STL → Built in Function

Bits - stdc++.h

sort (arr, arr+n) → Merge sort O(n log N)

Quick sort

* Searching → find (...) → Binary Search

Search

cari(7)

1 2 3 4 5 6 7

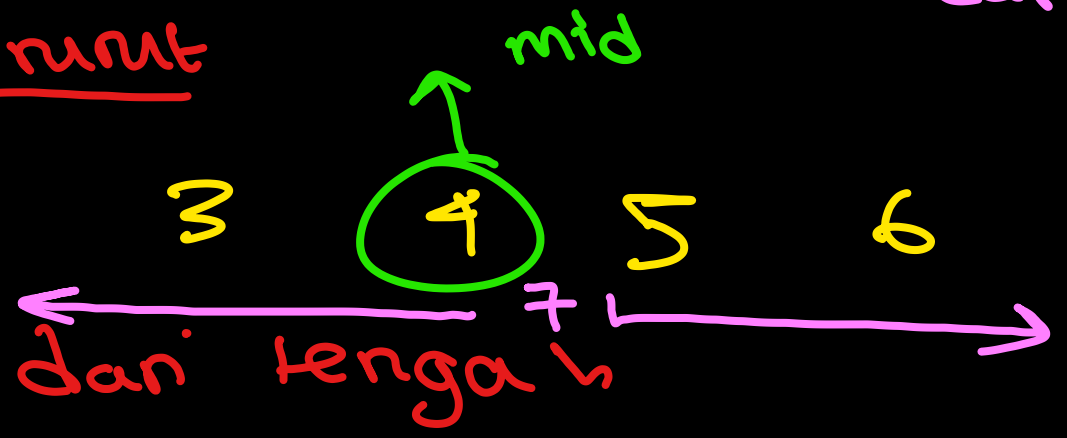
cara I:
1 ... 7 x
2 ... 7 x
3 ... 7 x
- - - -
7 ... 7 v → ketemu

N data → O(N)

Sequential Search

Binary Search

- Pastikan terurut

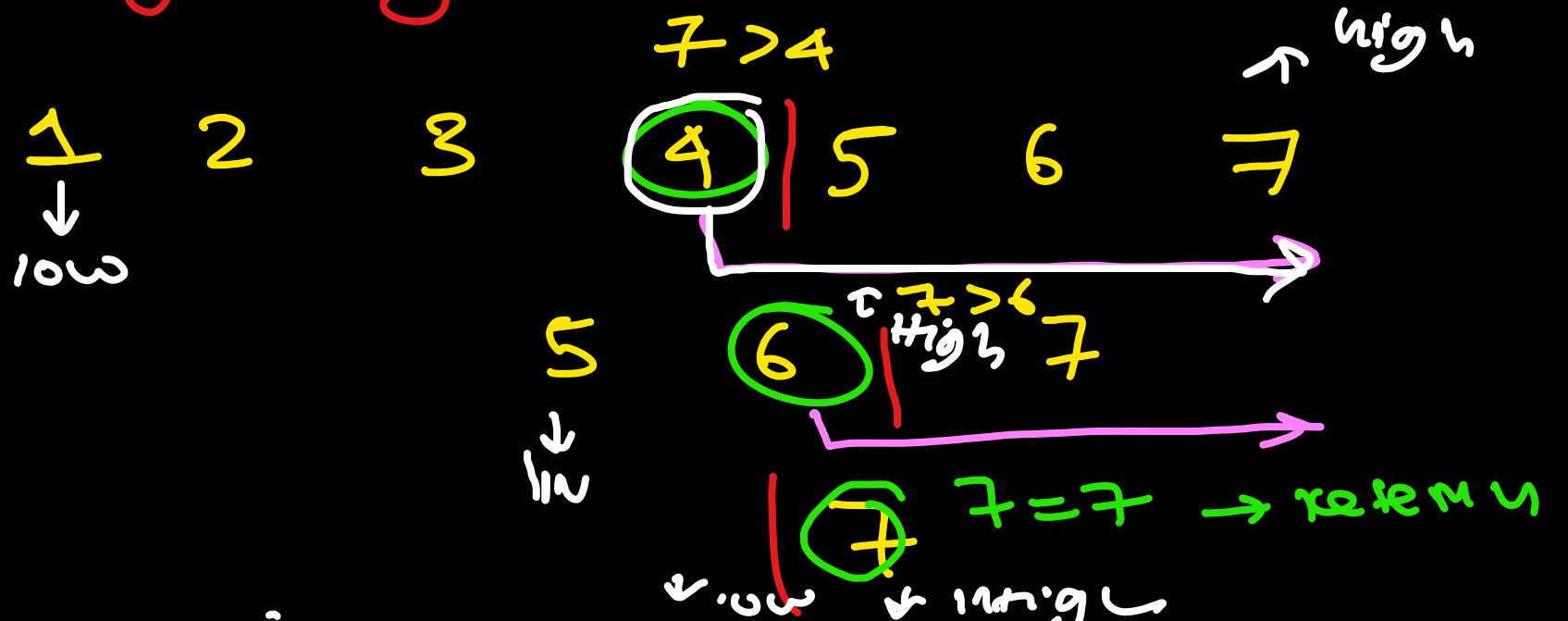


- Mulailah dari tengah
- Cek apakah data yang kita cari
  > Mid, < Mid, = Mid

  Mid = 4, cari = 7

  - data dicari > mid → ke kanan
  - data dicari < mid → ke kiri
  - data dicari = mid → ketemu

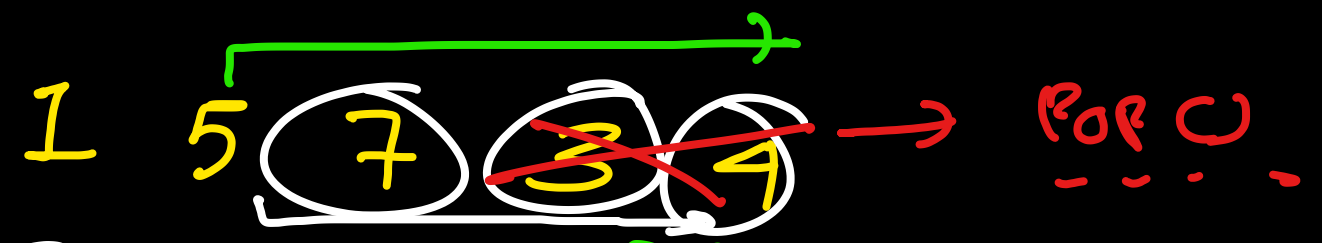
Ulangi langkah sampai data ditemukan



Mencari data dgn cara membandingkan Median & menentukan arah pencarian selanjutnya [kanan, kiri] → Binary search

$n/2/2/2/2 \rightarrow 0 \quad (\log N)$

Struktur data
- Primitif
- non primitif
  - array
    - static
    - dynamic



Pop ()

$B_0 = 1$

$B_1 = 5$

B_j = $j > i$

$B_i \geq B_j$ binary search → 1

Set $P_1 = \{\ \}$ → 1 kali

$P_5 = \{3, 4\}$ → 1 kali

$B_2 = 7$ =

$P_7 = \{\ \}$ → 1 kali

total = 5