



# ALGORITMA dan STRUKTUR DATA

## PERTEMUAN KE - 10

**ENDANG SRI RAHAYU**

**Teknik Elektro – Fakultas Teknologi Industri – Universitas Jayabaya**

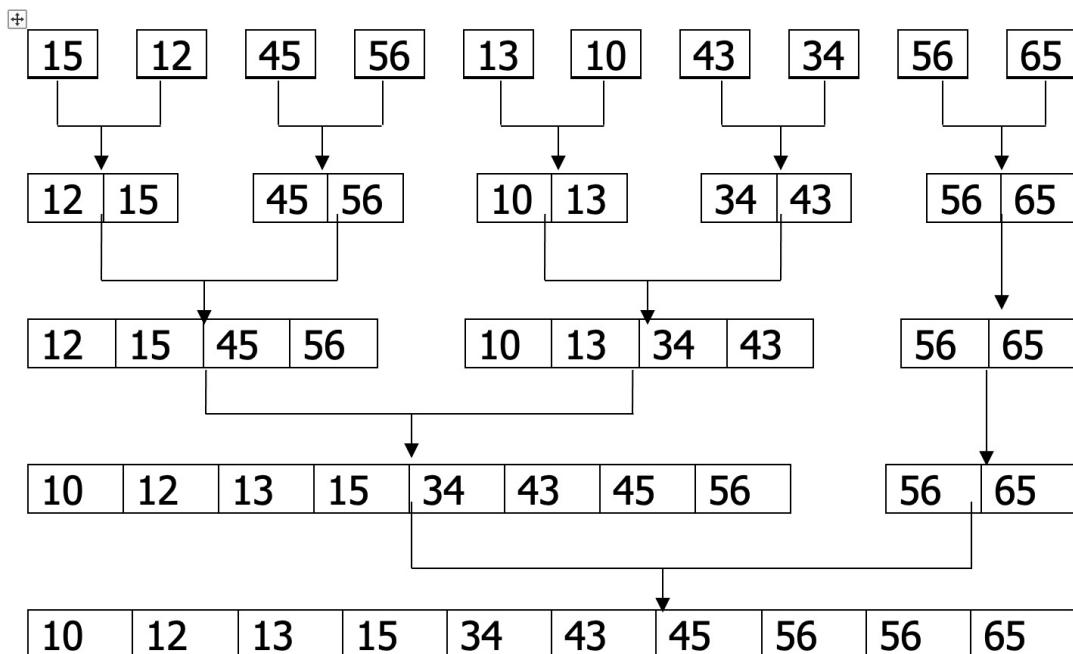
## *Outlines:*

- **SORTING**
  - Merge Sort
  - Counting Sort

## Merge Sort

Dinamakan juga metoda MergeSort Dua Arah (Two-way MergeSort). Metoda ini memanfaatkan keteraturan yang diperoleh dari hasil merging dua buah vektor.

Ilustrasi :



## KONSEP MERGING

Merging : Penggabungan dua kumpulan data yang kedua-duanya dalam keadaan urut menjadi satu kumpulan yang juga dalam keadaan urut.

### Algoritma Merging

- Langkah 0      Baca kedua vektor yang akan dimerge, vektor A dan vektor B, banyaknya elemen pada kedua vektor  $C_a$  dan  $C_b$ . Lakukan proses pengurutan pada kedua vektor tsb.
- Langkah 1      (Menentukan subskrip awalan)  
                  Tentukan  $i=1, j=1$  dan  $C_c=0$
- Langkah 2      (Proses merging berlangsung)  
                  Kerjakan langkah 3 dan 4 sampai  $(i > C_a)$  atau  $(j > C_b)$
- Langkah 3      Tentukan :  $C_c = C_c + 1$
- Langkah 4      test apakah  $A[i] < B[j]$  ?  
                  Jika ya, tentukan :  $C[C_c] = B[j]$  dan  $i=i+1$   
                  Jika tidak, tentukan :  $C[C_c] = B[j]$  dan  $j=j+1$
- Langkah 5      Test apakah  $i > C_a$  ?  
                  Jika ya, berarti semua elemen pada vektor pertama sudah diproses semua, kerjakan langkah 6  
                  Jika tidak, berarti semua elemen pada vektor kedua sudah diproses semua, kerjakan langkah 7
- Langkah 6      Untuk  $k=j$  sampai  $C_b$ , tentukan :  
                   $C_c=C_c+1$  dan  $C[C_c]=B[k]$ , lompat ke langkah 8
- Langkah 7      Untuk  $k=i$  sampai  $C_a$ , tentukan :  
                   $C_c=C_c+1$  dan  $C[C_c]=A[k]$
- Langkah 8      Selesai

## Counting Sort

Counting Sort menganggap bahwa setiap elemen-elemen masukan  $n$  adalah integer pada range 1 sampai dengan  $k$ . Ide dasar dari counting sort adalah menentukan, setiap elemen input  $x$ , jumlah elemen-elemen yang kurang dari  $x$ . Informasi ini dapat digunakan untuk menempatkan setiap elemen  $x$  secara langsung ke posisinya pada array keluaran. Sebagai contoh, jika terdapat 17 elemen yang kurang dari  $x$ , maka  $x$  berada pada posisi ke-18.

### Ilustrasi

Array masukan  $A[1..8]$ , dimana setiap elemen  $A$  adalah positif integer yang tidak lebih besar dari  $k=6$

A

3	6	4	1	3	4	1	4
---	---	---	---	---	---	---	---

C

2	0	2	3	0	1
1	2	3	4	5	6

Array C setelah isinya dijumlahkan secara komulatif (baris 7)

C

2	2	4	7	7	8
1	2	3	4	5	6

Hasil iterasi dari baris 9 – 11

⇒

B							4	
---	--	--	--	--	--	--	---	--

C	2	2	4	6	7	8
1	2	3	4	5	6	

⇒

B		1				4		
---	--	---	--	--	--	---	--	--

C	1	2	4	6	7	8
1	2	3	4	5	6	

⇒

B		1				4	4	
---	--	---	--	--	--	---	---	--

C	1	2	4	5	7	8
1	2	3	4	5	6	

dan seterusnya (ikuti algoritma), sehingga hasil akhir disimpan pada array B sebagai berikut :

B	1	1	3	3	4	4	4	6
---	---	---	---	---	---	---	---	---

Algoritma counting sort

**CountingSort(A,B,k)**

1. for  $i \leftarrow 1$  to  $k$
2. do  $C[i] \leftarrow 0$
3. for  $j \leftarrow 1$  to  $\text{length}[A]$
4. do  $C[A[j]] \leftarrow C[A[j]] + 1$
5.  $C[i]$  now contains the number of elements equal to  $i$
6. for  $i \leftarrow 2$  to  $k$
7. do  $C[i] \leftarrow C[i] + C[i-1]$
8.  $C[i]$  now contains the number of elements less than or equal to  $i$
9. for  $j \leftarrow \text{length}[A]$  downto 1
10. do  $B[C[A[j]]] \leftarrow A[j]$
11.  $C[A[j]] \leftarrow C[A[j]] - 1$

Terima kasih