



Institut Teknologi  
**Telkom**  
Purwokerto

*Teknik Informatika - Fakultas Informatika*

# Pertemuan 1 – Pengantar Algoritma dan Struktur Data

Author: **Condro Kartiko [CKO]**

Co-Author: **Wahyu Andi Saputra [WAA]**

# TUJUAN

Agar dapat melakukan perancangan dan pemilihan struktur data yang tepat implementasi

melakukan analisis secara umum pada algoritma yang dibuat

Meningkatkan Skala sistem yang lebih besar dan lebih efisien dan elegan.

*“Programming to an interface”*

Prinsip-prinsip dasar RPL: abstraksi, modularitas, dst.

# Arti kata (Webster)

- **da•ta (n.pl.)**

1. facts or figures to be processed; evidence, records, statistics, etc. from which conclusions can be inferred; information

- **struc•ture (n.)**

1. manner of building, constructing, or organizing
2. something built or constructed, as a building or dam
3. the arrangement or interrelation of all the parts of a whole; manner of organization or construction [**the structure of the atom, the structure of society**]
4. something composed of interrelated parts forming an organism or an organization

# Arti kata (Webster)

- **al•go•rithm (n.)**

1. *Math.* a) any systematic method of solving a certain kind of problem b) the repetitive calculations used in finding the greatest common divisor of two numbers (called in full Euclidean algorithm)

2. *Comput.* a predetermined set of instructions for solving a specific problem in a limited number of steps

- **Contoh:**

- Problem: mencari sebuah elemen dalam array terurut
- Algoritma: *binary search*

# Struktur Data

- Semua program berurusan dengan data
  - Sistem informasi: informasi, laporan, user, ...
  - Game: posisi & status pemain, musuh, skor, ...
  - Search engine: URL, isi, hyperlink, bobot, ...
- Mengapa data itu disimpan?
  - Supaya bisa diakses/diproses di kemudian waktu
- Mengapa dalam penyimpanan data diperlukan sebuah struktur?
  - Supaya lebih mudah/efisien dalam pengaksesan/pemrosesan data tersebut

# Struktur Data

- Struktur data memudahkan kita untuk
  - *component reuse*.
    - – Sekali kita implementasi, dapat digunakan berkali-kali dalam aplikasi yang berbeda

# Mengapa kuliah ini penting?

- Apakah kuliah Aprosaja tidak cukup?
- Perhatikan program untuk menghitung jumlah kemunculan angka 1 sampai 500 dalam sebuah file:

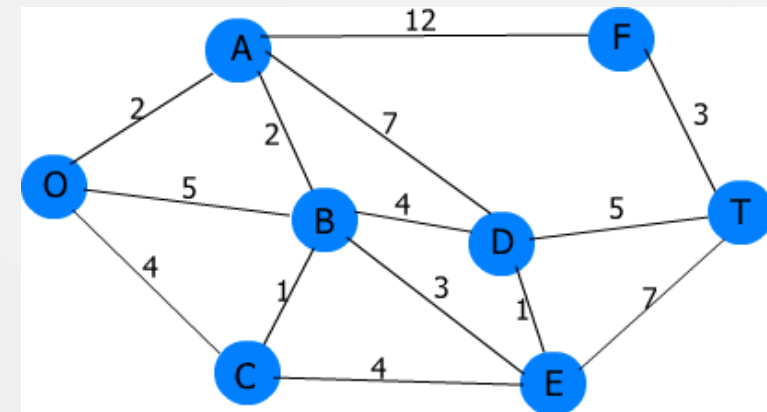
```
if (k == 1) c001++;  
if (k == 2) c002++;  
...  
if (k == 500) c500++;
```

- Program di atas >500 baris.
- Program di atas benar walaupun tidak efisien, sangat besar (500 lines of code), dan sulit dipelihara.
- Solusi sederhana: gunakanlah array integer yang terdiri dari 500 elemen:

```
int c[500];  
c[k]++;
```

# Mengapa kuliah ini penting?

- Moral of the story:
  - Pemilihan **struktur data** maupun **algoritma** yang tepat dapat membuat program lebih: efisien, mudah, elegan
- Contoh Aplikasi:
  - Mencari jarak terpendek antara dua kota
    - menggunakan struktur data Graph





# Mengapa kuliah ini penting?

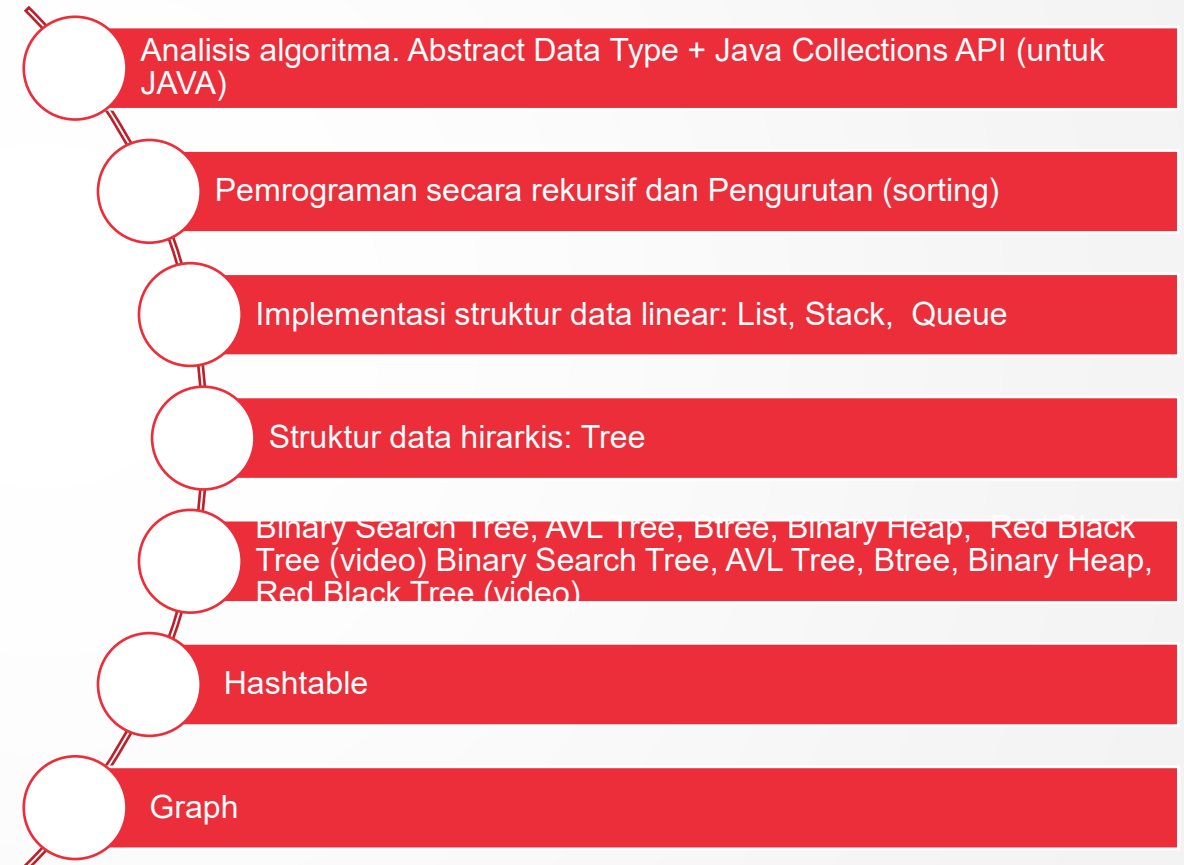
- Contoh Aplikasi:
  - Sistem basis data (Oracle, SQLServer, dll)
    - menggunakan struktur data BTree, Hashtable
  - Menghitung ekspresi:  $(5 + 2) * 7$ 
    - menggunakan struktur data Stack/Tree

# *Mengapa Belajar Implementasi Struktur Data?*

- Mengetahui kelebihan dan kekurangan dari masing- masing struktur data.
- Cara yang terbaik untuk benar-benar dapat memahami masing-masing struktur data adalah membuatnya.
- Menyesuaikan struktur data yang ada untuk problem baru (*augmented data structure*)
- Dalam industri, bahasa yang digunakan tidaklah selalu Java. Mungkin saja di bahasa tersebut tidak terdapat library untuk struktur data.
- Melatih berpikir tentang efisiensi



# *Topik-Topik yang Dibahas*





# SUMMARY

- Struktur data + Algoritma = Program
- Pemilihan struktur data dan algoritma yang tepat dapat membuat program lebih efisien, mudah, dan elegan



Institut Teknologi  
**Telkom**  
Purwokerto

# TERIMA KASIH

Angkat tangan apabila ada pertanyaan