

Tugas Pendahuluan Modul 4
STRUKTUR DATA - Genap 2024/2025
"Single Linked List "

A. Ketentuan Tugas Pendahuluan

1. Tugas Pendahuluan dikerjakan secara **Individu**.
2. TP ini bersifat **WAJIB**, tidak mengerjakan = **PENGURANGAN POIN JURNAL / TES ASESMEN**.
3. Hanya **MENGUMPULKAN** tetapi **TIDAK MENGERJAKAN** = **PENGURANGAN POIN JURNAL / TES ASESMEN**.
4. Deadline pengumpulan TP Modul 4 adalah Senin, 9 Oktober 2023 pukul 06.00 WIB.
5. **TIDAK ADA TOLERANSI KETERLAMBATAN, TERLAMBAT ATAU TIDAK MENGUMPULKAN TP MAKA DIANGGAP TIDAK MENGERJAKAN**.
6. **DILARANG PLAGIAT (PLAGIAT = E)**.
7. Kerjakan TP dengan jelas agar dapat dimengerti.
8. File diupload di LMS menggunakan format **PDF** dengan ketentuan: **TP_MOD_[XX]_NIM_NAMA.pdf**
9. **SOAL TEORI WAJIB DIKERJAKAN TULIS TANGAN, TIDAK BOLEH DIKETIK!**

CP (WA):

- Raihan (089638482851)
- Kayyisa (085105303555)
- Abiya (082127180662)
- Rio (081210978384)

SELAMAT MENGERJAKAN^^

B. Soal Praktek

Ikuti langkah-langkah berikut untuk mengerjakan TP modul 4 Materi Single Linked List :

1. Membuat deklarasi tipe List

Buat file list.h dan ketik sintak sesuai gambar berikut.

```
#include <iostream>
#define first(L) L.first
#define next(P) P->next
#define info(P) P->info
using namespace std;
typedef int infotype;
typedef struct elmlist *address;

struct elmlist {
    infotype info;
    address next;
};
struct List{
    address first;
};
```

Buat file list.cpp dan ketik sintak berikut

```
#include <iostream>
#include "list.h"
using namespace std;
```

2. Membuat list kosong, yaitu procedure createList.

Tambahkan pada list.h primitif dari procedure createList

```
void createList(List &L);
```

Tambahkan pada list.cpp implementasi dari procedure createList, sintak c++ sebagai berikut:

```
void createList(List &L){
    /** this procedure will initialize
        the list L*/
    first(L) = NULL;
}
```

3. Setelah list sudah ada, selanjutnya buatlah elemen dengan menggunakan fungsi allocate.

Tambahkan pada list.h primitif dari fungsi allocate

```
address allocate(infotype x);
```

Tambahkan pada list.cpp implementasi dari fungsi allocate, sintak c++ sebagai berikut:

```
address allocate(infotype x){  
    address p = new elmList;  
    info(p) = x;  
    next(p) = NULL;  
    return p;  
}
```

4. Setelah List dan elemen sudah ada, maka selanjutnya elemen tersebut harus diinsert ke List agar bisa menjadi elemen list. Proses insert dapat menggunakan procedure Insert First, procedure Insert Last, atau procedure insert After. Pada Tugas Pendahuluan kali ini, akan dicontohkan menggunakan insert first.

Tambahkan pada list.h primitif procedure insertFirst

```
void insertFirst(List &L, address P);
```

Tambahkan pada list.cpp implementasi dari procedure insertFirst sesuai sintak berikut :

```
void insertFirst(List &L, address P){  
    /** TODO: Insert the new element pointed  
    by P to the first of list L*/  
    // YOUR CODES HERE  
    //-----  
    next(P) = first(L);  
    first(L) = P;  
    //-----  
}
```

5. Setelah proses insert elemen, maka agar bisa mengetahui apakah elemen berhasil diinsertkan, maka kita perlu menampilkan isi list.

Tambahkan pada list.h primitif procedure printInfo

```
void printInfo (List L);
```

Tambahkan pada list.cpp implementasi dari proc printInfo, sintak C++ sebagai berikut :

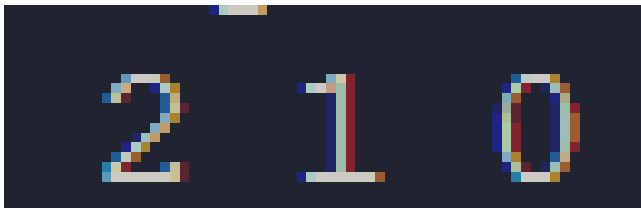
```
void printInfo(List L){  
    /** this procedure will output  
    the info of each element  
    in list L*/  
  
    address p = first(L);  
    while (p != NULL){  
        cout << info(p) << ", ";  
        p = next(p);  
    }  
    cout << endl;  
}
```

6. Sekarang, setelah ADT List sudah terisi dengan beberapa fungsi Procedur di atas, maka mari buat sebuah List berisi 3 elemen yang berisi 3 digit nim terakhir Anda di main.cpp
Adapun gambaran isi dari main.cpp nya adalah sbb :

```
1  #include <iostream>
2  #include "list.h"
3  using namespace std;
4
5
6  int main()
7  {
8      // 1. Panggilah create list
9
10     // 2. Buat sintak menanyakan angka pertama yang ingin diinputkan user ke list
11
12     // 3. Panggil fungsi allocate agar data tersebut diisikan elemen
13
14     // 4. Panggil procedure insert first yang telah dibuat
15
16     // 5. Panggil procedure show info untuk mengecek apakah anak tersebut berhasil unrolli elemen di List.
17
18     // 6. buat kembali sintak no 2 s/d no 5 untuk data angka kedua dari user
19
20     // 7. buat kembali sintak no 2 s/d no 5 untuk data angka ketiga dari user
21
22
23     return 0;
24 }
25
26
```

Tugas rekan-rekan adalah mengisi main.cpp di atas dengan sintak C++ sesuai dengan petunjuk. Buka kembali modul sebelumnya ya ^_^. Setelah selesai, compile dan run. Lalu **screen capture hasilnya**.

Jika dilayar muncul 3 digit nim Anda yang muncul secara terurut TERBALIK dari urutan input.



7. SESI HAVE FUN. Rekan-rekan dapat mencoba hal di bawah ini agar memudahkan saat praktikum:
- Tambahkan procedure insertLast, insertAfter, deleteLast, deleteAfter pada list.h dan list.cpp
 - Tambahkan Function searchInfo pada list.h dan list.cpp
 - Ubah main.cpp agar proses insert N data tidak satu persatu, tapi sesuai dengan jumlah digit NIM yaitu 10 data (clue : gunakan looping). Dan NIM yang diinput, saat di show tidak boleh terurut terbalik (clue : gunakan insert Last) Tampilan (underscore adalah inputan user):

Masukkan NIM perdigit

Digit 1 : 1
Digit 2 : 1
Digit 3 : 3
Digit 4 : 1
Digit 5 : 9
Digit 6 : 6
Digit 7 : 4
Digit 8 : 7
Digit 9 : 4
Digit 10 : 2

Isi list : 1131964742

```
PS D:\04_Single_Linked_List_Bagian_1\04_Single_Linked_List_Bagian_1\TP> ./program
Masukkan digit NIM ke-1: 2
Masukkan digit NIM ke-2: 3
Masukkan digit NIM ke-3: 1
Masukkan digit NIM ke-4: 1
Masukkan digit NIM ke-5: 1
Masukkan digit NIM ke-6: 0
Masukkan digit NIM ke-7: 4
Masukkan digit NIM ke-8: 0
Masukkan digit NIM ke-9: 2
Masukkan digit NIM ke-10: 1
Isi List: 2 3 1 1 1 0 4 0 2 1
PS D:\04_Single_Linked_List_Bagian_1\04_Single_Linked_List_Bagian_1\TP> █
```

SELAMAT ANDA BERHASIL MEMBUAT SINGLE LINGKED LIST.