

# Project UAS

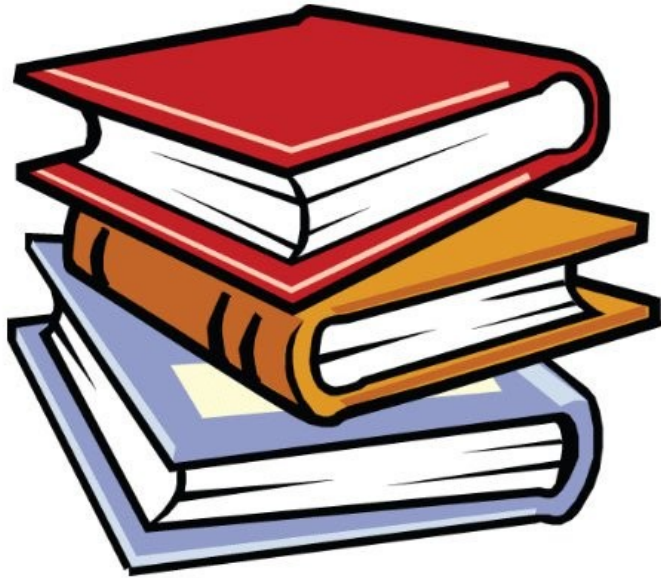
MK IF21172 Struktur data & algoritme  
Kelas C,D,E (dosen IFA)

# Project UAS

- Buatlah kelompok 3 orang, daftar kelompok ini sekaligus submit ide gambaran program ke dalam Assignment di LMS (format docx)
- Laporkan progress pada hari Kamis (21 Juli), Deadline 1 Agustus
- Delivery: Kode program dan demo. Jadwal demo menyusul
- Berikut 2 contoh kasus penerapan struktur data (queue & graph)
- **NB:** contoh kasus ini sekadar gambaran saja, karena idenya sudah dipaparkan maka jangan dijadikan kasus pada tugas Anda (cari ide kasus yang lain)

# Contoh penerapan Queue

- Sistem peminjaman buku di Perpustakaan



Diberikan sistem yg menyimpan data buku sebagai array

Buku meliputi data: id (no. buku), judul buku, tahun terbit, jumlah yg ada di perpustakaan

Contoh command: print katalog

```
=====
no | judul          | thn terbit | jml sedia
=====
01  laskar pelangi  2008       2
02  hary potter    2005       3 * → tanda ada antrian
03  algoritma      2020       2
```

# Sistem perpustakaan

- Command 2 (Pinjam buku)

Input data : no. buku (id), nama mahasiswa, nim

Jika buku tersedia maka print “buku tersedia, silakan hubungi petugas”

Jika tidak tersedia, print “buku sedang dipinjam” & tawarkan kepada user: “apakah mau mengantri ?” jika ya maka tambahkan mahasiswa tsb ke dalam antrian.

Jika tidak mau antri, kembali ke menu awal

# Sistem perpus

- Command 3 (pengembalian buku)

Input: no/id buku

Akan melakukan Dequeue dari antrian (jika lebih dari satu peminjam)

Jika tidak, cukup print nama mhs dan NIM-nya yg terakhir meminjam

# Sistem perpustakaan

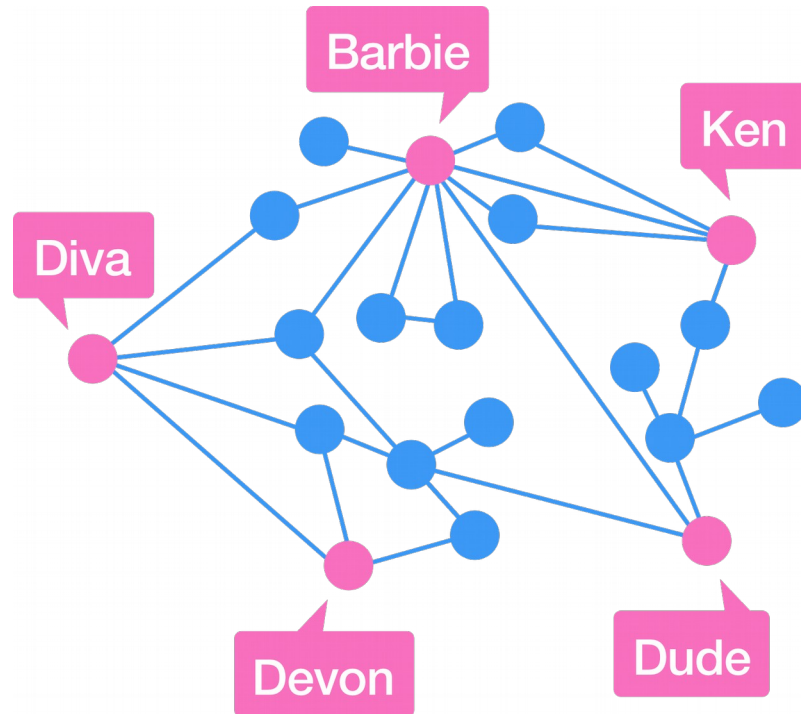
- Command 4 (lihat antrian )

input: no/id buku

akan mencetak isi antrian dari depan ke belakang, yaitu data mahasiswa yang sedang atau akan meminjam buku tsb.

# Contoh penerapan Graph

- Simulasi jaringan sosial



Tiap simpul pada graph merepresentasikan Orang, sedangkan sisi menggambarkan Hubungan pertemanan

# Simulasi jaringan sosial

- Simpul (node) sebagai orang

data: nama orang, umur, jenis kelamin (L/P)

## Input Data Orang

Kondisi awal/ pre kondisi : Graph kosong, atau graph sudah terisi

Input data orang (nama, usia, gender L/P). ini akan membuat simpul (node) lalu ditambahkan ke dalam graph tsb



# Simulasi (2)

- Pembentukan graph secara dinamis

Graph dibentuk secara dinamis dengan menambah simpul (node) saat runtime program, bukan statis Pre-kompilasi ( statik saat koding )

Yaitu dengan menambah simpul (data orang)

# Simulasi (3)

- Bagaimana graph dibentuk secara acak?

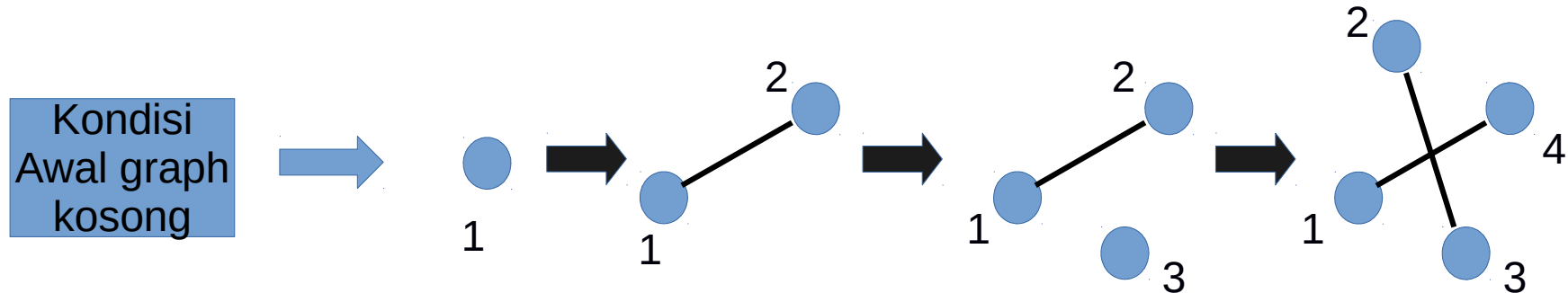
Tiap kali ditambahkan satu simpul baru (data orang),  
Pilih secara acak 2 simpul data yang sudah ada pada  
graph

Jika kondisi awal: graph kosong lalu ditambah 1 simpul,  
berarti langkah ini bisa dilewati karena baru ada 1 simpul  
dalam graph

# Simulasi (4)

- Gunakan fungsi acak (random)

Fungsi acak menghasilkan nilai desimal  $0 < s.d \leq 1$  (jika nilai  $\geq 0.5$  maka sambungkan 2 simpul tsb. Jika tidak, maka biarkan kondisi apa adanya). Contoh:



# Simulasi (5)

- Command

program membuat saran pertemanan, yaitu mencari temannya teman (teman dari teman)

- Input : id (nomor simpul dlm graph) / id orang

output: mencetak nama dari temannya teman, jika mungkin cetak sebagai list dengan nomor

# Contoh: output temannya teman

- Saran teman untuk node id 1 (Eko):
  1. Indra
  2. Wina
  3. Andini
  4. Toni

# Simulasi (6)

- Command :

print seluruh orang dalam graph tersebut terurut dari orang yang memiliki teman paling banyak dulu kemudian menurun hingga yang temannya paling sedikit.

Contoh output:

```
=====
nama  jml teman
=====
Andi   3
Budi   2
Nia    2
Ani    1
```