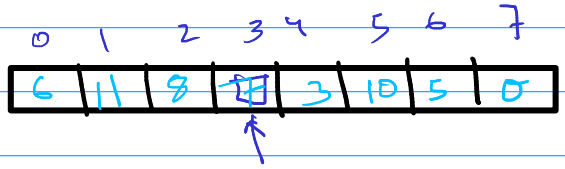
Selection Sort



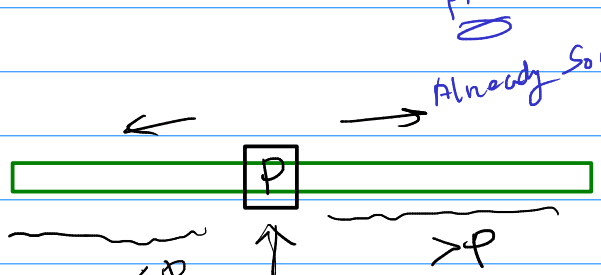
Selection	Insertion
$O(n^2)$	$O(n^2)$
$O(n^2)$	$O(n^2)$
$O(n^2)$	$O(n^2)$

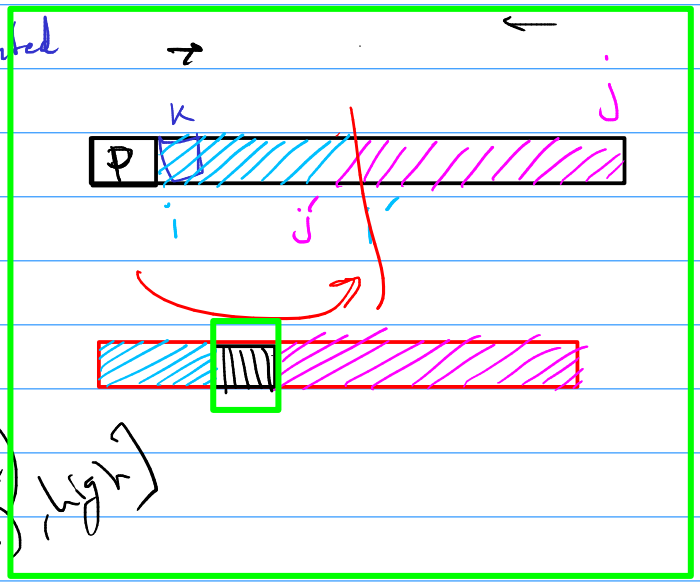
Re
- Sort in element
- Subtle array

Quicksort

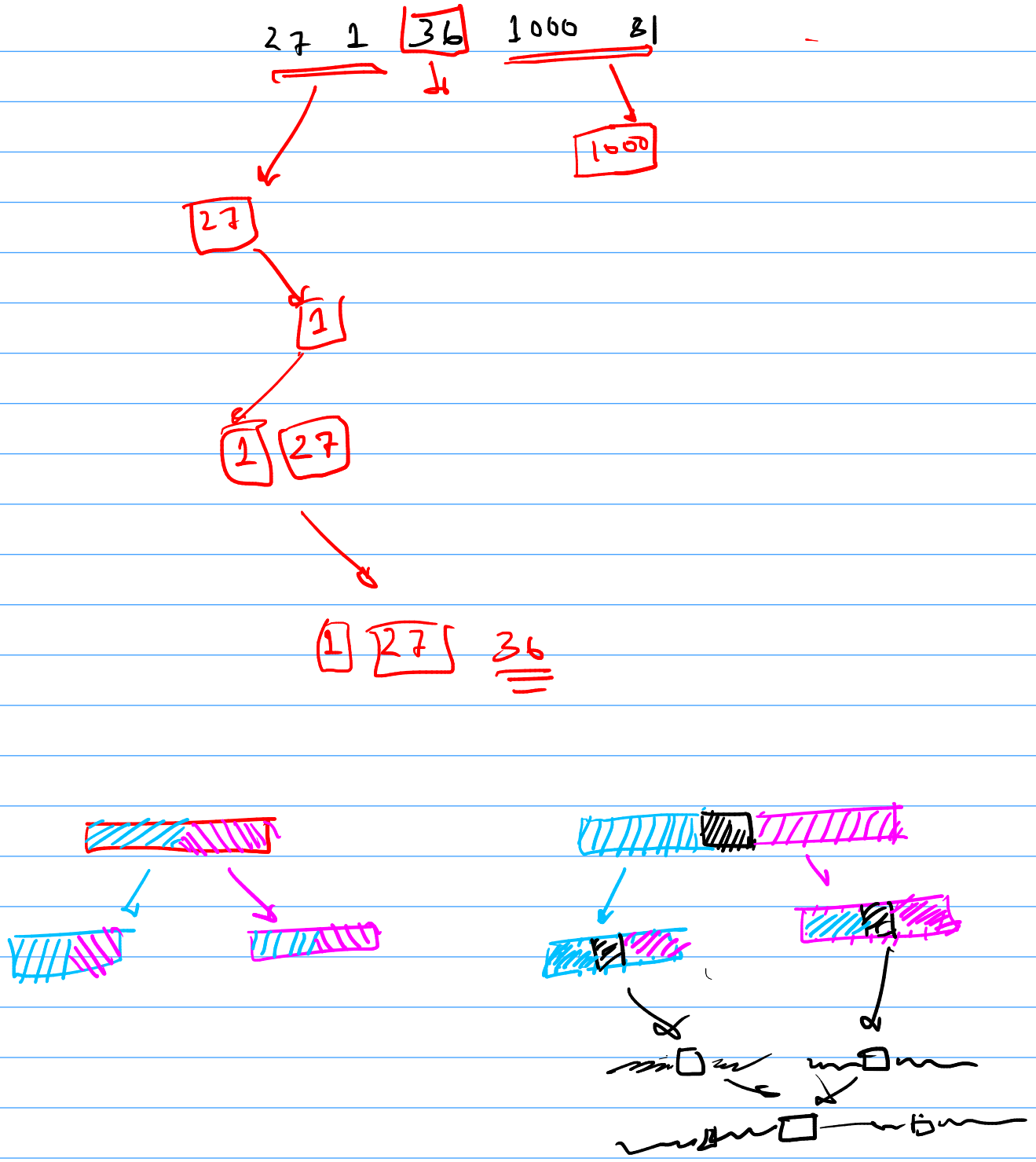


$\frac{0+7}{2} = 3.5 \downarrow 3$






```
QSRT(high, low)
while(i< j)
pivot = A[i+j]/2
QSRT(low, A[pivot-1])
QSRT(A[pivot+1], high)
```


27 1 36 1000 81
1000
27
1
1 27
1 27 36