

**LAPORAN PRAKTIKUM
“UJIAN PRAKTIKUM”**



**Disusun Oleh:
Aji Prasetyo Nugroho - 2211104049
S1SE-07-2**

**Assisten Praktikum :
Aldi Putra
Andini Nur Hidayah**

**Dosen :
Wahyu Andi Saputra, S.Pd., M.Eng**

**PROGRAM STUDI S1 SOFTWARE ENGINEERING
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY
PURWOKERTO
2024**

A. Program

Source Code

Main.cpp :

```
#include <iostream>
#include <string>
using namespace std;

// Aji Prasetyo Nugroho_2211104049_S1SE-07-2
struct Mahasiswa {
    string nama;
    string nim;
    string kelas;
    int nilaiAsesmen;
    int nilaiPraktikum;
};

struct Element {
    Mahasiswa info;
    Element* next;
};

typedef Element* address;
typedef address List;

address newElement(Mahasiswa data) {
    address p = new Element;
    p->info = data;
    p->next = nullptr;
    return p;
}

List createNewList() {
    return nullptr;
}

bool isEmpty(List a) {
    return (a == nullptr);
}
```

```

void insertFirst(List& a, address p) {
    if (isEmpty(a)) {
        a = p;
    } else {
        p->next = a;
        a = p;
    }
}

```

```

void addMahasiswa(List& a, int N) {
    for (int i = 0; i < N; ++i) {
        Mahasiswa data;
        cout << "Mahasiswa ke-" << (i + 1) << ":" << endl;

        cout << "Nama: ";
        cin.ignore();
        getline(cin, data.nama);

        while (true) {
            cout << "NIM (max 15 angka): ";
            cin >> data.nim;

            if (data.nim.length() > 15) {
                cout << "NIM terlalu panjang. Maksimal 15 angka. Silakan ulangi.\n";
            } else if (data.nim.find_first_not_of("0123456789") != string::npos) {
                cout << "NIM harus berupa angka. Silakan ulangi.\n";
            } else {
                break;
            }
        }

        cout << "Kelas: ";
        cin.ignore();
        getline(cin, data.kelas);

        cout << "Nilai Asesmen: ";
        cin >> data.nilaiAsesmen;

        cout << "Nilai Praktikum: ";
        cin >> data.nilaiPraktikum;

        address p = newElement(data);

        if (stoll(data.nim) % 2 == 1) {
            insertFirst(a, p);
        }
        cout << endl;
    }
}

```

```

void printList(List a) {
    if (isEmpty(a)) {
        cout << "List kosong" << endl;
        return;
    }

    address temp = a;
    while (temp != nullptr) {
        cout << "Nama: " << temp->info.nama << ", NIM: " << temp->info.nim
            << ", Kelas: " << temp->info.kelas
            << ", Nilai Asesmen: " << temp->info.nilaiAsesmen
            << ", Nilai Praktikum: " << temp->info.nilaiPraktikum << endl;
        temp = temp->next;
    }
}

```

```

Mahasiswa findMaxNilaiAsesmen(List a) {
    if (isEmpty(a)) {
        cout << "List kosong" << endl;
        return {};
    }

    address temp = a;
    Mahasiswa maxMahasiswa = temp->info;

    while (temp != nullptr) {
        if (temp->info.nilaiAsesmen > maxMahasiswa.nilaiAsesmen) {
            maxMahasiswa = temp->info;
        }
        temp = temp->next;
    }

    return maxMahasiswa;
}

```

```

void deleteDuplicate(List& a) {
    if (isEmpty(a)) return;

    address current = a;
    while (current != nullptr) {
        address runner = current;
        while (runner->next != nullptr) {
            if (runner->next->info.nim == current->info.nim) {
                address duplicate = runner->next;
                runner->next = runner->next->next;
                delete duplicate;
            } else {
                runner = runner->next;
            }
        }
        current = current->next;
    }
}

```

```

int main() {
    List mahasiswaList = createNewList();
    int N;

    cout << "Masukkan jumlah mahasiswa: ";
    cin >> N;

    addMahasiswa(mahasiswaList, N);
    printList(mahasiswaList);

    Mahasiswa maxMahasiswa = findMaxNilaiAsesmen(mahasiswaList);
    cout << "Mahasiswa dengan nilai asesmen tertinggi: " << maxMahasiswa.nama << endl;

    deleteDuplicate(mahasiswaList);
    cout << "List setelah menghapus duplikat:" << endl;
    printList(mahasiswaList);

    return 0;
}

```

Output :

Main.cpp

```
Masukkan jumlah mahasiswa: 2
Mahasiswa ke-1:
Nama: Aji
NIM: 2211104049
Kelas: S1SE0702
Nilai Asesmen: 100
Nilai Praktikum: 100

Mahasiswa ke-2:
Nama: Yogi
NIM: 2211104061
Kelas: S1SE0702
Nilai Asesmen: 90
Nilai Praktikum: 100

Nama: Yogi, NIM: 2211104061, Kelas: S1SE0702, Nilai Asesmen: 90, Nilai Praktikum: 100
Nama: Aji, NIM: 2211104049, Kelas: S1SE0702, Nilai Asesmen: 100, Nilai Praktikum: 100
Mahasiswa dengan nilai asesmen tertinggi: Aji
List setelah menghapus duplikat:
Nama: Yogi, NIM: 2211104061, Kelas: S1SE0702, Nilai Asesmen: 90, Nilai Praktikum: 100
Nama: Aji, NIM: 2211104049, Kelas: S1SE0702, Nilai Asesmen: 100, Nilai Praktikum: 100
PS D:\Praktikum STD_2211104049>
```

A. Penjelasan Program

Program ini adalah implementasi struktur data Single Linked List untuk menyimpan data mahasiswa yang meliputi nama, NIM, kelas, nilai asesmen, dan nilai praktikum. Program memungkinkan pengguna untuk menambahkan data mahasiswa ke dalam list menggunakan metode *Insert First*. Selain itu, program dapat mencetak seluruh data mahasiswa, mencari dan menampilkan mahasiswa dengan nilai asesmen tertinggi, serta menghapus data duplikat berdasarkan NIM. Program ini dirancang untuk menangani validasi input seperti panjang NIM maksimum 15 karakter dan memastikan hanya angka yang dapat diterima sebagai NIM. Setelah data dimasukkan, pengguna dapat melihat data mahasiswa dalam list dan hasil setelah penghapusan duplikat.

