



# **PENGURUTAN (SORTING)**

**Dosen : Sulistyowati, ST., M.Kom.**

# Metode Sorting (4) – QUICK SORT

- **Quick sort** adalah proses sorting dengan cara menentukan pivot, kemudian menentukan penunjuk depan/PD (mulai dari data pertama) dan penunjuk belakang/PB (mulai dari data n). Kemudian gerakkan PD ke kanan atau ke data berikutnya sampai menemukan data yang lebih besar atau sama dengan pivot serta gerakkan PB ke kiri atau ke data sebelumnya sampai menemukan data yang lebih kecil atau sama dengan pivot. Lalu tukar tempat PD dan PB. Ulangi langkah tersebut sampai PD dan PB bersilangan.

# Metode Sorting (4) – QUICK SORT

## Algoritma :

Langkah 1 : Tentukan pivot =  $(L + R) \div 2$ , dengan L=posisi data awal dan R=posisi data akhir.

Langkah 2 : Bandingkan data pertama yang ditunjuk oleh PD (menuju ke belakang/data berikutnya) dengan pivot sampai ketemu data yang  $\geq$  pivot dan berhenti.

Bandingkan data terakhir yang ditunjuk oleh PB (menuju ke depan/data sebelumnya) dengan pivot sampai ketemu data yang  $\leq$  pivot dan berhenti.

Lalu tukar tempat PD dengan PB.

Langkah 3 : Ulangi langkah 2 sampai PD dan PB bersilangan

Langkah 4 : Ulangi langkah 1 sampai 3 dengan ketentuan :

~ jika  $L < PB$ , maka  $L$ =tetap dan  $R=PB$

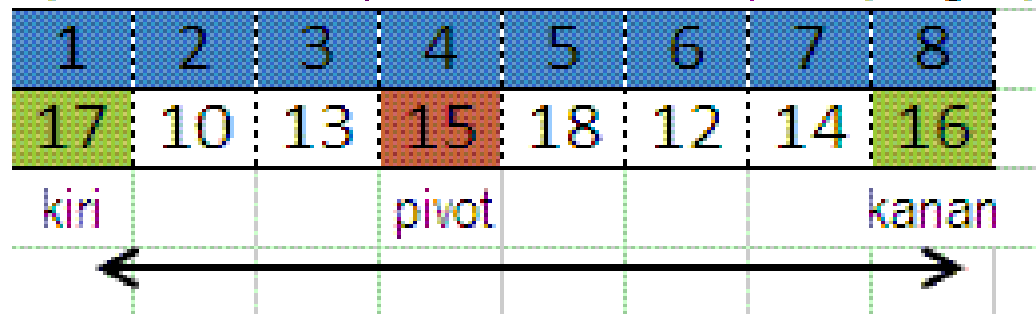
~ jika  $R > PD$ , maka  $L=PD$  dan  $R$ =tetap

# Metode Sorting (4) – QUICK SORT

- Contoh :


1) Menentukan posisi kiri, kanan, pivot (tengah)

1	2	3	4	5	6	7	8
17	10	13	15	18	12	14	16
kiri			pivot				kanan



2) posisi kiri bergerak ke kanan, posisi yg kanan bergerak ke kiri, berhenti jika  $\text{data}[\text{kiri}] \geq \text{data}[\text{pivot}]$  dan  $\text{data}[\text{kanan}] \leq \text{data}[\text{pivot}]$


1	2	3	4	5	6	7	8
17	10	13	15	18	12	14	16
kiri			pivot			kanan	



## Metode Sorting (4) – QUICK SORT

3) Tukar data[kiri] dan data[kanan]

1	2	3	4	5	6	7	8
14	10	13	15	18	12	17	16
kiri			pivot			kanan	



4)  $kiri = kiri + 1$ ,  $kanan = kanan - 1$

1	2	3	4	5	6	7	8
14	10	13	15	18	12	17	16
	kiri		pivot		kanan		



## Metode Sorting (4) – QUICK SORT

5) bergerak lagi							
1	2	3	4	5	6	7	8
14	10	13	15	18	12	17	16
			pivot		kanan		
			kiri	↔			

6) tukar lagi							
1	2	3	4	5	6	7	8
14	10	13	12	18	15	17	16
			pivot		kanan		
			kiri	↔			

## Metode Sorting (4) – QUICK SORT

7) bergerak lagi, kiri=kiri+1, kanan=kanan+1

1	2	3	4	5	6	7	8
14	10	13	12	18	15	17	16
			pivot	kiri			
				kanan			

8) berhenti karena, karena kiri > kanan

1	2	3	4	5	6	7	8
14	10	13	12	18	15	17	16
			pivot	kiri			
			kanan				



Proses pengurutan berhenti bila pointer kiri overlap dengan pointer kanan (langkah 8 di gambar atas), sekaligus membagi (divide) 2 bagian yang akan diurutkan selanjutnya; yaitu partisi kiri dan kanan.

# Metode Sorting (4) – QUICK SORT

Batas kanan dan kiri yang overlap dibagi 2 partisi

\* semua data < 15

14 10 13 12

\* semua data >= 15

18 15 17 16

1) Tiap partisi, diproses rekursif spt di atas

Partisi 1

14 10 13 12

kiri pivot kanan



Partisi 2

18 15 17 16

kiri pivot kanan



2) kanan bergerak untuk mencari data[kanan] <= pivot

14 10 13 12

kiri pivot

kanan



18 15 17 16

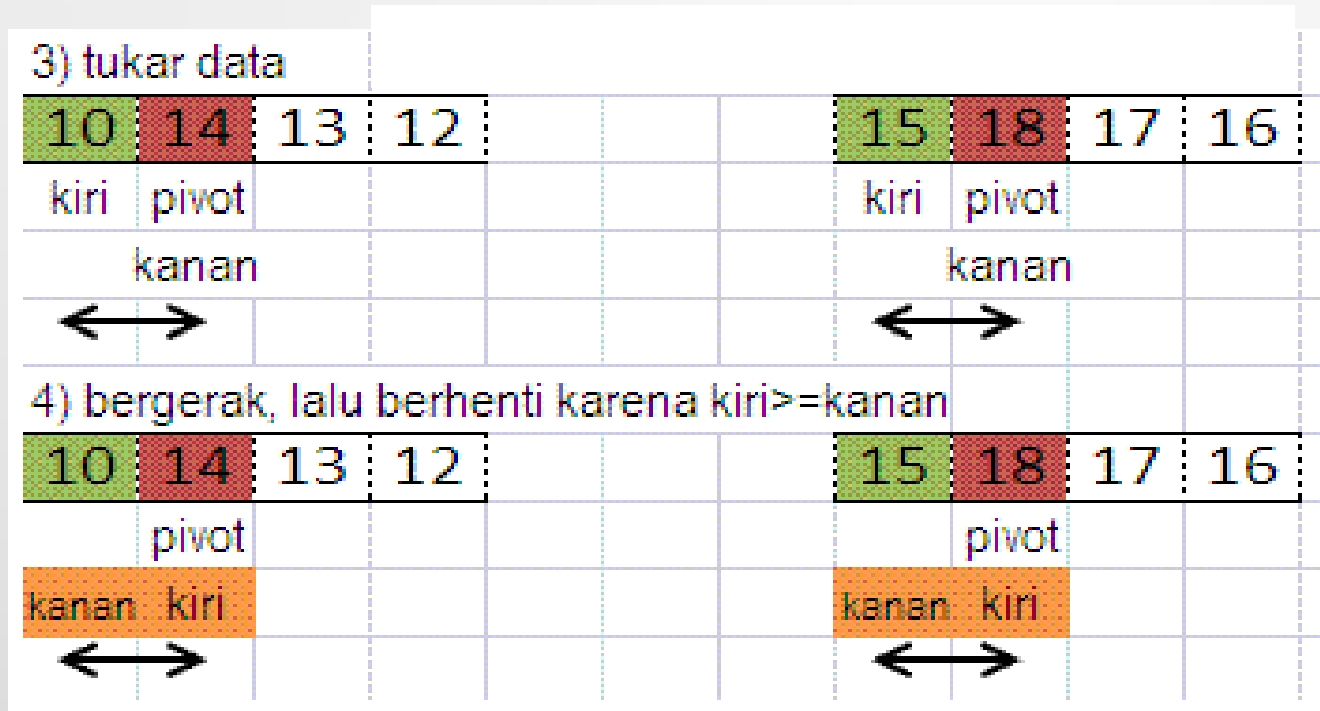
kiri pivot

kanan





## Metode Sorting (4) – QUICK SORT



Proses pengurutan dilakukan sama dengan langkah sebelumnya (rekursif) dan dilakukan pada partisi kiri dan kanan. Pembagian partisi berhenti bila tiap partisi hanya menyisakan satu elemen data saja (lihat warna hijau pada langkah 4 di atas).

# Metode Sorting (4) – QUICK SORT

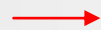
1) Partisi 1.1 dan Partisi 1.2				4) Partisi 2.1 dan Partisi 2.2			
stop	* semua data $\geq 10$			stop	* semua data $\geq 15$		
1	2	3	4	1	2	3	4
10	14	13	12	15	18	17	16
		pivot				pivot	
2) Lakukan partisi sampai berurutan							
1	1	1	1	1	1	1	1
10	12	13	14	15	16	17	18
3) lalu gabungkan semuanya							
1	2	3	4	5	6	7	8
10	12	13	14	15	16	17	18

# Metode Sorting (5) – RADIX SORT

- **Radix Sort** merupakan adalah proses sorting dengan cara mengklasifikasikan data (desimal) sesuai dengan posisi angkanya. Pada sistem desimal, radix adalah digit dalam angka desimal.
- Pertama kali mengurutkan nilai-nilai berdasarkan radix pertamanya (satuan), lalu pengurutan dilakukan berdasarkan radix keduanya (puluhan), dan begitu seterusnya.

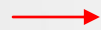
# Metode Sorting (5) – RADIX SORT

Data awal

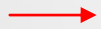


43 27 31 15 37 80 03

Pengkategorian  
data berdasar  
digit satuan

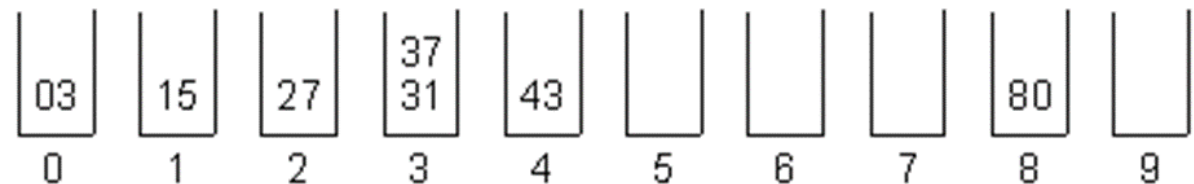
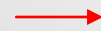


Data hasil iterasi

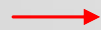


80 31 43 03 15 27 37

Pengkategorian  
data berdasar  
digit puluhan



Data hasil iterasi



03 15 27 31 37 43 80

# Metode Sorting (6) – MERGE SORT

- **Merge sort** adalah sorting dengan cara membagi (split) data menjadi dua bagian sama besar. Masing-masing bagian data diurutkan secara rekursif, dan kemudian digabungkan kembali untuk membentuk bagian data yang terurut.
- Merge sort menggunakan teknik *divide* dan *conquer*, yang tahapannya sbb :
  1. *Divide* : data dibagi menjadi 2 bagian. Jika pembagiannya masih terlalu besar, maka bagian tersebut dibagi lagi menjadi dua bagian yang lebih kecil, dst sampai diperoleh data tunggal.
  2. *Recursive*: masing-masing bagian diurutkan secara rekursif
  3. *Conquer* : setelah diurutkan, masing-masing bagian digabungkan dan diurutkan sehingga menjadi satu bagian data yang utuh

# Metode Sorting (6) – MERGE SORT

- Contoh :

