Teknik Informatika Kampus 3 Nganjuk
Jurusan Teknologi Informasi

Politeknik Negeri Jember

Multi Linked List

Struktur Data

Ulfa Emi Rahmawati, S.Kom., M.Kom.

Definisi

Multi linked list

Multi linked list

- Modifikasi single atau double linked list sehingga mereka dapat tampil lebih baik dengan case yang ada
- Bertujuan untuk menggambarkan hubungan antar data
  - Relasi 1 to N
  - Relasi N to M
  - Tree dan Graph
- Dalam bentuk :
  - List di dalam list
  - Koneksi antara 2 atau lebih list

Contoh

Multi linked list

Contoh

Student – Class

- Misalkan kita ingin menyimpan data siswa dan relasinya dengan default class
- Relasi?
  - Relasi 1 to N

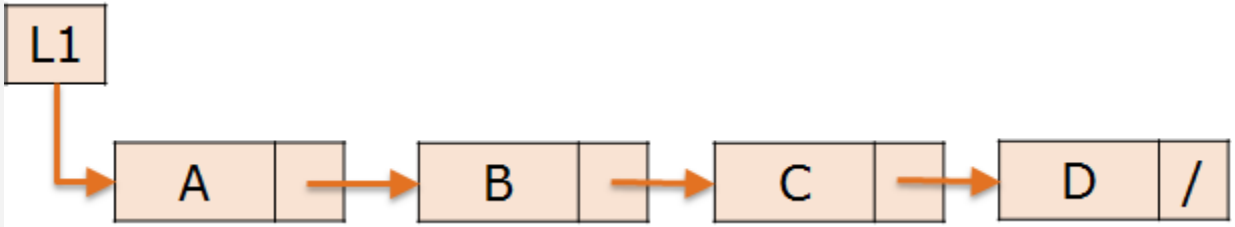
Contoh (Student - Class)

```
Type infotype_student <
        id          : string
        name        : string
>

Type adr_student : pointer to elm_student

Type elm_student <
        info        : infotype_student
        next        : adr_student
>

Type list_student <
        first : adr_student
>
```

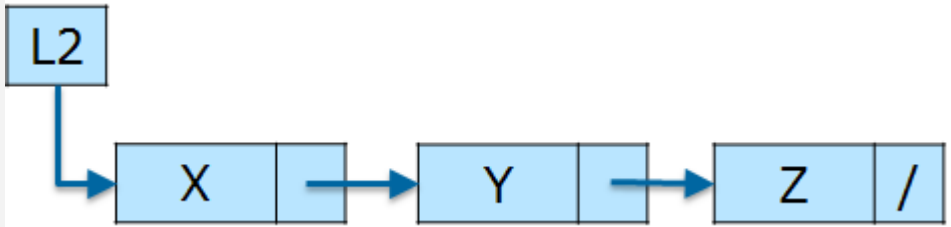
```
Type infotype_class <
      class_name    : string
      supervisor    : string
>

Type adr_class : pointer to elm_class

Type elm_class <
      info  : infotype_class
      next : adr_class
>

Type list_class <
      first : adr_class
>
```

L1 : list_student
L2 : list_class

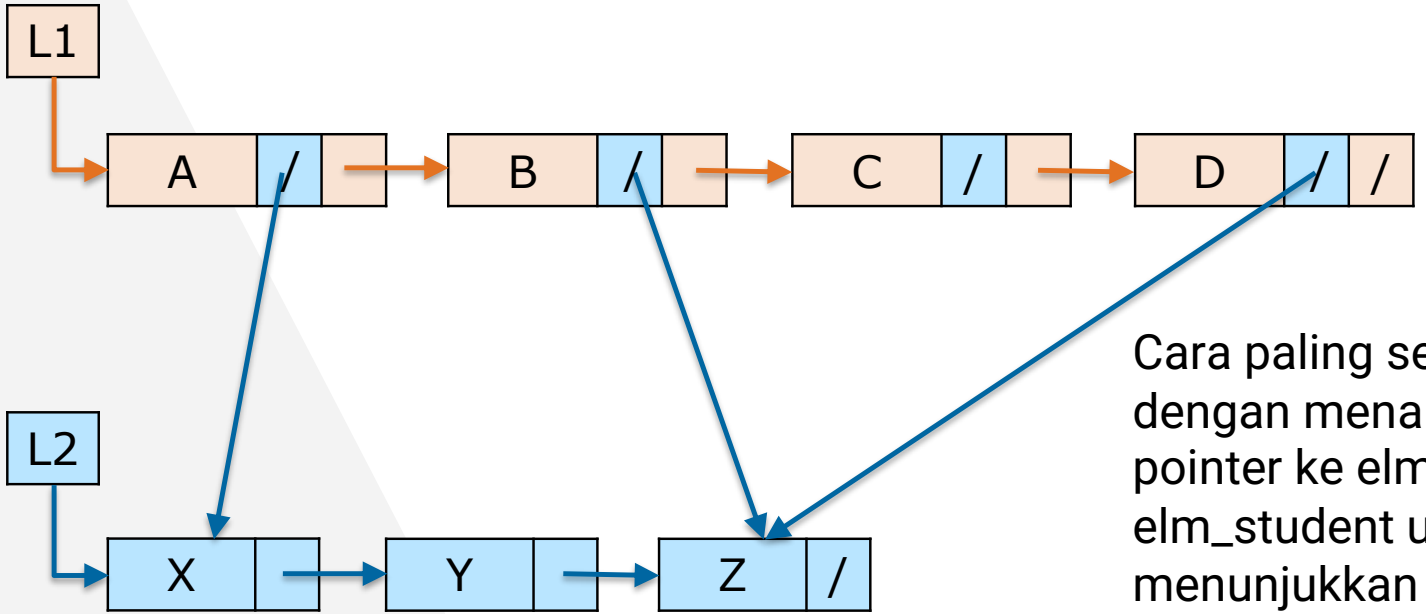
Contoh (Student – Class)



Sekarang, bagaimana cara menggambar hubungan antar data?



Contoh (Student - Class)



Cara paling sederhana adalah dengan menambahkan pointer ke elm_class di elm_student untuk menunjukkan class mana yang didaftarkan siswa

Contoh (Student - Class)

```
Type infotype_student <
        id          : string
        name        : string
>

Type adr_student : pointer to
elm_student

Type elm_student <
        info        : infotype_student
        next        : adr_student
        nextClass : adr_class
>
```

```
Type infotype_class <
    class_code    : string
    class_name    : string
    credit        : integer
>

Type adr_class : pointer to elm_class

Type elm_class <
    info        : infotype_class
    next        : adr_class
>
```

Contoh

Student – Course

- Misalkan kita ingin menyimpan data siswa dan hubungannya dengan course yang diambil
- Relasi?
  - Relasi N to M

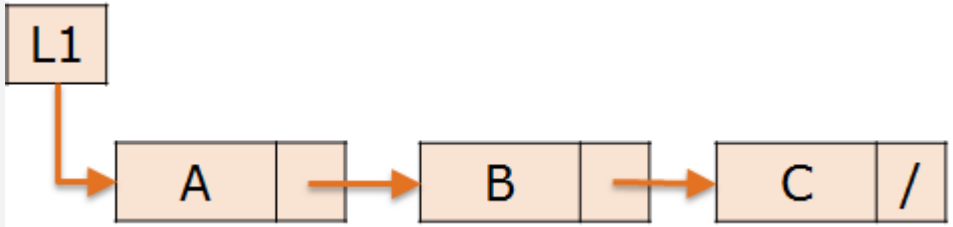
Contoh (Student - Course)

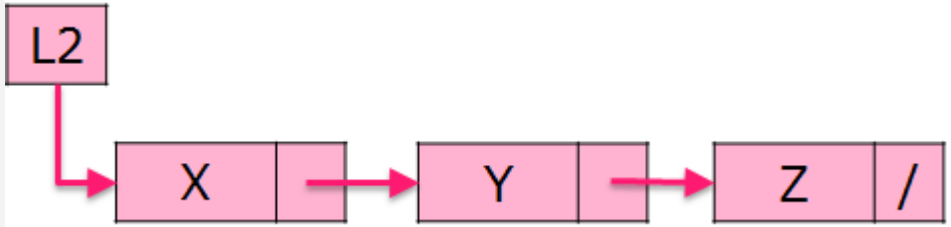
```
Type infotype_student <
        id          : string
        name        : string
>
Type adr_student : pointer to
elm_student
Type elm_student <
        info        : infotype_student
        next        : adr_student
>
Type list_student <
        first       : adr_student
>
```

```
Type infotype_course <
        course_id       : string
        course_name     : string
        credit          : integer
>
Type adr_course : pointer to
elm_course
Type elm_course <
        info   : infotype_course
        next : adr_course
>
Type list_course <
        first : adr_course
>
```

L1 : list_student
L2 : list_course

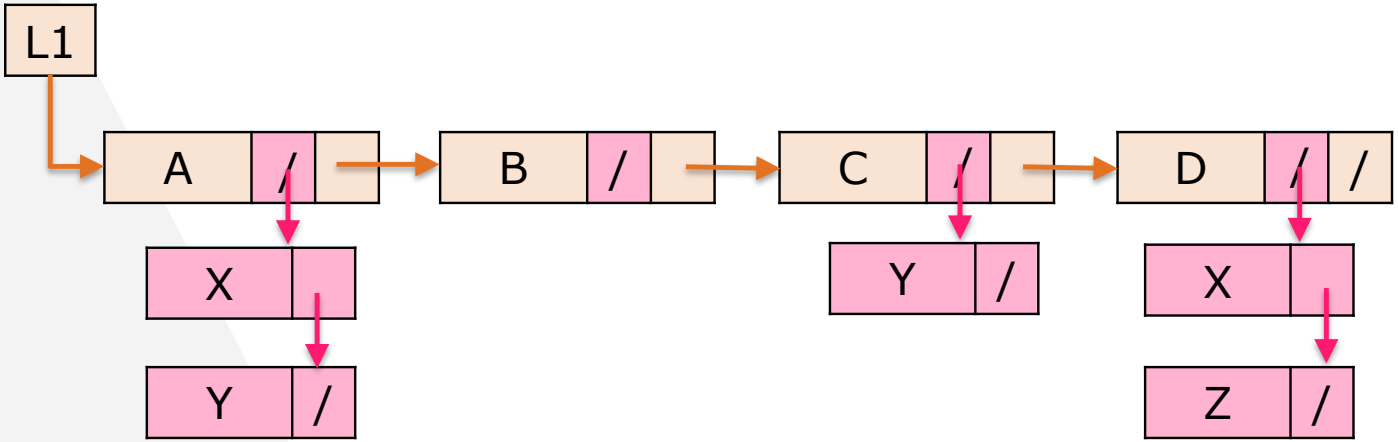
Contoh (Student – Course)

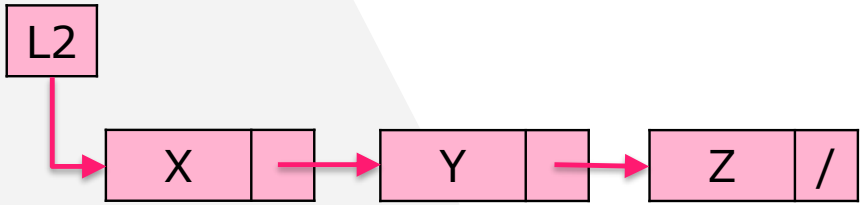




Sekarang, bagaimana cara menggambar hubungan antar data?

Contoh (Student – Course)





Untuk kasus ini, masukkan daftar course di dalam setiap elm_student untuk menunjukkan setiap course yang diambil oleh student

Contoh (Student - Course)

```
Type infotype_student <
        id          : string
        name        : string
>

Type adr_student : pointer to
elm_student

Type elm_student <
        info        : infotype_student
        next        : adr_student
        course      : list_course
>
```

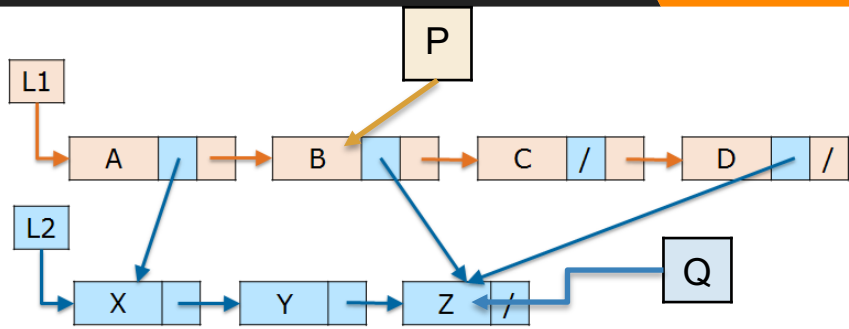
```
Type infotype_course <
    course_id       : string
    course_name     : string
    credit          : integer
>

Type adr_course : pointer to
elm_course

Type elm_course <
    info        : infotype_course
    next        : adr_course
>
```

Exercise

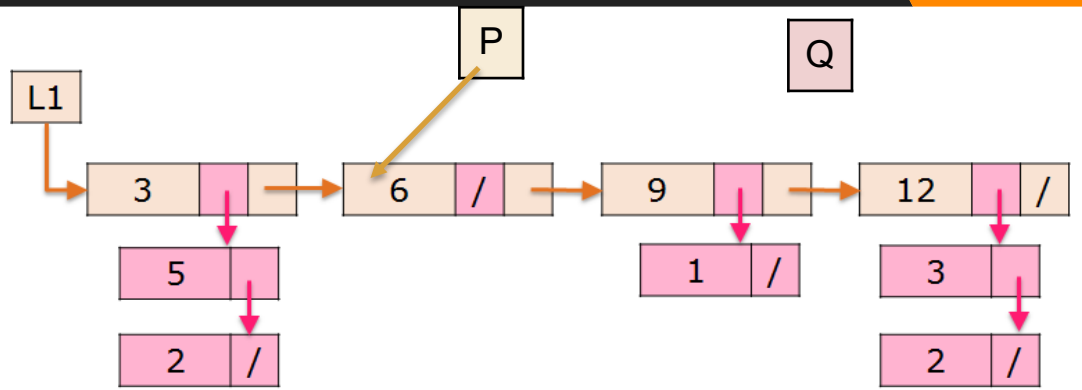
Latihan bersama

Exercise

No	Operation	Answer
1	Info(P)	
3	Info(next(child(first(L1))))	
4	Info(next(P))	
5	Info(first(L1)) + info(child(P))	
6	P ← next(P) Child(P) ← next(first(L2)) Info(child(P))	



Exercise



No	Operation	Answer
1	Info(next(child(first(L1))))	
2	Q ← next(child(next(P)))) P ← next(first(L1)) Info(P) - info(Q)	

THANKS!

Any questions?