

STL → Built in Function  
Bits - stdc++.h

sort(arr, arr+n) → Merge sort  $O(n \log n)$  /  
Quick sort

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

# Search

can(7)

1 2 3 4 5 6 7

can(1): 1 ... 7 x

2 ... 7 x

3 ... 7 x

...

7 ... 7 ✓ → return

$N$  data  $\rightarrow O(N)$

Sequential Search

# \* Binary Search

\* Pasukan terurut

cari 7

1 2 3 4 5 6 7

mid

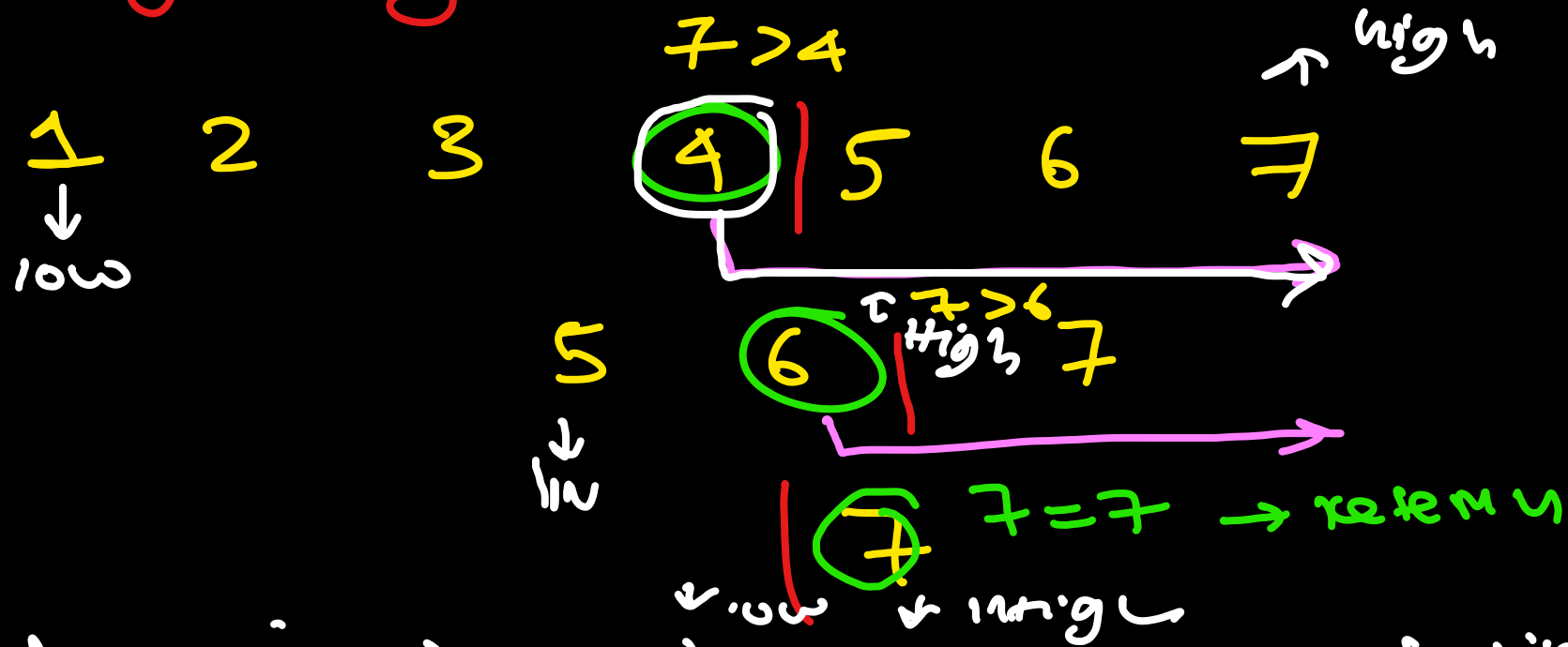
\* Mulai dari tengah

\* Cek apakah data yang kita cari  
 $> \text{mid}$ ,  $< \text{mid}$ ,  $= \text{mid}$

$\text{mid} = 4$ , cari = 7

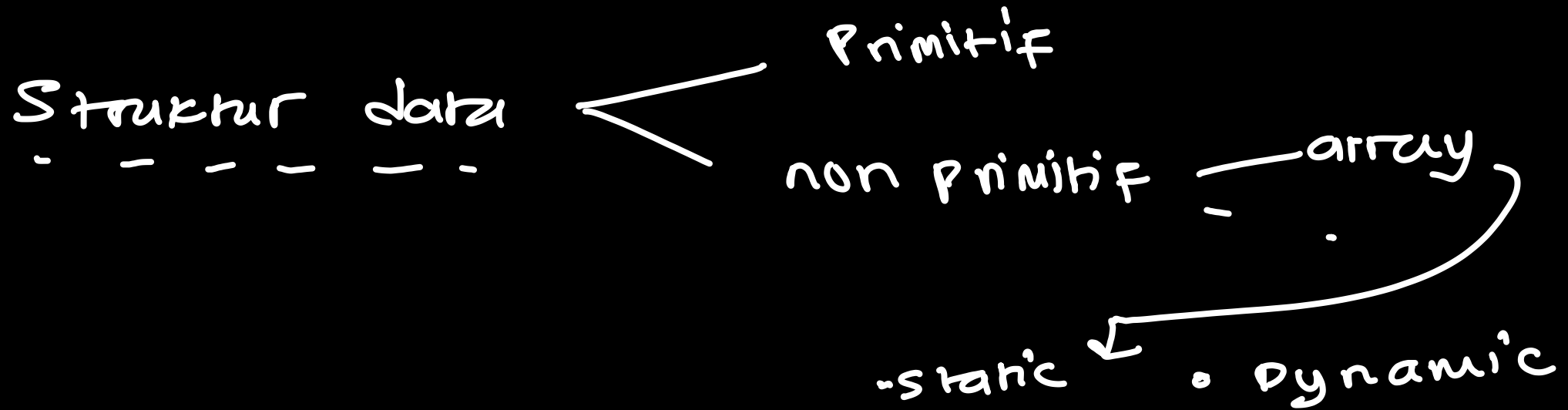
- data dicari  $> \text{mid}$  → ke kanan
- data dicari  $< \text{mid}$  → ke kiri
- data dicari  $= \text{mid}$  → ketemu

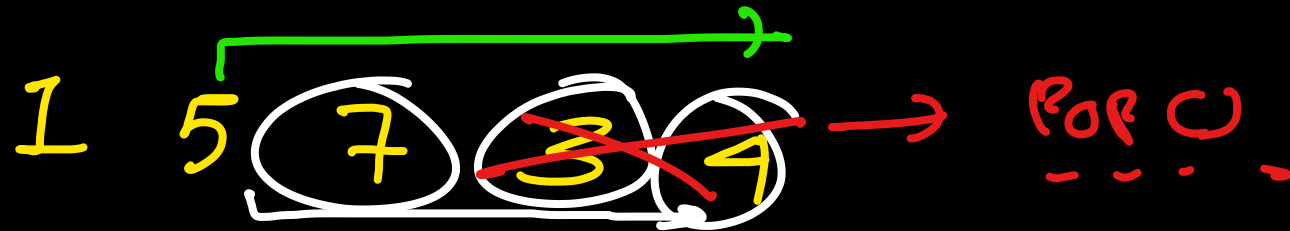
\* ulangi langkah sampai data ditemukan



Mencari data dgn cara Memerhatikan  
Median & Menentukan arah pencarian  
setelahnya [  $l_{\text{aron}}$ ,  $r_{\text{in}}$  ]  $\rightarrow$  Binary search

$12/2/2/2 \rightarrow 0 \quad ( \log N )$





$$B_0 = 1$$

$$B_j = j > i$$

$$B_1 = 5$$

$$B_i \geq B_j \quad \text{binary search} \rightarrow 1$$

Set  $P_1 = \{ \} \rightarrow 1 \text{ kai}$

$$P_5 = \{ 3, 4 \} \rightarrow 1 \text{ kai}$$

$$\text{total} = 5$$

$$- - - - -$$

$$B_2 = 7 =$$

$$P_7 = \{ \} \rightarrow 1 \text{ kai}$$

1 2 3 4

Start(4)  
1 → x

### \* Seq search

For (i = 1 → x) :

if (i == x) :

return i


52  
10 →

}  
Brute force /  
Seq. search

★ Binary search

sort (9)

1 2 3 4 5 6 7 8 9



$5^2 == 9$  x

$5^2 > 9$  ✓

1 2 3 4



1 2 3 4 5 6 7 8

$$4 \times 4 > 8$$

1 2 3

1 . . . 17

1 . . . 7  $8^2 > 17$

1

3

mid

7

$$3^2 > 7$$

$$(3-1)^2 \leq 7$$

low bound

$$2^2 \leq 7$$

$x \leq 1$

$(x+1)^2 > n$

1, 2, 3











































