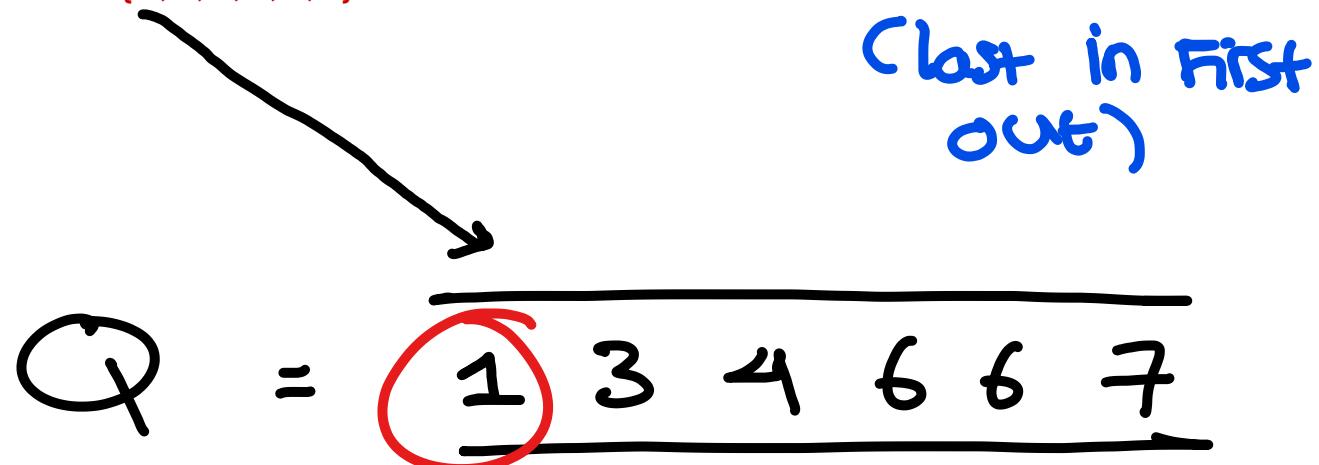


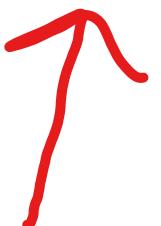
Struktur Data Non Linear + PQ

PQ, Set, HashMap, dll

$\text{arr} = \{1, 3, 4, 6, 6, 7\}$



Stack



Queue



Linear

Stack - LIFO
(last in first out)

Queue - FIFO
(First in First out)

Queue

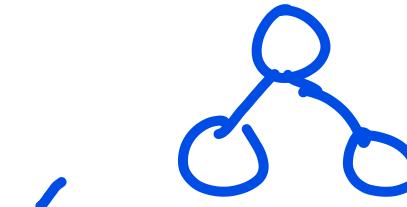
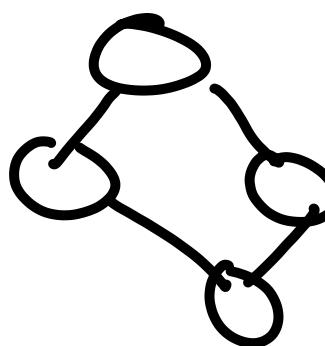
POP \Rightarrow First element

PUSH \Rightarrow Last element

Last - in

Last - out

Non Linear \Rightarrow



P.Q

arr = [9, 3, 6, 5, 4, 2]

$3 < 9$

min

max

9

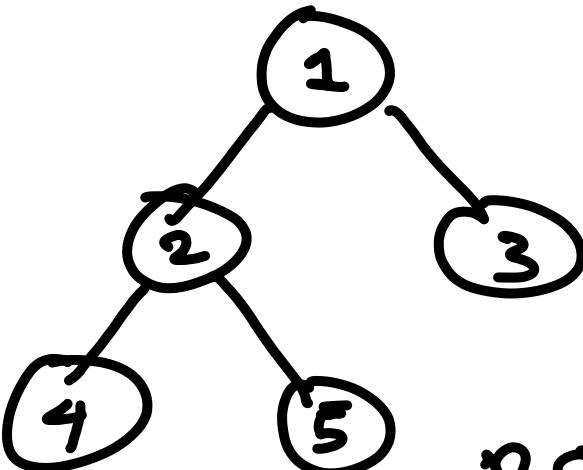
3 9

X 8 X 5 X 4

3 6 9

BFS: 1 - 2 - 3 - 4 - 5

Memprioritaskan
Node dengan
nilai paling kecil
dulu.



1 2 3 4 5

BFS C1)

Mark
Visited

$$PQ = \left[\begin{array}{c} \cancel{1} \\ PQ \cdot POPC \end{array} \right] = \begin{array}{c} \cancel{1} \\ 2, 3 \end{array}$$

PQ. Front + 1

$$PQ = \left[\cancel{2}, 3 \right]$$

$$PQ = [3]$$

BFS C2)

Push(2)
Push(3)

$$con(2) = \{1, 5\}$$

Push(4)
Push(5)

BFS C3)

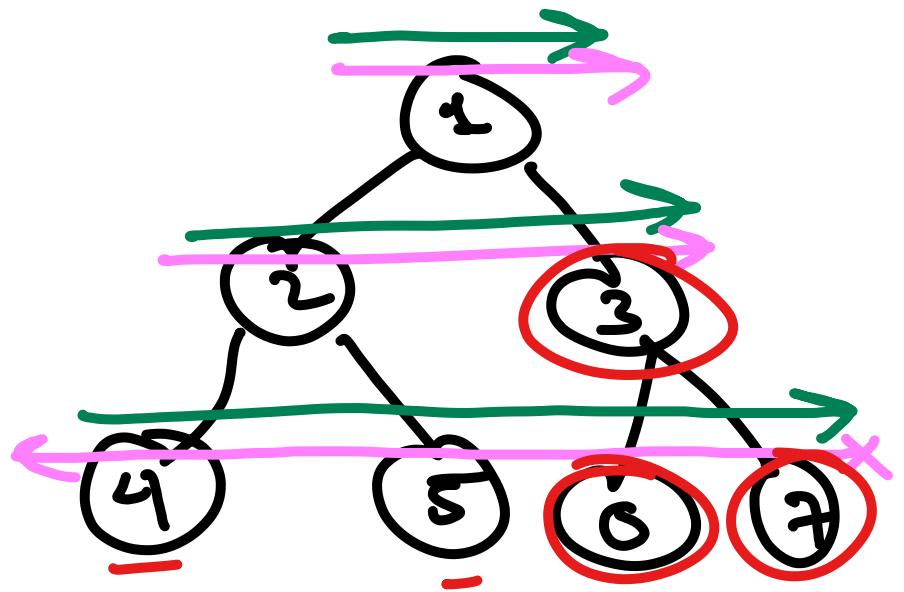
$$PQ = \left[\begin{array}{c} \cancel{3}, 4, 5 \\ PQ = [1, 5] \end{array} \right]$$

$$con(3) = \times$$

BFS C4)

$$PQ = [5]$$

BFS C5)



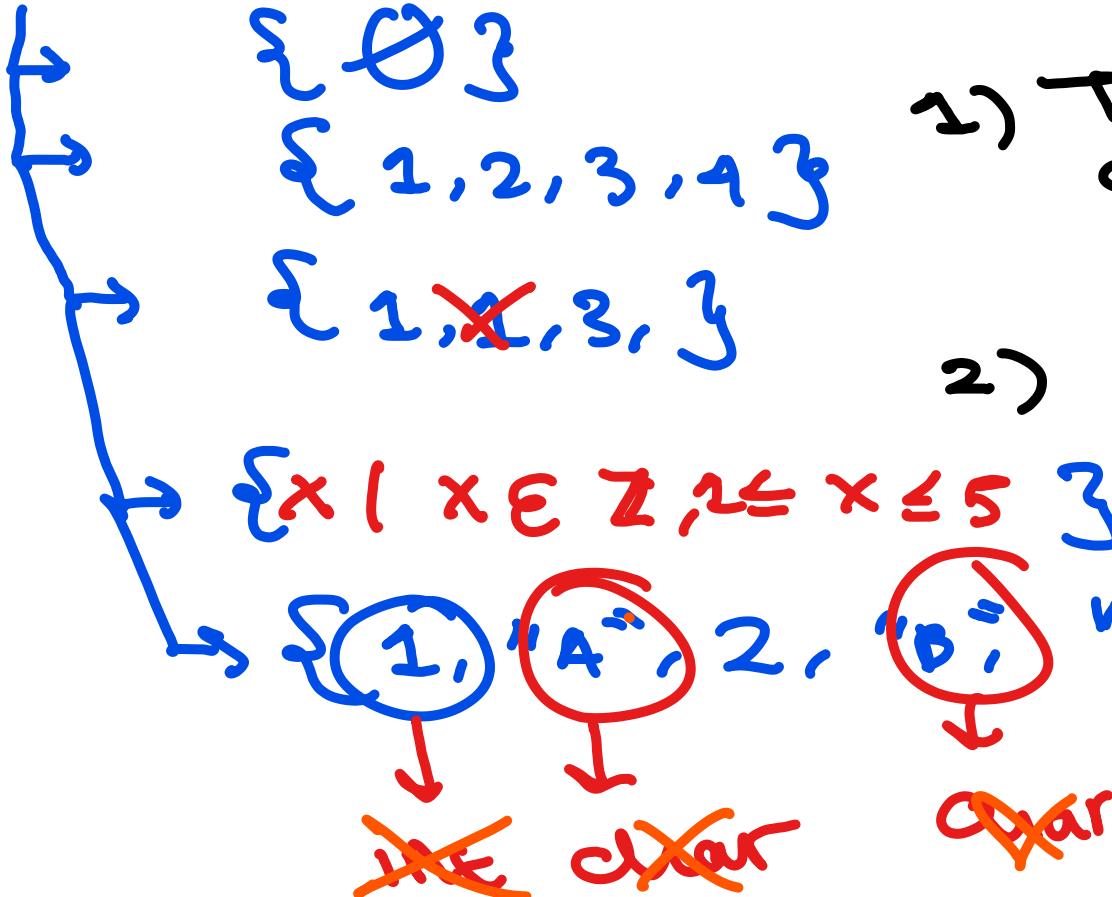
$BFS = 1 - 2 - 3 - 4 - 5 - 6 - 7$

$BFS = 1 - 2 - 3 - \underline{7} - \underline{6} - \underline{5} -$

$Q = [3, 2]$

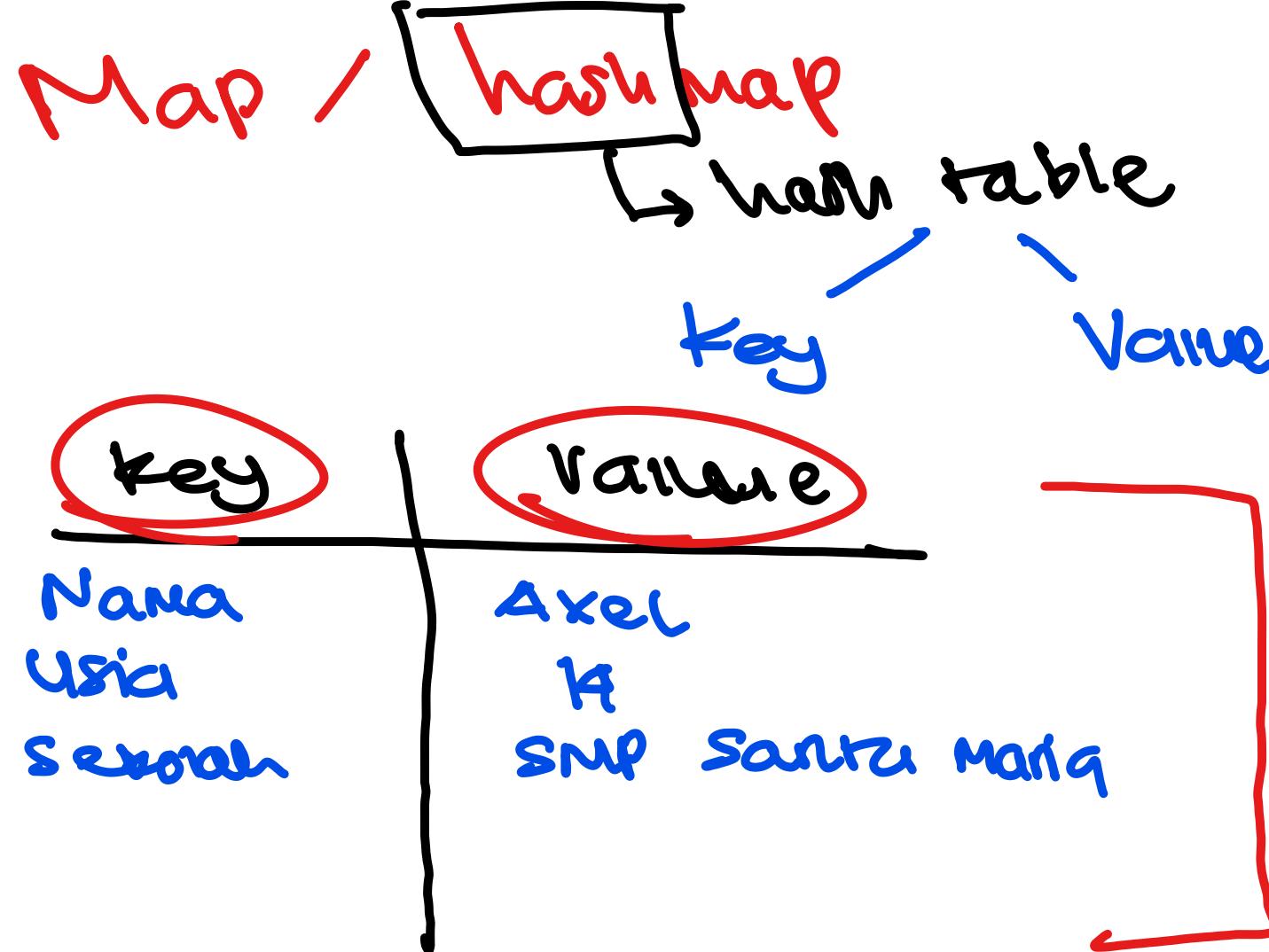
SET \Rightarrow himpunan $\Rightarrow \{ \dots \dots \}$

unique value
same type



- 1) Tidak boleh ada elemen sama
- 2) Tidak boleh elemen tipe nya berbeda

Multi-SET \Rightarrow Boleh ada elemen Yang berulang



data["Nana"]
↓
Axel

Nilai Siswa Fez

key	Value	N
		1
Axel	100	1
Revan	100	2
Bintang	100	3
Bejo	90	4

map

Nilai ["Axel"] → 100

key ⇒ {"Axel", "Revan", "Bintang", "Bejo"}

Value ⇒ {100, 100, 100, 90}

array biasa

nama_siswa = ["Axel", "Revan", ...]
nilai = [100, 100]

for

// Nilai Axel
idx = nama_siswa.find_index("Axel")
cout << nilai[idx]

