

STL: map $\begin{cases} \text{unordered} \\ \text{ordered} \end{cases}$ $\langle K, V \rangle$ pair (dictionary)
vector \rightarrow dynamic-array $\left\{ \begin{matrix} 2 & : & 8 \\ n & & n^3 \end{matrix} \right\}$

Data Structures & Algorithms \Rightarrow sorted array $\begin{matrix} \leftarrow N \rightarrow \\ \frac{N}{2} \quad \frac{N}{2} \\ \frac{N}{2}, \frac{N}{4}, \frac{N}{8}, \dots, \frac{N}{2^k} \quad k=0,1,2,3 \\ \frac{N}{2^k} \quad k=\log_2 N \end{matrix}$

Algos

Searching

Linear $O(n)$
Binary $O(\log n)$
Recursive Binary $O(\log n)$
Jump $\rightarrow \sqrt{n}$
Interpolation (pos)

Sorting

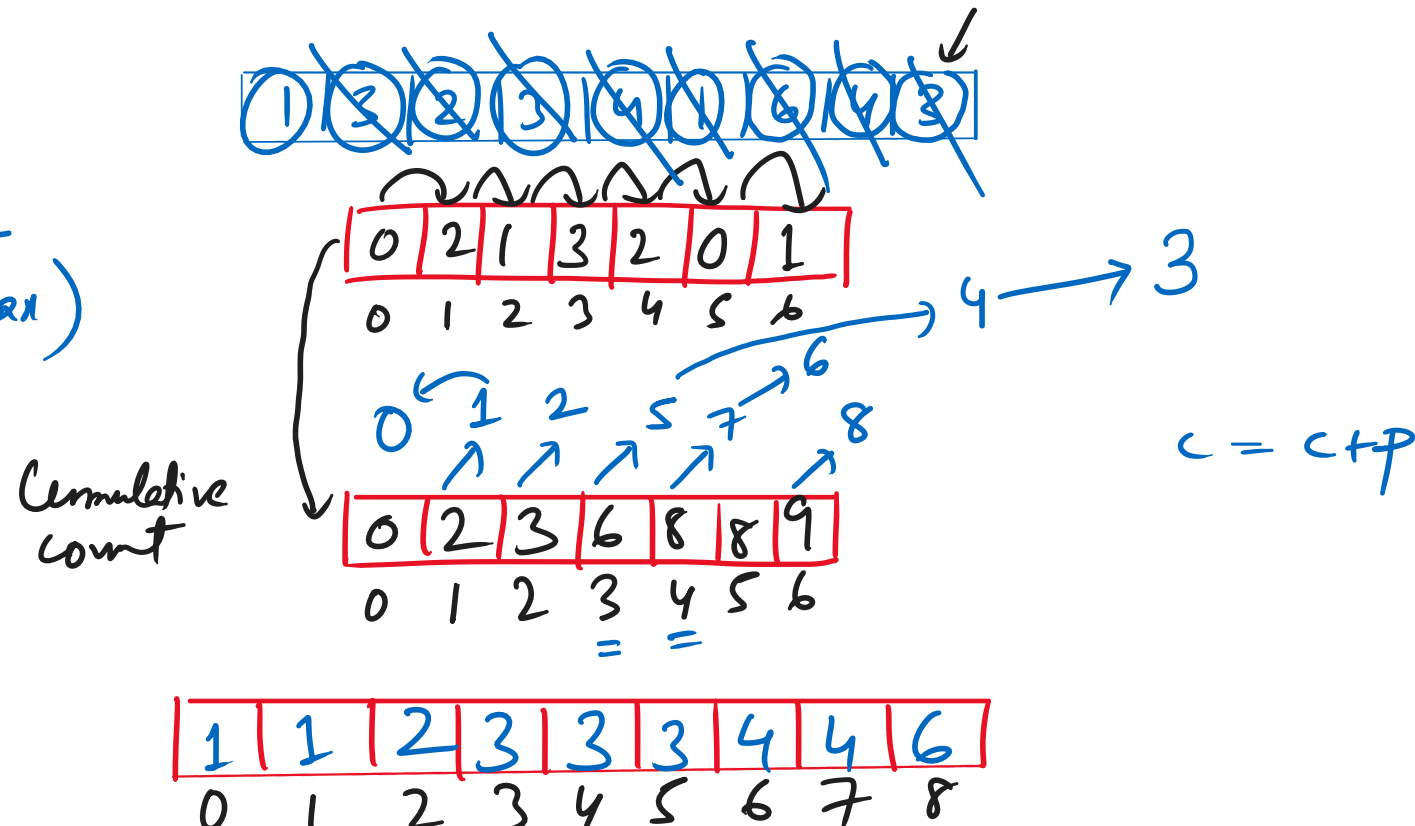
Bubble $O(n^2)$
Selection $O(n^2)$ swap
Insertion $O(n^2)$ shift
Merge $O(n \log n)$
Quick $O(n \log n)$
Heap $O(n \log n)$
Shell
A.M.L. \leftarrow Count Sort
D.S. \leftarrow Radix Sort
N.C.A.

Time Complexities

Special Algos

* Kadane's Algo
* Rabin Karp
* Euclid's Algo
GCD
* Sieve of Eratosthenes

max = 6
 $O(n + \max)$



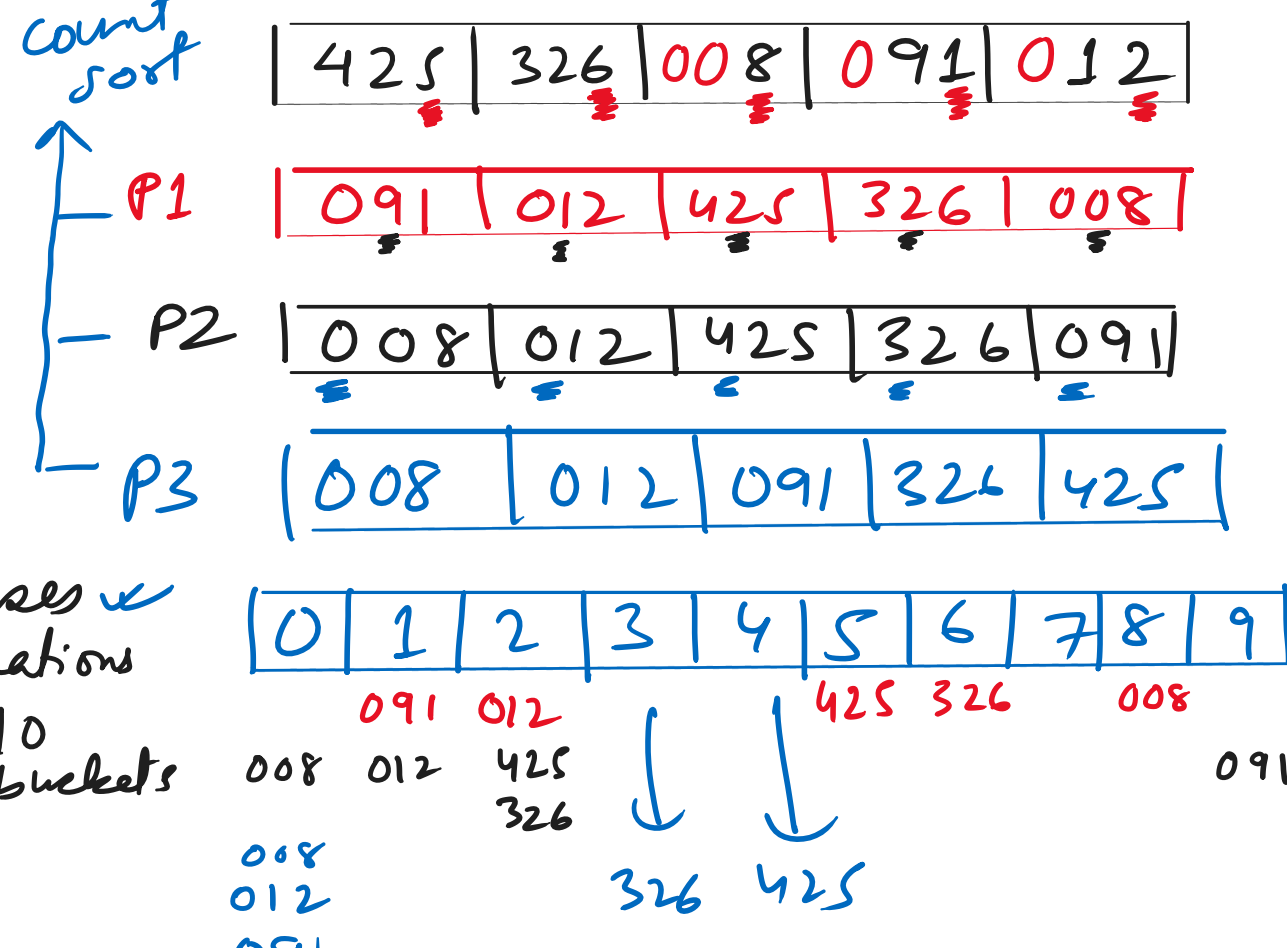
Radix Sort Algorithm \Rightarrow 1's, 10's, 100's, and so on

* Non-Comparison Algo
* Multi-digit nos
* Constant length strings
'rahul', 'samai', 'priya'

Steps: ① Find the max: 425
② Digits in max: 3 digits \rightarrow 3 passes

③ Bucket Sort \rightarrow (0-9) 10 buckets

$O(n + \max)$



* How do we control the number of (passes / iterations / rounds) in the radix sort algorithm?

\rightarrow The no. of passes is controlled by the no. of digits in the max value. 1, 2, 3 \rightarrow stop

Pseudo Code: for (int exp=1; max/exp > 0; exp *= 10) { count sort (arr, exp); }

* Binary Search \rightarrow ADA

* Recursion
* Heaps

Kadane's Algorithm \Rightarrow (Maximum Sub Array Sum)

int[] arr = { 5, -8, 1, 2, -1, 4 } = 6
int cmax = arr[0];
int gmax = arr[0];
for (int i=1; i<n; i++) {
cmax = max(arr[i], cmax+arr[i]);
gmax = max(cmax, gmax);
}
return gmax;

i	arr[i]	cmax+arr[i]	max c, arr[i]	mx cm gm
1	-8	5+(-8)	-3	5
2	1	1+(-3)	1	5
3	2	2+1	3	5
4	-1	-1+3	2	5
5	4	4+2	6	6

* Binary Search \Rightarrow $x \times x = n$

Square root of a number (Integral Part) $\rightarrow (\log N)$

0-3 sq-rt (int n) {

$3 \times 3 = 9$

ans = mid = 3

$5 \times 5 = 25$

ans = mid = 5

$6 \times 6 = 36 = n$

* very large \rightarrow left side
* very small \rightarrow store & move right

0-17

e = m-1

0-7

S = m+1

4-7

S = m+1

n = 36 \rightarrow 6

0-36

S

e = m-1

m = 8

6-7

13

2 = 6

m = 36

m = 18

m = 9

m = 6

m = 6

m = 6

m = 6

Important Binary Search Questions for Placements $\rightarrow \log n$

* Square root (Integral Part) using Binary Search

* Search in a 2D Matrix \rightarrow Left Code

* First / Last / Total Occurrences of an element in an array.

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

* Missing Element in an Array

* Aggressive Cows / Lions

* Painter's Partition Problem

Book Allocation Problem

k=3
f0=2
l0=4
t0 = l0-f0+1
= 4-2+1
= 3
1, 2, 3, 4, 3, 2, 1

* Peak in a Mountain