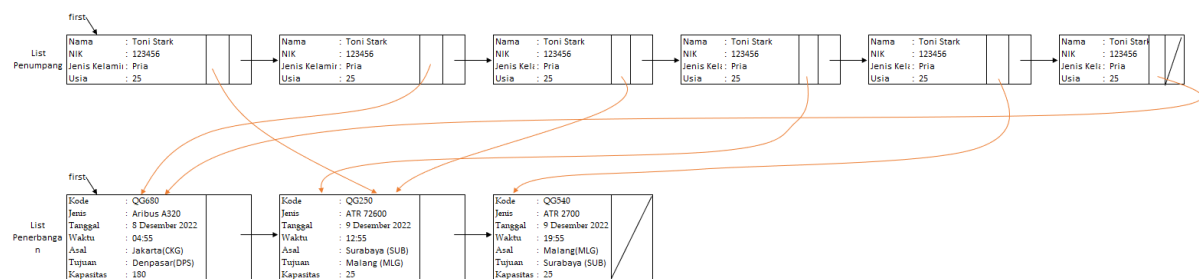


Jurnal Modul 10

STRUKTUR DATA – Ganjil 2024/2025

"Multi Linked List"

Diketahui sebuah multi linked list hubungan 1 ke N yang menyimpan data penumpang pesawat beserta jadwal penerbangan. Penumpang disimpan dalam list penumpang, dan jadwal penerbangan disimpan dalam list jadwal. Setiap penumpang akan memiliki 1 jadwal penerbangan. Ilustrasi sebagai berikut



1. Buatlah Code::Block Console C++ Project baru.
2. Bukalah file header dengan nama file: "flight.h"
3. Bukalah file implementasi (*.cpp), nama file: "flight.cpp"
4. Di dalam file flight.h, sudah terdapat ADT untuk menyimpan data diatas, dengan bentuk:

```
Type adr_jadwalP : pointer to elemenJadwal
Type adr_penumpangP : pointer to elemenPenumpang
Type infotypePenumpang : penumpang
Type infotypeJadwal : jadwalPenerbangan

type penumpang <
    nama, nik, jk: string
    usia: int
>

Type jadwalPenerbangan <
    kode, jenis, tanggal, waktu, asal, tujuan: string
    kapasitas: int
>

Type elemenJadwal <
    info : infotypeJadwal
    next : adr_jadwalP
>
```

```
Type elemenPenumpang <
  info : infotypePenumpang
  jadwalP : adr_jadwalP
  next: adr_penumpangP
>

Type listPenumpang <
  first: adr_penumpangP
>

Type listJadwal <
  first : adr_jadwalP
>
```

5. Di dalam flight.h, terdapat beberapa fungsi procedure yang sudah Anda kerjakan saat TP (no 1-6) dan **6 tambahan fungsi procedure yang Anda kerjakan saat jurnal (no 7-13).**

1. **Procedure** createListJadwal(**input/output** LJ : listJadwal)
2. **Function** createElemenJadwal(x: infotypeJadwal) → adr_jadwalP
3. **Procedure** insertLastJ (**input/output** LJ : listJadwal, **input** p: adr_jadwalP)
4. **Procedure** showJadwal(**input** LJ : listJadwal)
5. **Procedure** deleteFirstJ(**input/output** LJ : listJadwal, **output** p: adr_jadwalP)
6. **Function** SearchJ (LJ : listJadwal, dari : string, ke :string, tanggal : string) → adr_jadwalP

7. **Procedure** createListPenumpang(**input/output** LP : listPenumpang)
8. **Function** createElemenPenumpang(x: infotypePenumpang)→ adr_penumpangP
9. **Procedure** insertLastP(**input/output** LP : listPenumpang, **input** p: adr_penumpangP)
10. **Procedure** booking(**input/output** LP : listPenumpang, **input** x : infotypePenumpang, LJ : listJadwal, dari : string, ke : string, tanggal : string)
11. **Procedure** showPenumpang(**input** LP : listPenumpang)

- ```

12. Procedure reschedule(input/output LP : listPenumpang,
 input LJ : listJadwal, NIK : string, input newDari :
 string, newKe : string, newTanggal : string)
13. Procedure showAllListPenumpang(LJ : listJadwal, LP :
 listPenumpang, kode: string)

```

6. Beralih ke dalam file implementasi flight.cpp. Dari 11 Fungsi procedure, fungsi procedure 1-7 telah Anda kerjakan di TP, tugas Anda melengkapi 6 fungsi procedure nomor 7-13 dengan detail spesifikasi sbb :

```

// Function dan procedure nomor 1-6 sudah ada
// implementasinya pada file flight.cpp

1. Procedure createListJadwal(input/output LJ : listJadwal)
{
 FS: mengembalikan sebuah list kosong
}

2. Function createElemenJadwal(x: infotypeJadwal) →
 adr_jadwalP
{
 IS: Tersedia data jadwal penerbangan
 FS: Mengembalikan alamat elemen list hasil alokasi,
 info berisi data dari masukan
}

3. Procedure insertLastJ (input/output LJ : listJadwal,
 input p: adr_jadwalP)
{
 IS: Terdefinisi sebuah list L (mungkin kosong) dan
 sebuah pointer p yang berisi alamat elemen baru
 FS: Elemen baru yang ditambahkan sebagai elemen
 terakhir dari list L
}

4. Procedure showJadwal(input LJ : listJadwal)
{
 IS: terdefinisi sebuah list L (mungkin kosong)
 FS: menampilkan isi dari list L secara horizontal
}

```

```

 }

5. Procedure deleteFirstJ(input/output LJ : listJadwal,
output p: adr_jadwalP)

{
 IS: Terdefinisi sebuah list L (mungkin kosong)
 FS: Pointer P berisi alamat elemen terakhir dari list
 L, dan elemen tersebut dihapus dari list L
}

6. Function SearchJ (LJ : listJadwal, dari : string, ke
:string, tanggal : string) → adr_jadwalP

{
 IS: Terdefinisi sebuah list L (mungkin kosong)
 FS: mengembalikan alamat dari elemen list L yang
 memiliki info asal=dari, tujuan=ke dan tanggal
 keberangkatan=tanggal, atau NIL apabila tidak
 ditemukan
}

// Fungsi procedure yang Anda kerjakan saat jurnal

7. Procedure createListPenumpang(input/output LP :
listPenumpang)

{
 FS: mengembalikan sebuah list kosong
}

8. Function createElemenPenumpang(x: infotypePenumpang) →
adr_penumpangP
{
 Function akan membuat element yang berisi data
 penumpang dan mengembalikan alamat dari element
 penumpang tersebut
}

9. Procedure insertLastP(input/output LP : listPenumpang,
input p: adr_penumpangP)
{
 I.S. Terdefinisi list Penumpang yang mungkin kosong
 dan sebuah element baru yang disimpan oleh pointer
 p

```

```
 F.S. Element baru menjadi element terakhir pada list
}

10.Procedure booking(input/output LP : listPenumpang, input
x : infotypePenumpang, LJ : listJadwal, dari : string,
ke : string, tanggal : string)
{
 I.S. Terdefinisi sebuah list penumpang, data
 penumpang, dan data jadwal penerbangannya berupa
 asal penerbangan (dari), tujuan penerbangan(ke),
 dan tanggal penerbangan(tanggal). Asumsi hanya ada
 1 jadwal penerbangan dari A ke B pada tanggal X
 pada list Jadwal, namun bisa saja tidak ditemukan.
 List Penumpang bisa kosong.
 F.S. Elemen penumpang baru masuk menjadi elemen akhir
 di list penumpang, dan jadwal nya terkoneksi ke
 element pada list Jadwal
}

11.Procedure showPenumpang(input LP : listPenumpang)
{
 I.S. terdefinisi List Penumpang yang mungkin kosong
 F.S. Jika list tidak kosong maka tampilkan data Nama,
 NIK, jenis Kelamin, usia, dan Kode penerbangan
}

12.Procedure reschedule(input/output LP : listPenumpang,
input LJ : listJadwal, NIK : string, input newDari :
string, newKe : string, newTanggal : string)
{
 I.S. Terdefinisi NIK penumpang yang akan reschedule
 tanggal penerbangannya. Data tanggal baru
 penerbangannya(newTanggal). Asal dan tujuan
 penerbangan masih sama.
 F.S. Jika tanggal yang baru tersedia, maka data
 penerbangan penumpang diubah, sebaliknya maka
 customer tidak jadi direschedule.
}

13.Procedure showAllListPenumpang(LJ : listJadwal, LP :
listPenumpang, kode: string)
```

```
{
 I.S. Terdefinisi list jadwal, list penumpang, dan kode
 penerbangan yang ingin ditampilkan list
 penumpangnya. kode penerbangan bisa jadi tidak
 ditemukan.
 F.S. Jika kode penerbangan ditemukan dan ada
 penumpangnya, maka tampilkan total jumlah
 penumpangnya beserta data Nama dan NIK setiap
 penumpang penerbangan tsb ke layar.
}
```

7. Kemudian pada main.cpp isilah dengan sintak c++ sesuai tahapan sbb :

1. Buatlah 1 buah list Jadwal  
.....
2. Dengan menggunakan perulangan, Buatlah 4 element berisi jadwal penerbangan. Data jadwal penerbangan sesuka Anda, **namun pastikan kode jadwal unik dan ada minimal 2 element yang berisi jadwal penerbangan pada asal dan tujuan yang sama namun berbeda tanggal**  
.....
3. Tampilkan 4 jadwal yang sudah Anda insert ke list jadwal  
.....
4. Buatlah 1 buah list penumpang  
.....
5. Dengan menggunakan perulangan, panggil procedure insertLastP dan masukkan data 6 penumpang. Aturan nama pada 6 data sbb adalah:  
(1) Nama Anda,  
(2) Nama Panggilan Dosen Wali Anda,  
(3) Nama/kode asprak Anda hari ini,  
(4) Nama panggilan teman Anda yang duduk di sebelah kiri/kanan Anda,  
(5) Nama penyanyi yang Anda sukai,  
(6) Salah satu nama dosen MK Anda semester ini.  
Data detail penumpang selain nama bebas, namun pastikan **saat mengisi NIK datanya UNIK. Data jadwal per penumpang silahkan Anda tentukan sesuai data pada list jadwal yang Anda buat sebelumnya**  
.....
6. Tampilkan 6 data penumpang yang telah Anda buat dengan memanggil procedure show Penumpang  
.....
7. Tampilkan data jadwal dan list penumpangnya dengan memanggil procedure showAllListPenumpang  
.....
8. Pindahkan 1 penumpang ke tanggal penerbangan lain (asal dan tujuan sama)  
.....
9. Tampilkan data jadwal dan list penumpangnya dengan memanggil procedure showAllListPenumpang. Cek apakah data yang Anda reschedule sudah berubah

.....

Perlu diperhatikan:

Pada file implementasi flight.cpp dan main.cpp jangan lupa sertakan **#include "flight.h"**, pada baris pertama kode.

**Contoh** masukan dan keluaran (teks bergaris bawah adalah masukan):

```
Input Penerbangan
Data ke-1
Kode: QG680
Jenis: Aribus-A320
Tanggal: 8-Desember-2022
Waktu: 04:55
Asal: Jakarta (CKG)
Tujuan: Denpasar (DPS)
Kapasitas: 180

Data ke-2
Kode: QG250
Jenis: ATR-72600
Tanggal: 9-Desember-2022
Waktu: 12:55
Asal: Surabaya (SUB)
Tujuan: Malang (MLG)
Kapasitas: 25

Data ke-3
Kode: QG540
Jenis: ATR-2700
Tanggal: 9-Desember-2022
Waktu: 19:55
Asal: Malang (MLG)
Tujuan: Surabaya (SUB)
Kapasitas: 25

Data ke-4
Kode: QG790
Jenis: GTR-370
Tanggal: 10-Desember-2022
Waktu: 13:30
Asal: Jakarta (CKG)
Tujuan: Denpasar (DPS)
```



Kapasitas: 45

=====JADWAL=====

Kode: QG680

Jenis: Aribus-A320

Tanggal: 8-Desember-2022

Waktu: 04:55

Asal: Jakarta (CKG)

Tujuan: Denpasar (DPS)

Kapasitas: 180

Kode: QG250

Jenis: ATR-72600

Tanggal: 9-Desember-2022

Waktu: 12:55

Asal: Surabaya (SUB)

Tujuan: Malang (MLG)

Kapasitas: 25

Kode: QG540

Jenis: ATR-2700

Tanggal: 9-Desember-2022

Waktu: 19:55

Asal: Malang (MLG)

Tujuan: Surabaya (SUB)

Kapasitas: 25

Kode: QG790

Jenis: GTR-370

Tanggal: 10-Desember-2022

Waktu: 13:30

Asal: Jakarta (CKG)

Tujuan: Denpasar (DPS)

Kapasitas: 45

=====

Input Jadwal

Data ke-1

Nama: nama-Anda

NIK: 12345890

Jenis Kelamin: Pria

Usia: 20

Asal Penerbangan: Jakarta (CKG)

Tujuan Penerbangan: Denpasar (DPS)

Tanggal Penerbangan: 8-Desember-2022

Data ke-2

Nama: Nama-Panggilan-Dosen-Wali-Anda

NIK: 6475829107

Jenis Kelamin: Pria

Usia: 20

Asal Penerbangan: Surabaya (SUB)

Tujuan Penerbangan: Malang (MLG)

Tanggal Penerbangan: 9-Desember-2022

Data ke-3

Nama: Nama/kode-asprak-Anda-hari-ini

NIK: 5638291605

Jenis Kelamin: Pria

Usia: 20

Asal Penerbangan: Jakarta (CKG)

Tujuan Penerbangan: Denpasar (DPS)

Tanggal Penerbangan: 10-Desember-2022

Data ke-4

Nama: Nama-panggilan-teman-Anda-yang-duduk-di-sebelah-kiri/kanan-Anda

NIK: 547593420

Jenis Kelamin: Pria

Usia: 20

Asal Penerbangan: Jakarta (CKG)

Tujuan Penerbangan: Denpasar (DPS)

Tanggal Penerbangan: 8-Desember-2022

Data ke-5

Nama: Nama-penyanyi-yang-Anda-sukai

NIK: 6475870367

Jenis Kelamin: Pria

Usia: 20

Asal Penerbangan: Malang (MLG)

Tujuan Penerbangan: Surabaya (SUB)

Tanggal Penerbangan: 9-Desember-2022

Data ke-6

Nama: Salah-satu-nama-dosen-MK-Anda-semester-ini

NIK: 3658693067

Jenis Kelamin: Pria

Usia: 20

Asal Penerbangan: Jakarta (CKG)

Tujuan Penerbangan: Denpasar (DPS)

Tanggal Penerbangan: 10-Desember-2022

=====PENUMPANG=====

Nama: nama-Anda

NIK: 12345890

Jenis Kelamin: Pria

Usia: 20

Kode: QG680

Nama: Nama-Panggilan-Dosen-Wali-Anda

NIK: 6475829107

Jenis Kelamin: Pria

Usia: 20

Kode: QG250

Nama: Nama/kode-asprak-Anda-hari-ini

NIK: 5638291605

Jenis Kelamin: Pria

Usia: 20

Kode: QG790

Nama: Nama-panggilan-teman-Anda-yang-duduk-di-sebelah-kiri/kanan-Anda

NIK: 547593420

Jenis Kelamin: Pria

Usia: 20

Kode: QG680

Nama: Nama-penyanyi-yang-Anda-sukai

NIK: 6475870367

Jenis Kelamin: Pria

Usia: 20

Kode: QG540

Nama: Salah-satu-nama-dosen-MK-Anda-semester-ini

NIK: 3658693067

Jenis Kelamin: Pria

Usia: 20

Kode: QG790

=====

Tampilkan Penumpang Berdasarkan Kode Penerbangan

Kode: QG790

=====PENUMPANG=====

```
Nama: Nama/kode-asprak-Anda-hari-ini
NIK: 5638291605

Nama: Salah-satu-nama-dosen-MK-Anda-semester-ini
NIK: 3658693067

Total Penumpang : 2
=====

Reschedule
NIK: 12345890
Asal: Jakarta (CKG)
Tujuan: Denpasar (DPS)
Tanggal: 10-Desember-2022

Tampilkan Penumpang Berdasarkan Kode Penerbangan
Kode: QG790

=====PENUMPANG=====
Nama: nama-Anda
NIK: 12345890

Nama: Nama/kode-asprak-Anda-hari-ini
NIK: 5638291605

Nama: Salah-satu-nama-dosen-MK-Anda-semester-ini
NIK: 3658693067

Total Penumpang : 3
=====
```

**Jika ada kesulitan jangan sungkan untuk bertanya kepada asisten praktikum**