Asia Cyber University

PERTEMUAN 14

Struktur Data & Algoritma

SORTING

Beberapa Metode *Sorting*

No.	Sorting Dasar
1	Bubble Sort
2	Insertion Sort
3	Selection Sort

No.	Sorting Lanjutan
1	Merge Sort
2	Quick Sort
3	Bucket Sort
4	Shell Sort
5	Radix Sort
6	External Sort

Metode Merge Sort

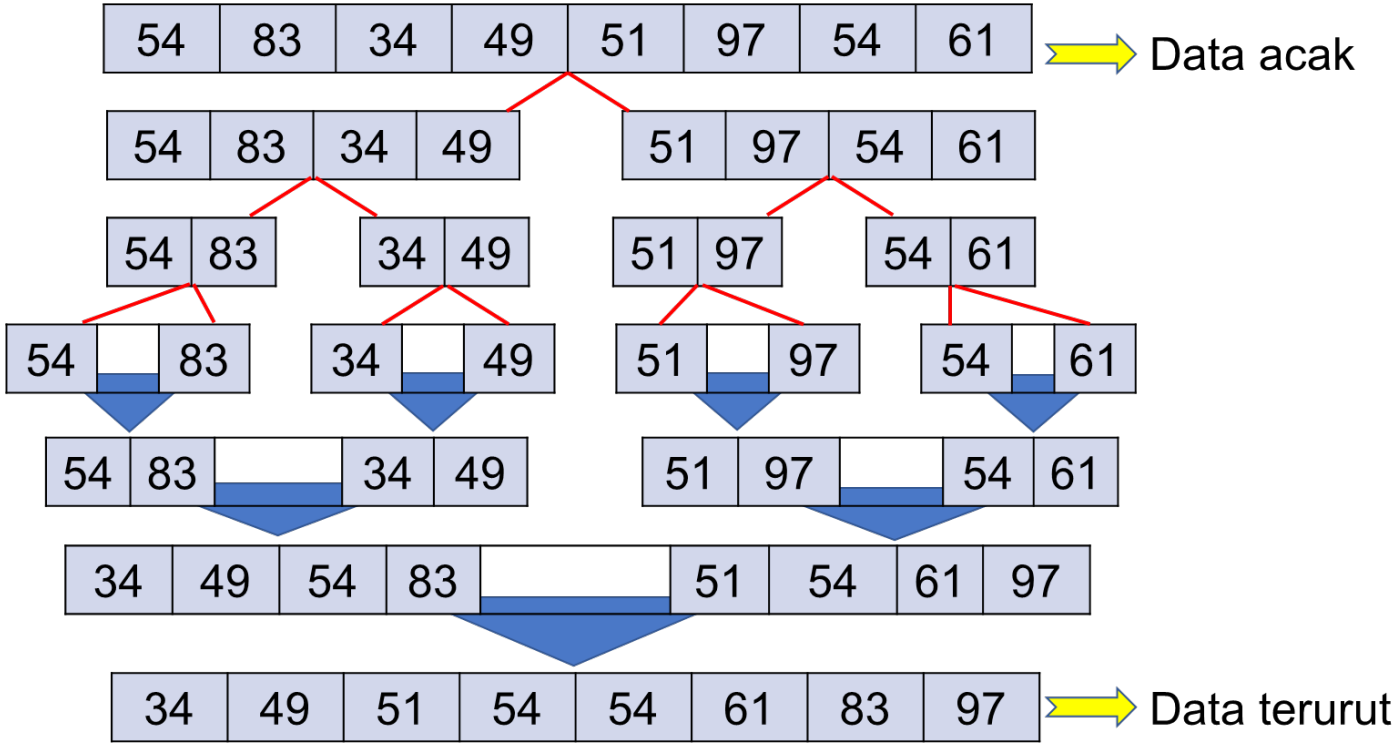
Metode Merge Sort merupakan proses pembagian (*divide*) dan penggabungan (*merge*) setiap elemen yang ada.

Prinsip kerja Metode Merge Sort:

1) Membagi semua elemen array menjadi dua bagian hingga semua sub-array berisi satu elemen.
2) Menggabungkan kembali semua elemen dengan diikuti proses membandingkan elemen-elemen yang ada untuk diurutkan.
3) Lakukan proses di atas hingga semua elemen terurut.

Metode Merge Sort

Contoh:



Kelebihan: penggunannya lebih efisien dibandingkan dengan metode lain.

Kekurangan: membutuhkan setidaknya ruang memori dua kali lebih besar karena dilakukan berulang kali.

Metode Quick Sort

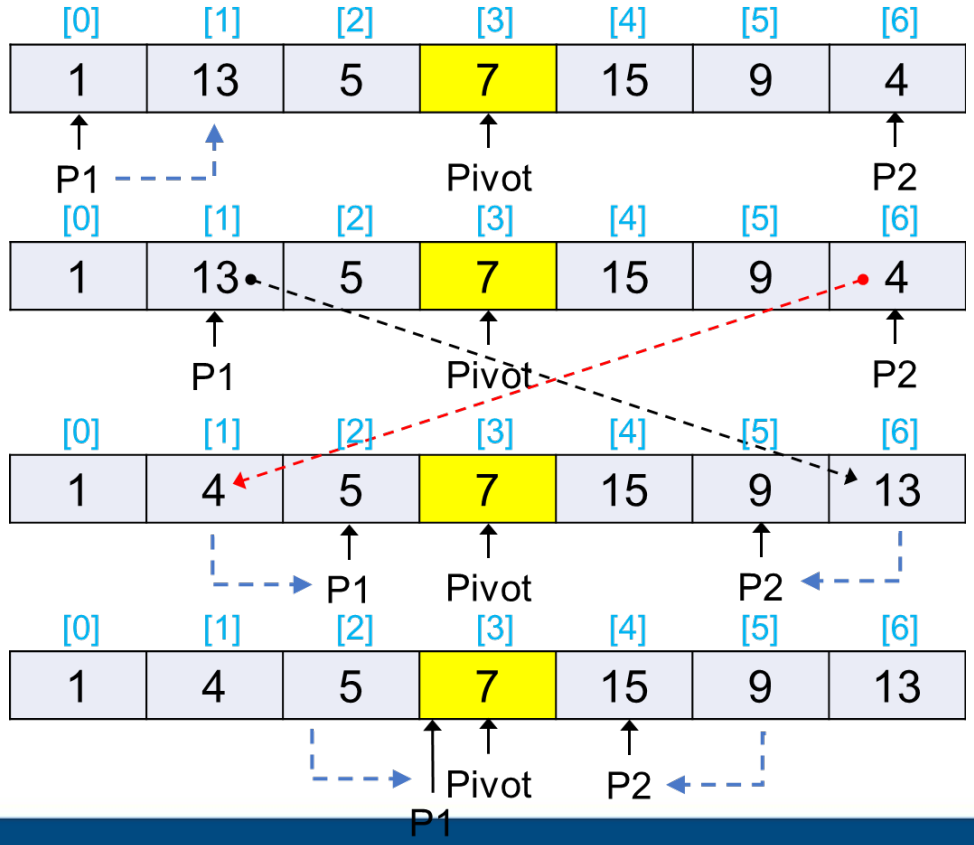
Metode Quick Sort dikenal juga sebagai Metode Partisi (*Partition Exchange Sort*). Teknik dari metode ini adalah dengan menukarkan dua elemen dengan jarak yang cukup besar dan tipe penyelesaian *divide and conquer*, dimana cara pengurutannya dengan membagi (*divide*) seluruh elemen menjadi beberapa sub-elemen, sehingga permasalahan menjadi lebih sederhana untuk dipecahkan (*conquer*).

Langkah-Langkah Metode Quick Sort:

1. Pilih nilai index (bebas) sebagai Pivot.
2. Ambil dua variable, Pointer kiri (P1) ditempatkan di data pertama dan Pointer kanan (P2) ditempatkan di data terakhir dari Array. P1 untuk nilai rendah dan P2 untuk nilai tinggi.
3. Jika data P1 < Pivot ✯ pindahkan P1 ke kanan.
4. Jika data P2 > Pivot ✯ pindahkan P2 ke kiri.
5. Ulangi langkah 3 dan 4, jika P1 dan P2 keduanya tetap (tidak dapat dipindahkan lagi) ✯ data ditukar.
6. Pindahkan P1 ke data pertama dan P2 ke data terakhir, lalu ulangi langkah 3 - 5.
7. Jika data P1 ≥ data P2 (titik dimana mereka bertemu) ✯ tentukan Pivot baru.

Metode Quick Sort

Contoh: *Ascending*



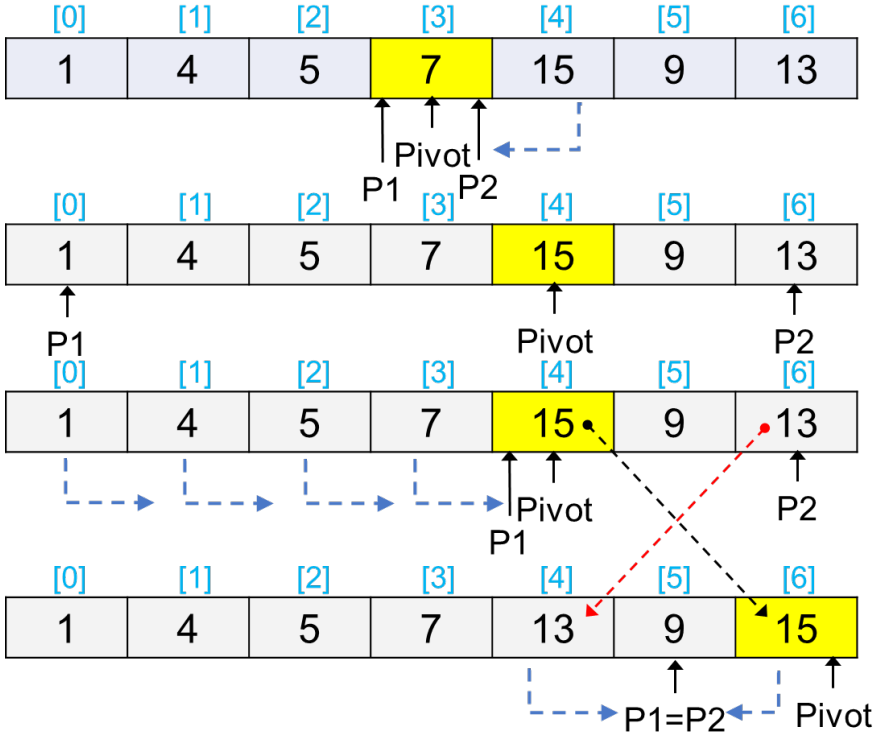
Pivot[3] = 7
data P1 < Pivot ➔ pindahkan P1 ke kanan
data P2 > Pivot, tidak cocok ➔ P2 tetap

data P1 < Pivot, tidak cocok ➔ P1 tetap
data P2 > Pivot, tidak cocok ➔ P2 tetap
proses P1 & P2 tidak cocok ➔ data ditukar
pindahkan P1 ke kanan & P2 ke kiri

data P1 < Pivot ➔ pindahkan P1 ke kanan
data P2 > Pivot ➔ pindahkan P2 ke kiri

data P1 < Pivot, tidak cocok ➔ P1 tetap
data P2 > Pivot ➔ pindahkan P2 ke kiri

Metode Quick Sort



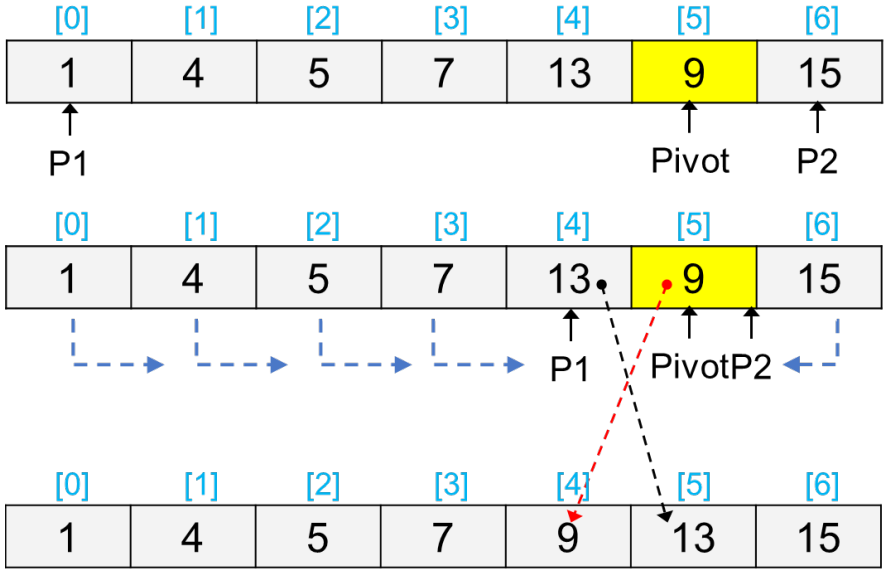
data P1 ≥ data P2 ➔ tentukan Pivot baru
P1 dipindahkan ke data pertama dan
P2 dipindahkan ke data terakhir

Pivot[4]=15
data P1 < Pivot ➔ pindahkan P1 ke kanan
data P2 > Pivot, tidak cocok ➔ P2 tetap

data P1 < Pivot ➔ pindahkan P1 ke kanan,
….hingga elemen 15
data P1 < Pivot, tidak cocok ➔ P1 tetap
data P2 > Pivot, tidak cocok ➔ P2 tetap
proses P1 & P2 tidak cocok ➔ data ditukar

data P1 ≥ data P2 ➔ tentukan Pivot baru
P1 dipindahkan ke data pertama dan
P2 dipindahkan ke data terakhir

Metode Quick Sort



Pivot[5] = 9
data P1 < Pivot ➔ pindahkan P1 ke kanan
data P2 > Pivot ➔ pindahkan P2 ke kiri

data P1 < Pivot ➔ pindahkan P1 ke kanan …hingga data 13
data P2 > Pivot, tidak cocok ➔ P2 tetap

data P1 < Pivot, tidak cocok ➔ P1 tetap
data P2 > Pivot, tidak cocok ➔ P2 tetap
proses P1 & P2 tidak cocok ➔ data ditukar

⇒ Data terurut

Metode Shell Sort

Metode Shell Sort dikenal juga sebagai Metode Pertambahan Menurun (*Diminishing Increment Sort*). Teknik dari metode ini adalah dengan membandingkan suatu data dengan data lain yang memiliki jarak tertentu (teratur) sehingga seperti membentuk sublist, lalu dilakukan penukaran bila diperlukan.

Jarak yang dipakai didasarkan pada *increment value* (nilai penambahan) / *sequence number*. Setiap sublist berisi elemen ke-i dari kumpulan elemen yang asli.

Metode Shell Sort

Misalkan sequence number yang dipakai adalah 4 ,2, 1, maka sublistnya sbb:

Untuk i = 4 → S[0],S[4],S[8],…
S[1],S[5],S[9],…
S[2],S[6],S[10],…dan seterusnya

Untuk i = 2 → S[0],S[2],S[4],…
S[1],S[3],S[5],…dan seterusnya

Untuk i = 1 → S[0],S[1],S[2],S[3],…dan seterusnya

Metode Shell Sort

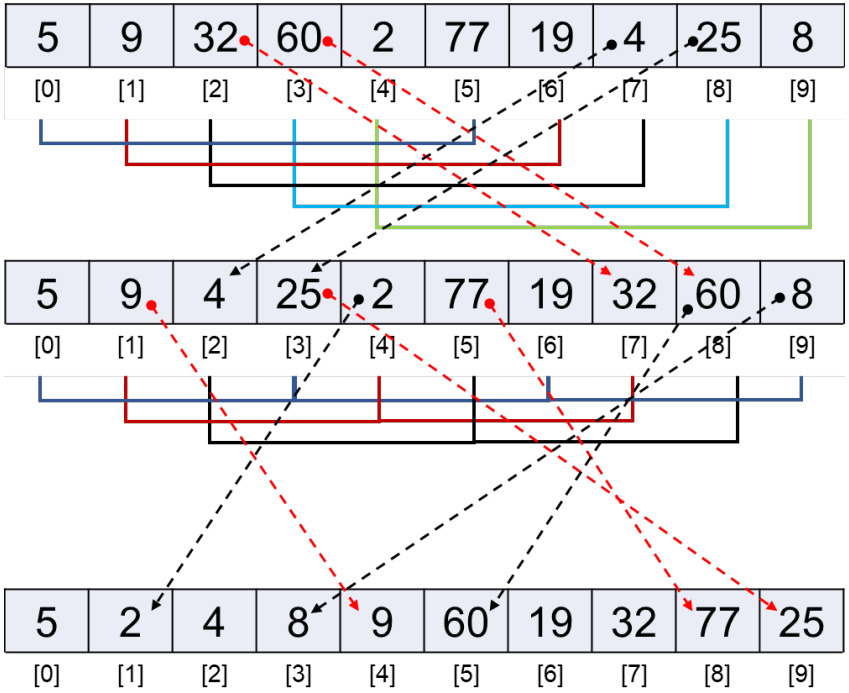
Pemilihan Sequence Number:

1. Disarankan jarak mula mula dari data adalah N/2.
2. Proses berikutnya N/4, lalu N/8 dan seterusnya, sampai jarak yang digunakan adalah 1.

Proses pengurutan Shell Sort:

3. Buat sublist yang didasarkan pada jarak sequence number yang dipilih.
4. Urutkan elemen dari setiap sublist tersebut.
5. Gabungkan semua sublist.

Contoh: *sequence number* 5, 3, 1.



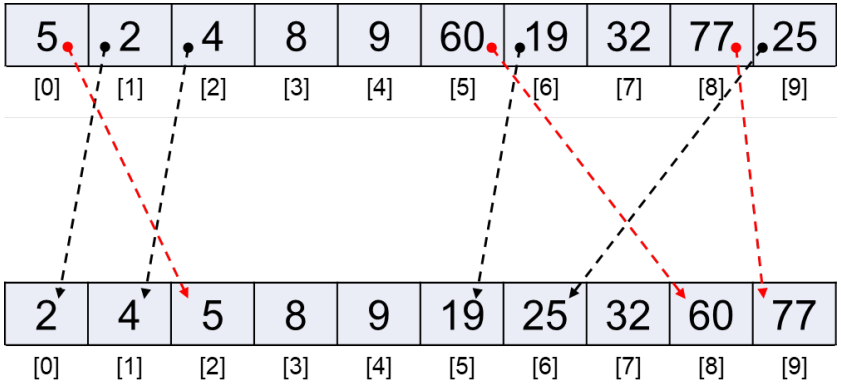
Sublist i = 5, urutkan dan gabungkan :
S[0],S[5] : 5 < 77 = true
S[1],S[6] : 9 < 19 = true
S[2],S[7] : 32 < 4 = false ➔ data ditukar
S[3],S[8] : 60 < 25 = false ➔ data ditukar
S[4],S[9] : 2 < 8 = true

Sublist i = 3, urutkan dan gabungkan :
S[0],S[3],S[6],S[9] : 5 < 25 < 19 < 8 = false ➔ data ditukar

S[1],S[4],S[7] : 9 < 2 < 32 = false ➔ data ditukar

S[2],S[5],S[8] : 4 < 77 < 60 = false ➔ data ditukar

Contoh: sequence number 5, 3, 1.



Sublist i = 1, urutkan :
S[0],S[1],…,S[9] : false ➔ data ditukar

data terurut

SELESAI