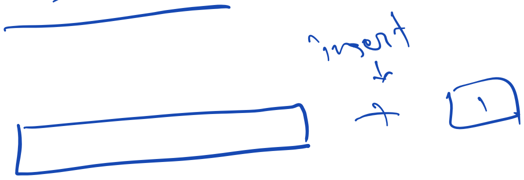
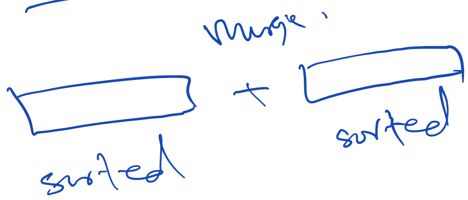
Quick Sort

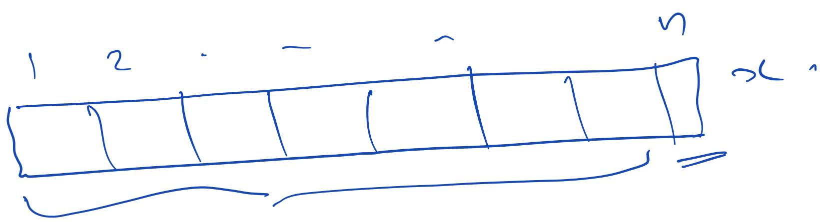
I/P : A sequence of n integers.
O/P : The numbers in non-decreasing order.

Insertion



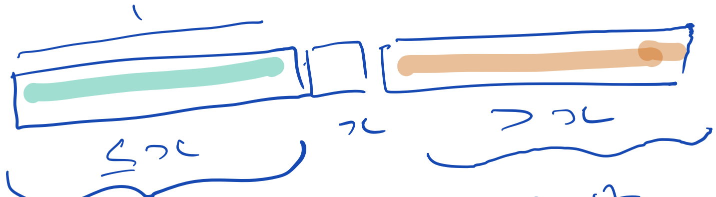
Merge Sort





① Choose one element as pivot.

② Partition according to x.

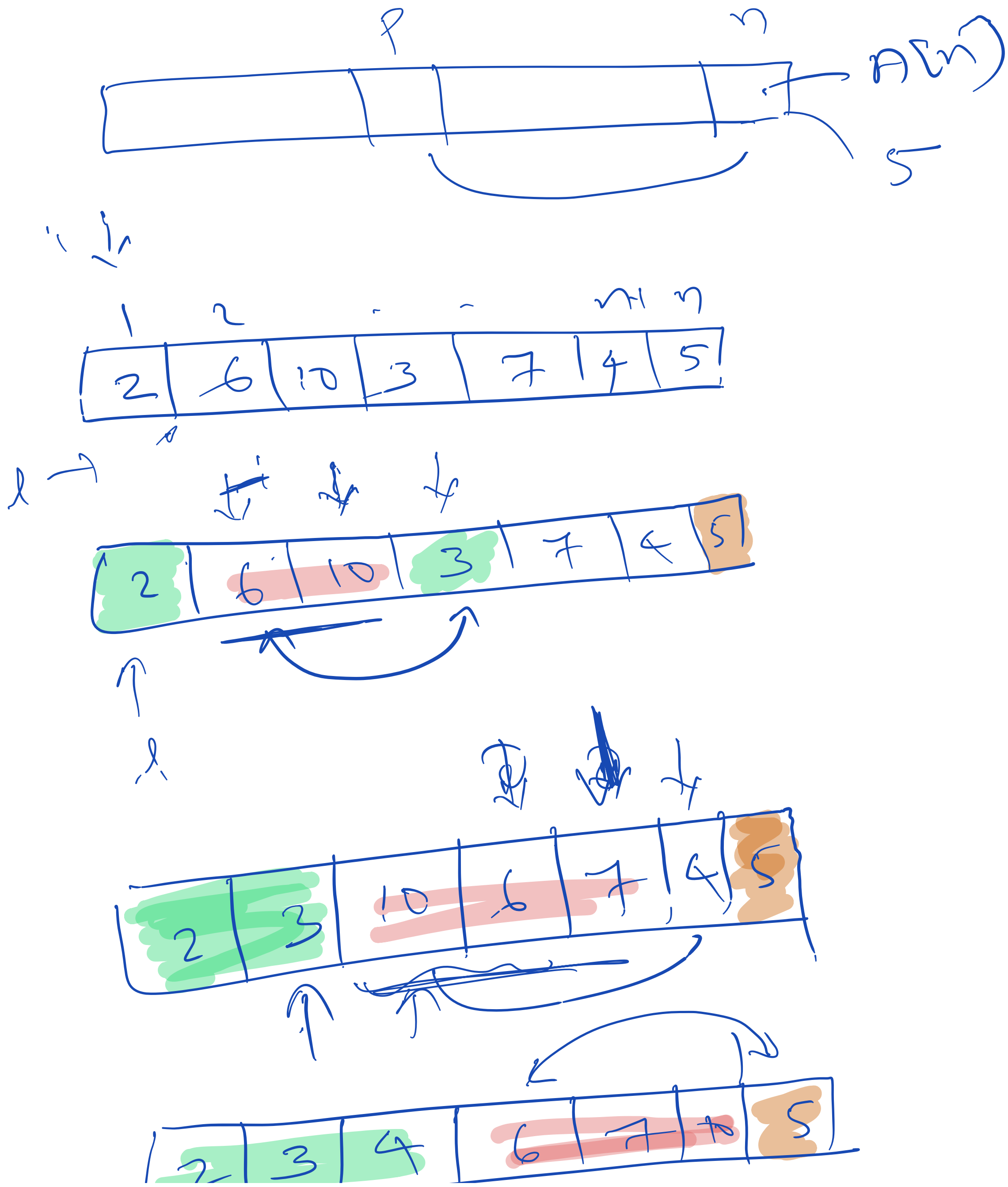


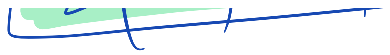
(recursively).

Input elements are distinct

```
QuickSort(A[1..n]):
  if (n > 1)
    Choose a pivot element A[p]
    r ← Partition(A, p)
    QuickSort(A[1..r − 1])     ⟨⟨Recurse!⟩⟩
    QuickSort(A[r + 1..n])     ⟨⟨Recurse!⟩⟩
```

```
Partition(A[1..n], p):
  swap A[p] ↔ A[n]
  ℓ ← 0                  ⟨⟨#items < pivot⟩⟩
  for i ← 1 to n − 1
    if A[i] < A[n]
      ℓ ← ℓ + 1
      swap A[ℓ] ↔ A[i]
  swap A[n] ↔ A[ℓ + 1]
  return ℓ + 1
```



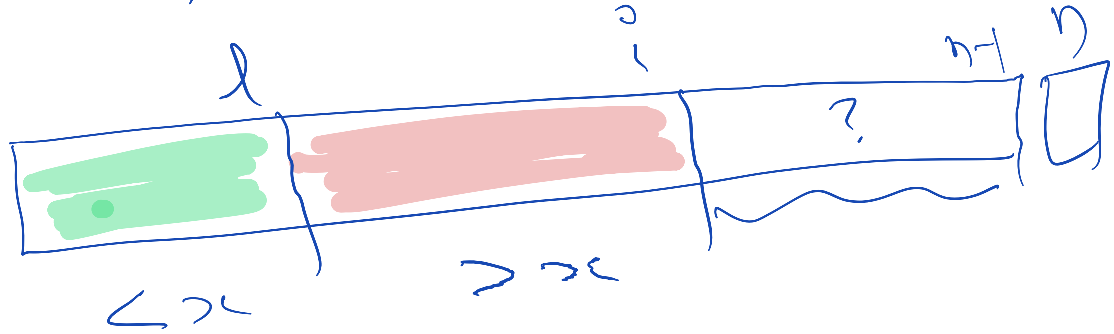


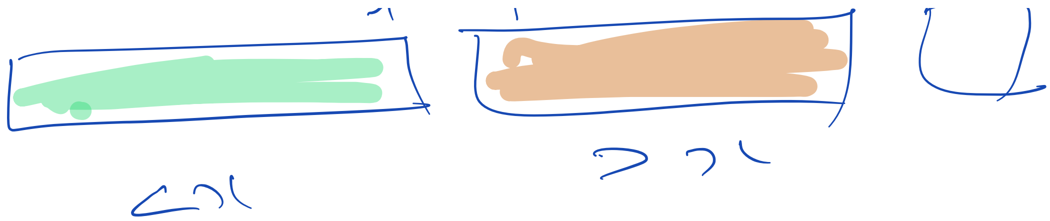
2	3	4	5	7	10	6

Correctness proof.

Loop-invariant.

At the end of each iteration, all the elements from 1 to l are less than pivot element. All the elements b/w l+1 and i are greater than the pivot.

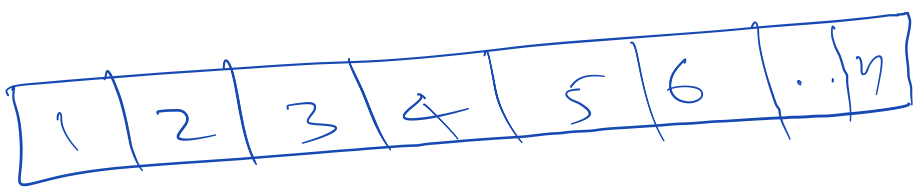




Time complexity:

$T(n)$ be the worst-case running time.

$$T(n) \leq 2T(n/2) + cn$$
$$\leq O(n \log n)$$



$$T(n) \leq T(n-1) + cn$$
$$\leq O(n^2).$$

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

$$T(1) = 1$$

Statement

$$T(n) \le 5cn^2$$

Proof by induction.

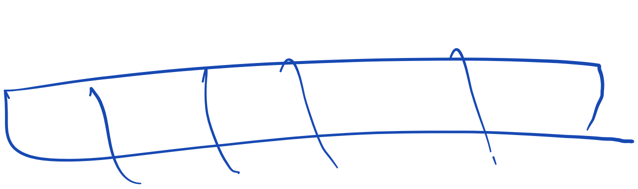
maximum when $a=0$ or $a=n-1$

$$T(n) \le \underbrace{5ca^2 + 5c(n-a-1)^2}_{f(a)} + cn$$

$$\le 5c(n-1)^2 + cn$$

$$\le 5cn^2$$

$$T(n) = \Omega(n^2)$$



for sorted array

$$cn + c(n-1) + \cdots$$

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

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

$\frac{1}{10}n \le$ size $\le \frac{9n}{10}$

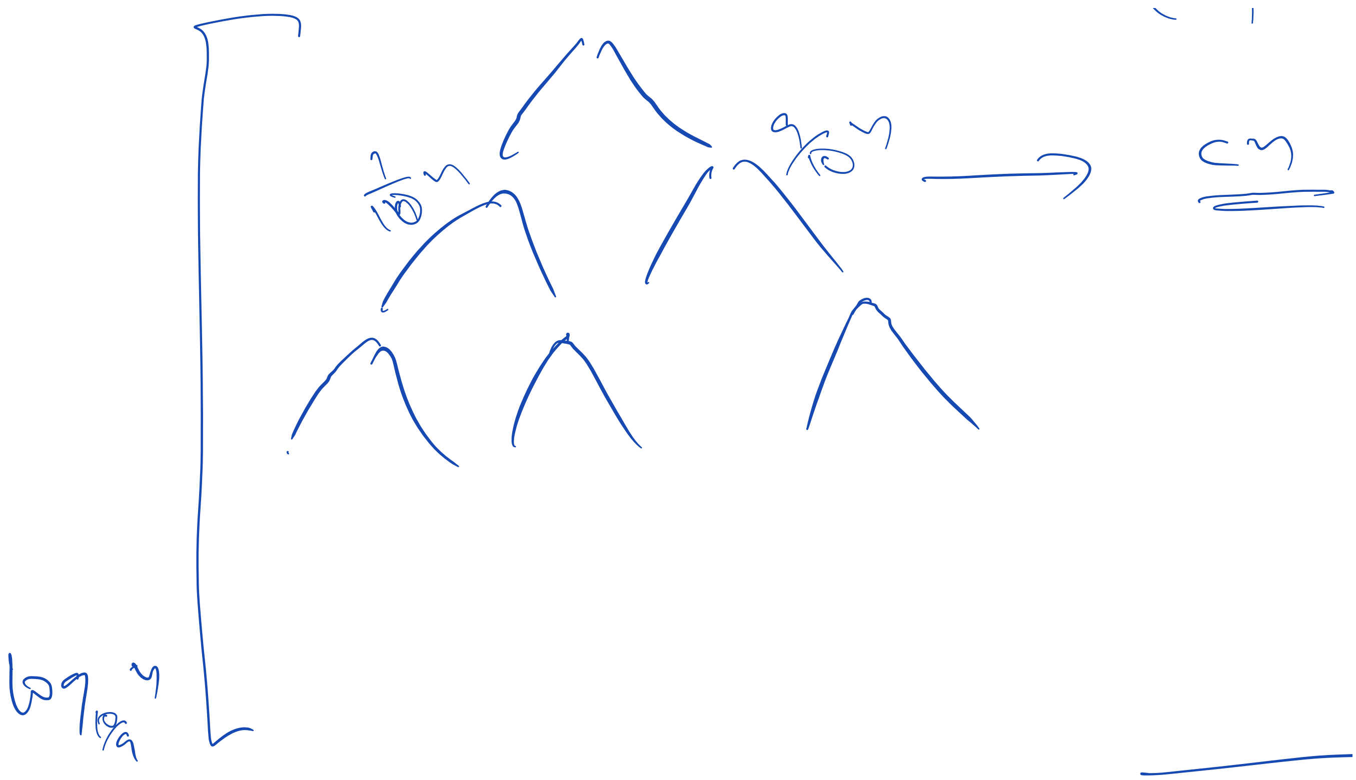
q

$\frac{1}{10}n \le$ size $\le n\frac{9}{10}$

$(n-q)$.

$$T(n) = T(q) + T(n-q) + cn$$

cn



$cn \log_{10/9} n$

$O(n \log n)$