

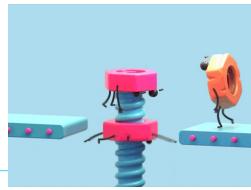
- Nut & Bolts example
- partitioning
- Quick sort
  - Worst Case scenario
  - Count Sort
- Bucket Sort
- Radix Sort
- Inversion Numbers

Quick sort 1961 is faster than merge sort & heap sort  
for randomized data.

Goal of this problem: intuition on  
partitioning

Nuts & Bolts

what is  
partitioning?



ex 1 Given  $N$  nuts &  $N$  bolts of different sizes



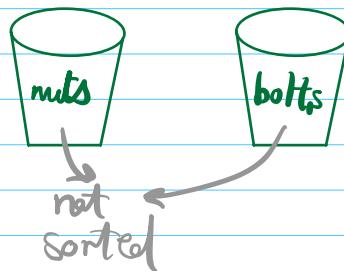
there is 1:1 mapping for nut & bolt.

what if... ?

match Nuts & Bolts. Constraint: we cannot compare a nut with other nuts & a bolt with other bolts.

nut  $\rightarrow$  bolt

- ① fit
- ② smaller
- ③ longer



foreach nut in nuts

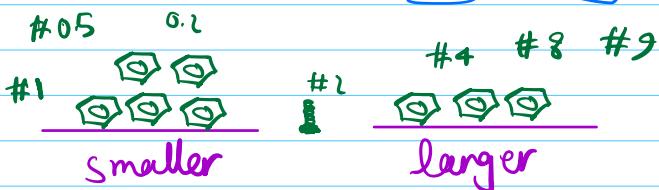
foreach bolt in bolts

$O(N^2)$

```
if (nut fits bolt){  
    match; break;  
}
```



pick a  
random  
nut



$N$  steps

partitioning

$$x \in A \{a_0, a_1, \dots, x, a_i, \dots, a_{n-1}\}$$

Given an integer array, can element  $X$  rearrange elements st.  
 $\forall i, \text{if } A[i] < X \text{ then it is on left side, otherwise}$   
on right side.

ex  $\{9, 8, 1, 6, 5, 8\} \quad x=6$

how to  
do this  
in a  
Code?!

$$\{\{1, 5\} | \{6, 8, 1, 8\}\} \checkmark$$

$$\{\{5, 1\} | \{6, 8, 1, 8\}\} \checkmark$$

$$\{\{5, 1\} | \underline{\{6, 1, 8, 8\}}\} \checkmark$$

$$\{\{5, 1\} | \{8, 1, 6, 8\}\} \times$$

1-how? later

2-what the problem  
means by left &  
right? to the  
left of  $x=6$  or  
in general to left  
Confusing how to verify?

3-Why do we  
do this?

\* this is not  
quick sort yet!  
it will be used  
in quick sort!

original array  $\{8, 1, 6, 5, 8\}$

Pivot  
element

$x=6$

$$\begin{array}{ccccccccc} & & \leftarrow \text{rand} & & & & \text{random} & & \\ & & \downarrow & & & & \uparrow & & \\ \{ & 0 & 1 & 2 & 3 & 4 & \dots & & \} & x & & & & & & & \\ & \downarrow & & & & & & & \\ \{ & 1, 5, | & 6, 8, 8 \} & x & & & & & \\ & i \nearrow & & & j \searrow & & & & \\ & & & & & & & & \end{array} \quad x = a[\text{rand}]$$

if ( $a[j] > x$ )

$j++$ ;  
do nothing

else //  $a[j] < x$

① swap  
( $a[i], a[j]$ )

②  $i++$ ;  $j++$

$$\{3, 1, 0, 2, 5, 7, 5\} \quad x=5$$

$$\{1, 0, | 2, 3, 5, 7, 5\} \quad x=2$$

$$\{00\} \{00000\}$$

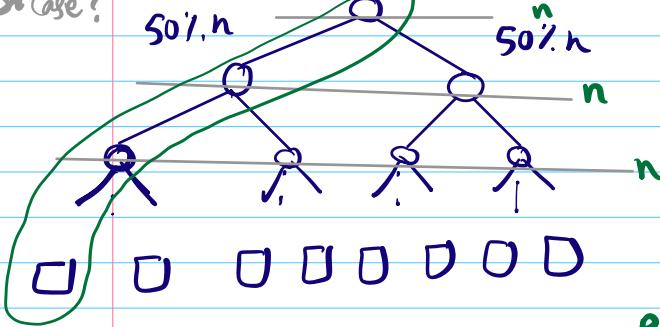


→ looks like a slow sort rather than quick sort!?

### Quick Sort:

{7, 3, 2, 5, 1, 6, 4}

average?  
what about  
worst case?



TC:  $O(n)$   
SC:  $O(1)$

← for each step  
not the total  
quick sort (not  
the stack)

$\log n / 2$  height

SC:  $O(\log n)$   
TC:  $O(n \log n)$

pivot Index

int partition(A, st, end){

```
void quickSort(A, st, end){
    if (st >= end) ret;
    pi = partition(A, st, end)
    quickSort(A, st, pi-1)
    quickSort(A, pi+1, end)
}
```

pe = A[end]

i = st

for(j = st; i <= end-1; j++) {

if (A[j] < pe) {

swap(a[i], a[j])

i++

no need to  
go to last

}

swap(A[i], A[end])

ret i

a[j] < x

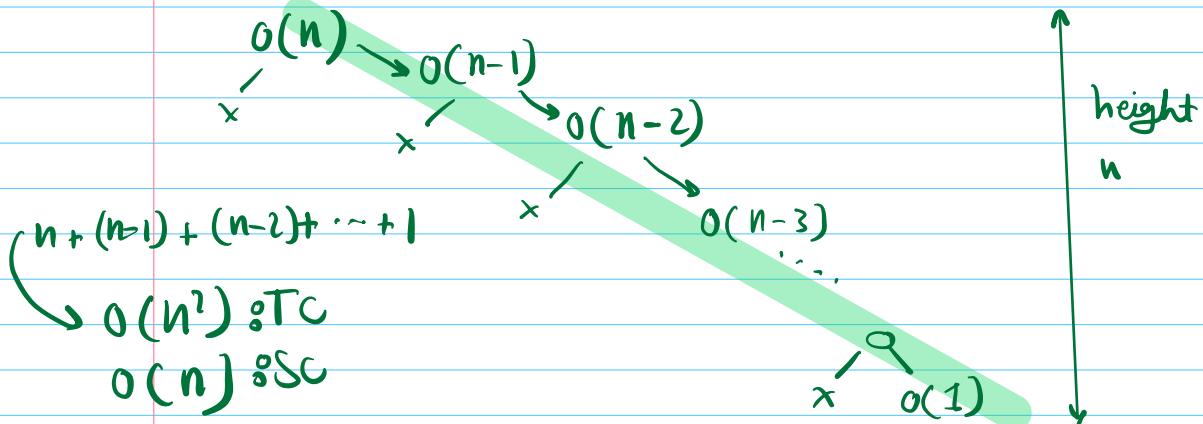
j++

worst  
case

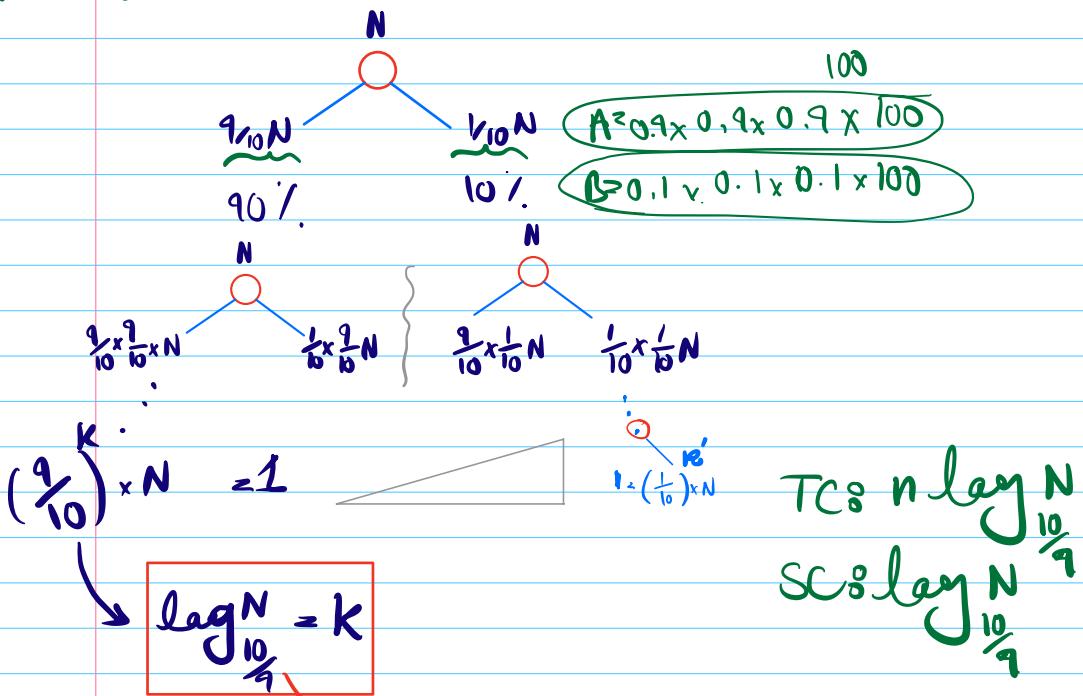
Quiz

1, 5, 4, 3, 2,

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



bad luck



$$16.6 \quad \log_c n = k$$

$$\log_2 10^5 = \log_2^{10} 10^3 + \log_2^{10}$$

$$16.6 \quad \log_{10/9} 10^5 = \log_2 10^5 / \log_2 10/9 \rightarrow 0.15$$

$$\log_2 N \approx 100$$

$$0.16 \dots$$

P1 Find the smallest number that can be formed by

arranging the digits of the given input

ex  $a \in \{1, 3, 5, 2, 3\}$

1, 2, 3, 3, 5  
12355

Sort asc.

q-sort or  
merge sort

u.lagn)

$0 \leq a[i] \leq 9$

$\underbrace{\text{len}(a)}_N \leq 10^9$

$n(\text{length})$

ex a: { 1, 5, 2, 1, 3, 0, 5, 1 }

$$\text{X} \quad \underline{0, 1, 1, 1, 2, 3, 5} \checkmark$$

Freq [10]

$$a \left\{ 0, 1, 2, 8, 0, 1, 2, 8, 0, 1, \dots, 2, 1, 1, \dots, 5 \right\}$$

0 1 2 3 4 5 6 7 8 9

Freq | 1 3 4 5 1 2 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 2

{11111225899}

ex  $a \in \{1, 3, 8, 3, 2, 6, 5, 3, 8, 1\}$

0 1 2 3 4 5 6 7 8 9

Freq | 0 0 2 1 3 0 1 1 0 2 0 | 1125555688

## Count

1123335688

| for( $i=0$ ;  $i < n$ ;  $i++$ )  
 $O(n)$  | freq[a[i]] += 1

## Sont

$$TC \approx O(n+n+k) \sim$$

2 for ( $i \geq 0$ ;  $i \leq k$ ;  $i++$ ).}

## Quiz

$\delta C \approx O(k)$

```

2 for (i=0; i < k; i++) {
    for(j=0; j < freq[i]; j++) {
        print(i)
    }
}

```

① what if  
 $0 \leq a[i] \leq 10^3$

$\tilde{t} \mapsto n$  times

$i++ \rightarrow k$  times

② what if

$$-10 \leq a[i] \leq 10$$

③ no custom Comparator

two prereqs

①  $\begin{array}{r} 2 \ 1 \ 0 \\ 365 \\ \downarrow \quad \downarrow \quad \downarrow \\ (n/10^0) \% 10 \quad (n/10^1) \% 10 \quad n \% 10 \end{array}$

i<sup>th</sup>  
start from 0  
right to left

$$(n/10^i) \% 10$$

② Custom Comparator

int cmp(a, b){

$a == b \rightarrow \text{ret } 0; // \text{eq}$   
 $a > b \rightarrow \text{ret } +1; // \text{greater}$   
 $a < b \rightarrow \text{ret } -1; // \text{smaller}$

}

p2 Sort an integer array respect to k<sup>th</sup> digit of the number

range digits  
ex arr { 326, 18, 523 } K=0 { 523, 326, 18 }

ex arr { 362, 399, 318 } K=2 { 362, 318, 399 }

TCS ex arr { 361, 432, 12, 78, 500, 112, 365, 169 } K=1

SCS O(n\*d)

bucket sort  
0 → 500  
1 → 12, 112

arraylist 3 → 432

4

5 → 361, 365, 169  
6 → 78  
7 → 12, 112  
8 → 432  
9 → 399

radix  
sort

ex as { 361, 432, 12, 78, 500, 112 }

Quiz

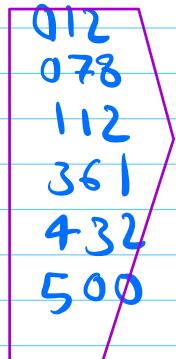
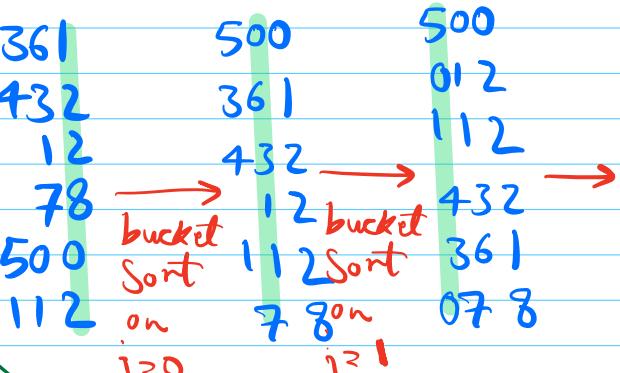
# of  
digits

$$TC: O(k \times (n+d)) =$$

$$SC: O(n+d) \rightarrow 0-9$$

vs

Quick  
Sort



Given an integer array, count the number of inversion pairs  
in the array.

what is inversion pair ??  $i < j$  and  $a[i] > a[j]$

ex  $a = \{8, 3, 4\}$   $\begin{array}{l} (8,3) \\ (8,4) \end{array}$  ans=2

Quiz  $\{4, 5, 1, 2, 6, 3\}$   $\begin{array}{l} (4,1) \\ (4,2) \\ (4,3) \end{array}$   $\begin{array}{l} (5,1) \\ (5,2) \\ (5,3) \end{array}$   $(6,3)$  ans=7

Quiz  $\{1, 2, 3, 4, 5, 6\}$   $\xrightarrow{\text{ans}=0}$

Quiz  $\{4, 4, 4, 4\}$   $\xrightarrow{\text{ans}=0}$

idea 1    for  $i = 0 \rightarrow n-1$   
              for  $j = i+1 \rightarrow n-1$   $a[i] > a[j]$  ans++       $O(n^2)$

idea 2

$b[n] \& c[m] \rightarrow a[m+n]$

int[] merge2(int b[], int c[]){

n = b.Len : i = 0

m = c.Len : j = 0

Tc: O(n+m)

a = new int[m+n]

for(k=0; k <= m+n-1; k++) {

if(i == n) // i >= n

| a[k] = c[j]  
| j++

ans += n-i + 1

}

else if(j == m) // j

| a[k] = b[i]

| i++;

}

else if(b[i] <= c[j])

| a[k] = b[i]

| i++;

}

else { // b[i] > c[j]

| a[k] = c[j]

| j++

ans += n-i + 1

}

ret a;

}

Inversion1	Inversion2
$a_8 \{ \begin{array}{cccccc cccccc} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ 9 & 8 & 7 & 3 & 6 & 4 & 1 & 5 & 0 & 10 \end{array} \}$	

9 8 7 3 6 | 4 1 5 0 10

9 8 | 7 3 6 | 4 1 5 0 10

9 | 8 | 7 | 3 6 | 4 | 1 | 5 | 0 10

↓ ↓      ↓      ↓      ↓      ↓

[8, 9]      [3, 6]      [1, 4]      [0, 10]

↓      ↓      ↓      ↓

[3, 6, 7]      [3, 6, 7]      [0, 5, 10]      [0, 5, 10]

↓      ↓      ↓      ↓

[3, 6, 7, 8, 9]      [0, 1, 4, 5, 10]

↓      ↓      ↓      ↓

[0, 1, 3, 4, 5, 6, 7, 8, 9, 10]

i-n

$\begin{matrix} [3, 6, 7] \\ i \end{matrix} \begin{matrix} [8, 9, 10] \\ j \end{matrix}$

Inversion1 + Inversion2 + cur\_Inversion