Nama : Reyner Atira Prasetyo

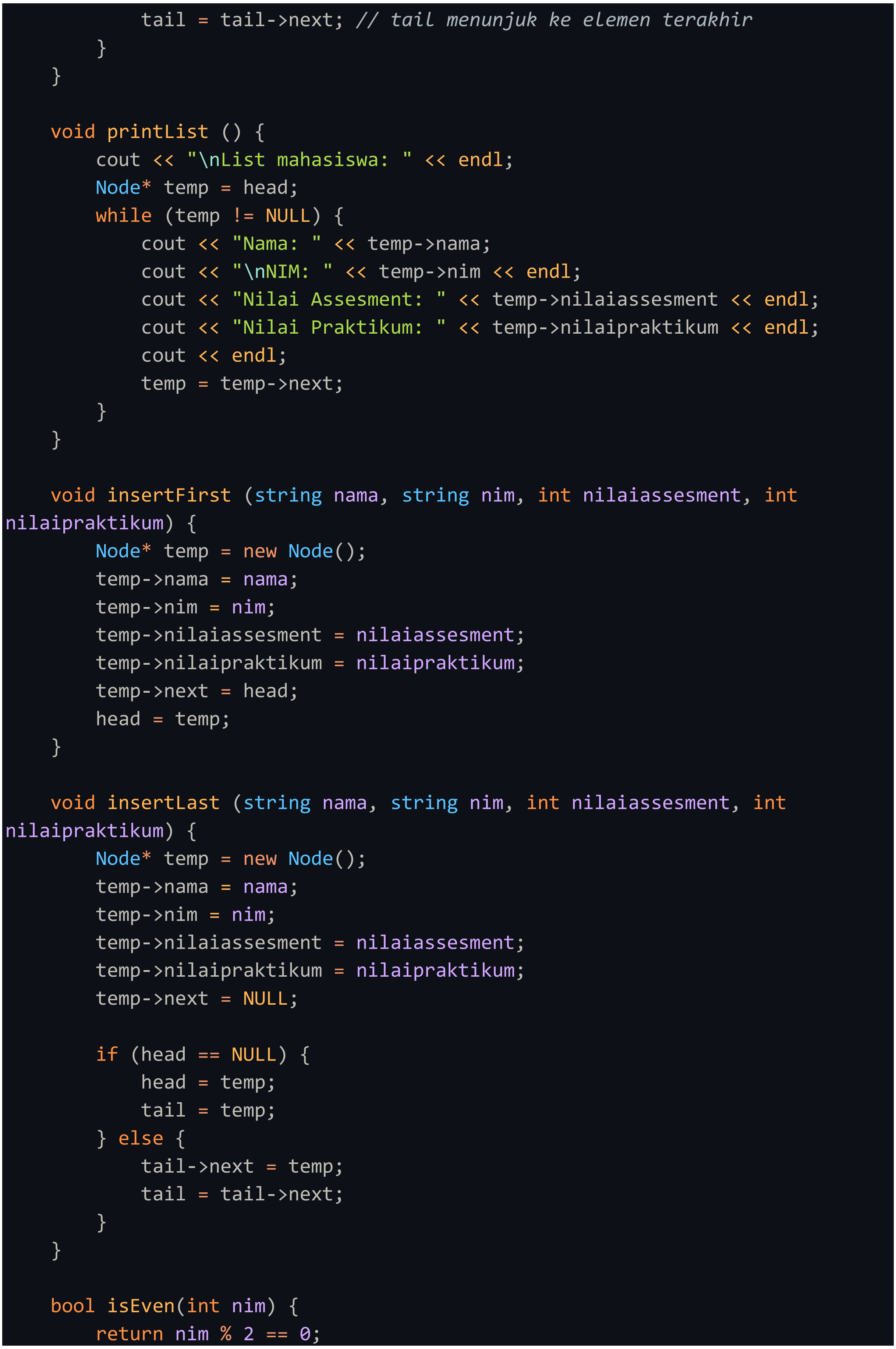
NIM   : 2311104057

Kelas : SE0702

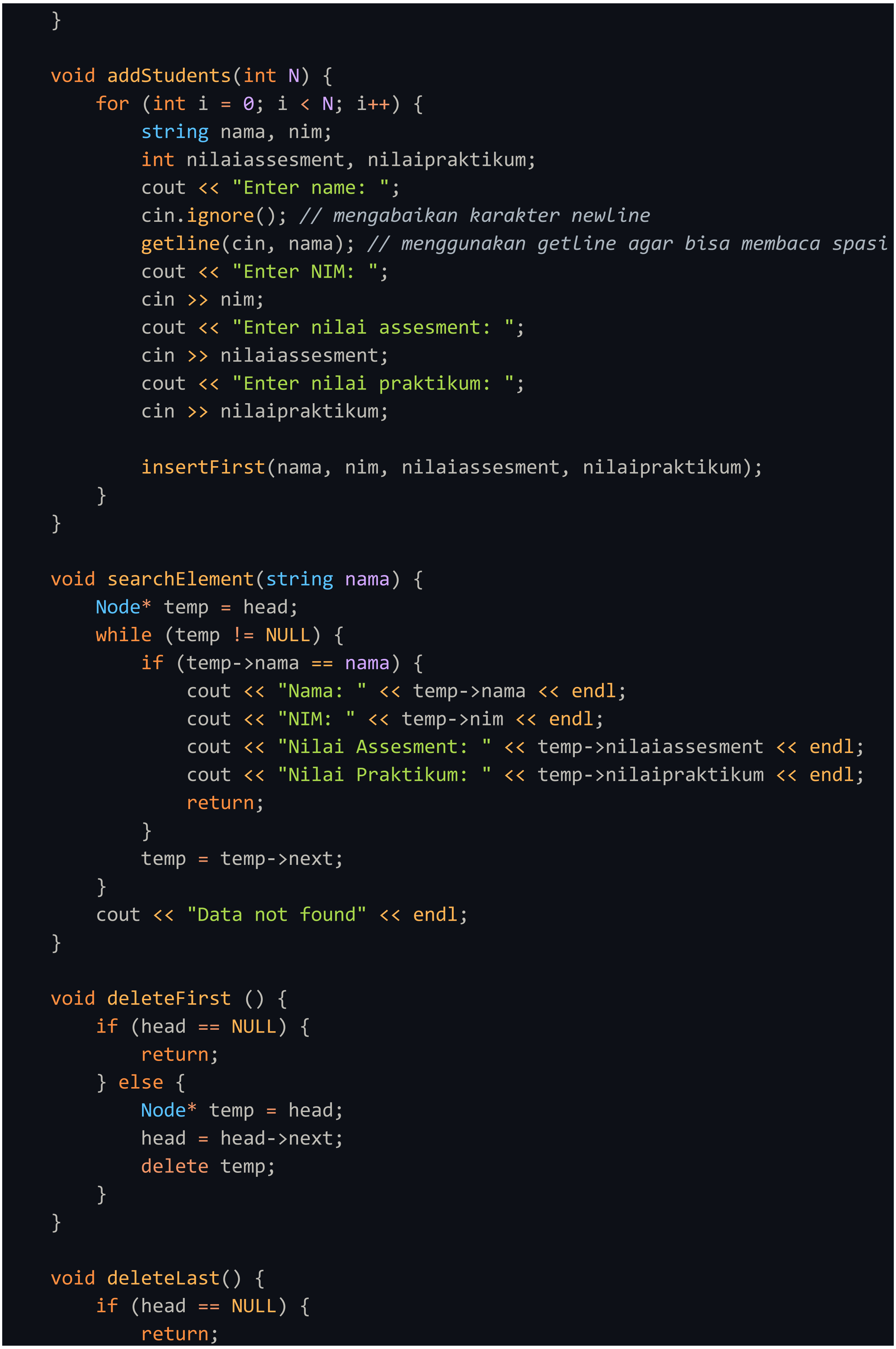
Kode Program :

1. sll.cpp

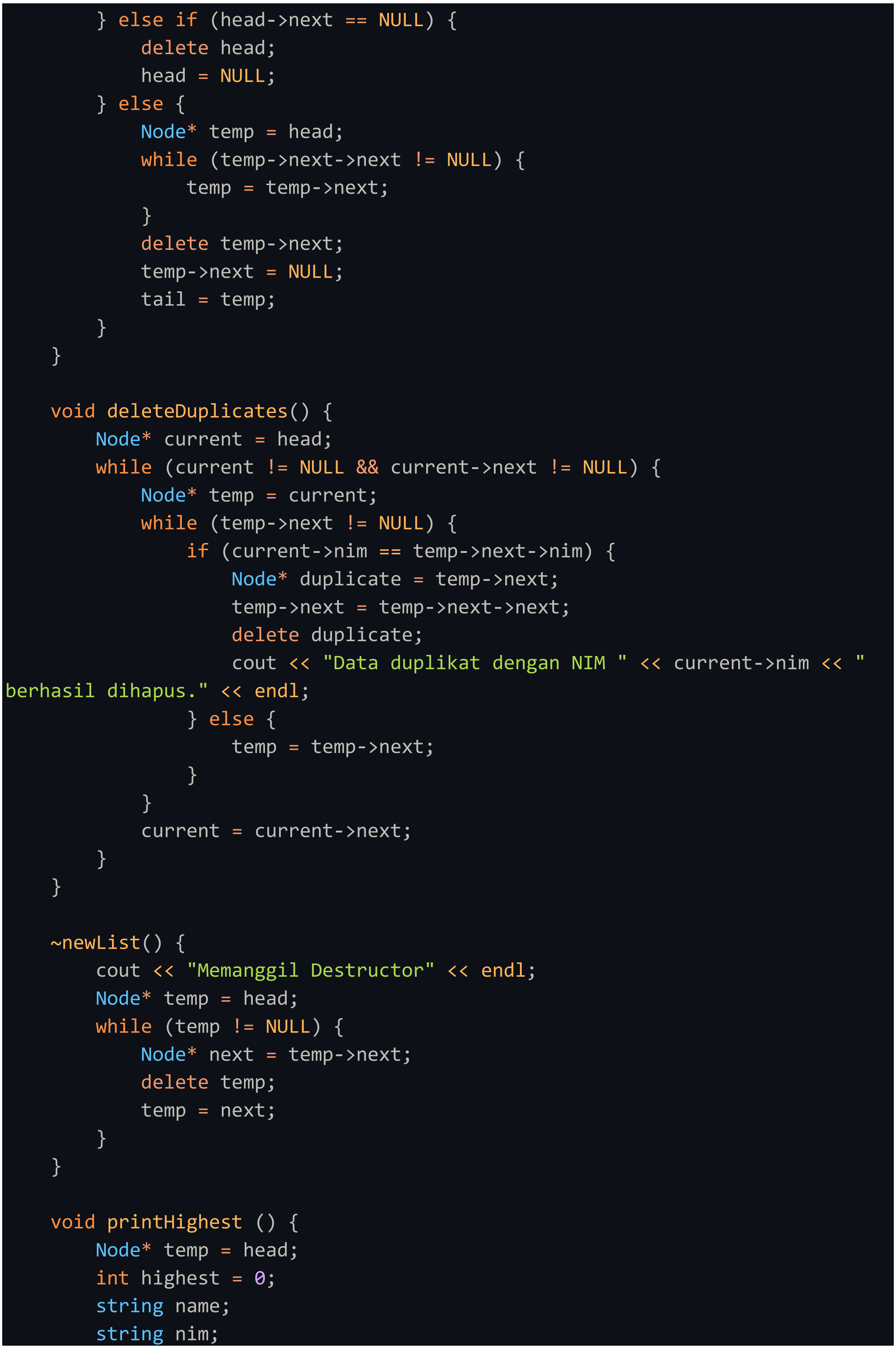
```
// Reyner Atira Prasetyo
// SE0702
// 2311104057

#include <iostream>
#include <string>

using namespace std;

class Node {
    public:
        string nama, nim;
        int nilaiassesment, nilaipraktikum;
        Node *next;
};

class newList {
public:
    Node* head;
    Node* tail;

    newList() {
        head = NULL;
        tail = NULL;
    }

    void newElement(string nama, string nim, int nilaiassesment, int 
nilaipraktikum) {
        Node* temp = new Node();
        temp->nama = nama;
        temp->nim = nim;
        temp->nilaiassesment = nilaiassesment;
        temp->nilaipraktikum = nilaipraktikum;
        temp->next = NULL;

        if (head == NULL) { // list kosong
            head = temp;
            tail = temp;
        } else {
            tail->next = temp; // menambahkan elemen di akhir
```



```
            tail = tail->next; // tail menunjuk ke elemen terakhir
        }
    }

    void printList () {
        cout << "\nList mahasiswa: " << endl;
        Node* temp = head;
        while (temp != NULL) {
            cout << "Nama: " << temp->nama;
            cout << "\nNIM: " << temp->nim << endl;
            cout << "Nilai Assesment: " << temp->nilaiassesment << endl;
            cout << "Nilai Praktikum: " << temp->nilaipraktikum << endl;
            cout << endl;
            temp = temp->next;
        }
    }

    void insertFirst (string nama, string nim, int nilaiassesment, int
nilaipraktikum) {
        Node* temp = new Node();
        temp->nama = nama;
        temp->nim = nim;
        temp->nilaiassesment = nilaiassesment;
        temp->nilaipraktikum = nilaipraktikum;
        temp->next = head;
        head = temp;
    }

    void insertLast (string nama, string nim, int nilaiassesment, int
nilaipraktikum) {
        Node* temp = new Node();
        temp->nama = nama;
        temp->nim = nim;
        temp->nilaiassesment = nilaiassesment;
        temp->nilaipraktikum = nilaipraktikum;
        temp->next = NULL;

        if (head == NULL) {
            head = temp;
            tail = temp;
        } else {
            tail->next = temp;
            tail = tail->next;
        }
    }

    bool isEven(int nim) {
        return nim % 2 == 0;
```



```
    }

    void addStudents(int N) {
        for (int i = 0; i < N; i++) {
            string nama, nim;
            int nilaiassesment, nilaipraktikum;
            cout << "Enter name: ";
            cin.ignore(); // mengabaikan karakter newline
            getline(cin, nama); // menggunakan getline agar bisa membaca spasi
            cout << "Enter NIM: ";
            cin >> nim;
            cout << "Enter nilai assesment: ";
            cin >> nilaiassesment;
            cout << "Enter nilai praktikum: ";
            cin >> nilaipraktikum;

            insertFirst(nama, nim, nilaiassesment, nilaipraktikum);
        }
    }

    void searchElement(string nama) {
        Node* temp = head;
        while (temp != NULL) {
            if (temp->nama == nama) {
                cout << "Nama: " << temp->nama << endl;
                cout << "NIM: " << temp->nim << endl;
                cout << "Nilai Assesment: " << temp->nilaiassesment << endl;
                cout << "Nilai Praktikum: " << temp->nilaipraktikum << endl;
                return;
            }
            temp = temp->next;
        }
        cout << "Data not found" << endl;
    }

    void deleteFirst () {
        if (head == NULL) {
            return;
        } else {
            Node* temp = head;
            head = head->next;
            delete temp;
        }
    }

    void deleteLast() {
        if (head == NULL) {
            return;
```



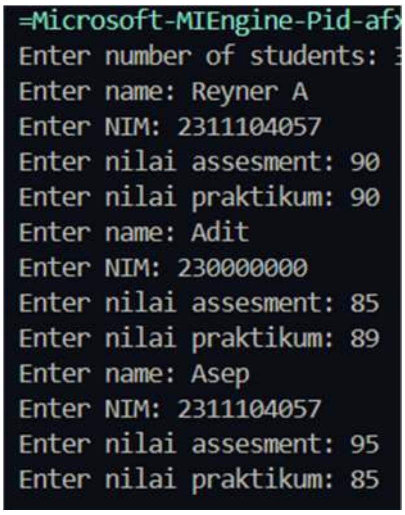
```
        } else if (head->next == NULL) {
            delete head;
            head = NULL;
        } else {
            Node* temp = head;
            while (temp->next->next != NULL) {
                temp = temp->next;
            }
            delete temp->next;
            temp->next = NULL;
            tail = temp;
        }
    }

    void deleteDuplicates() {
        Node* current = head;
        while (current != NULL && current->next != NULL) {
            Node* temp = current;
            while (temp->next != NULL) {
                if (current->nim == temp->next->nim) {
                    Node* duplicate = temp->next;
                    temp->next = temp->next->next;
                    delete duplicate;
                    cout << "Data duplikat dengan NIM " << current->nim << "
berhasil dihapus." << endl;
                } else {
                    temp = temp->next;
                }
            }
            current = current->next;
        }
    }

    ~newList() {
        cout << "Memanggil Destructor" << endl;
        Node* temp = head;
        while (temp != NULL) {
            Node* next = temp->next;
            delete temp;
            temp = next;
        }
    }

    void printHighest () {
        Node* temp = head;
        int highest = 0;
        string name;
        string nim;
        while (temp != NULL) {
            if (temp->nilaiassesment > highest) {
                highest = temp->nilaiassesment;
                name = temp->nama;
                nim = temp->nim;
            }
            temp = temp->next;
        }
        cout << "Mahasiswa dengan skor tertinggi: " << name << ", NIM: " <<
nim << ", dengan skor: " << highest << endl;
    }
};
```

2. main.cpp

```
// Reyner Atira Prasetyo
// SE0702
// 2311104057

#include <iostream>
#include "sll.cpp"

using namespace std;

int main() {
    newList L;
    int N;
    cout << "Enter number of students: ";
    cin >> N;
    L.addStudents(N);
    L.printList();
    L.printHighest();
    cout << "\n Menghapus duplikat... " << endl;
    L.deleteDuplicates();
    cout << "Setelah penghapusan duplikat: " << endl;
    L.printList();
    return 0;
}
```

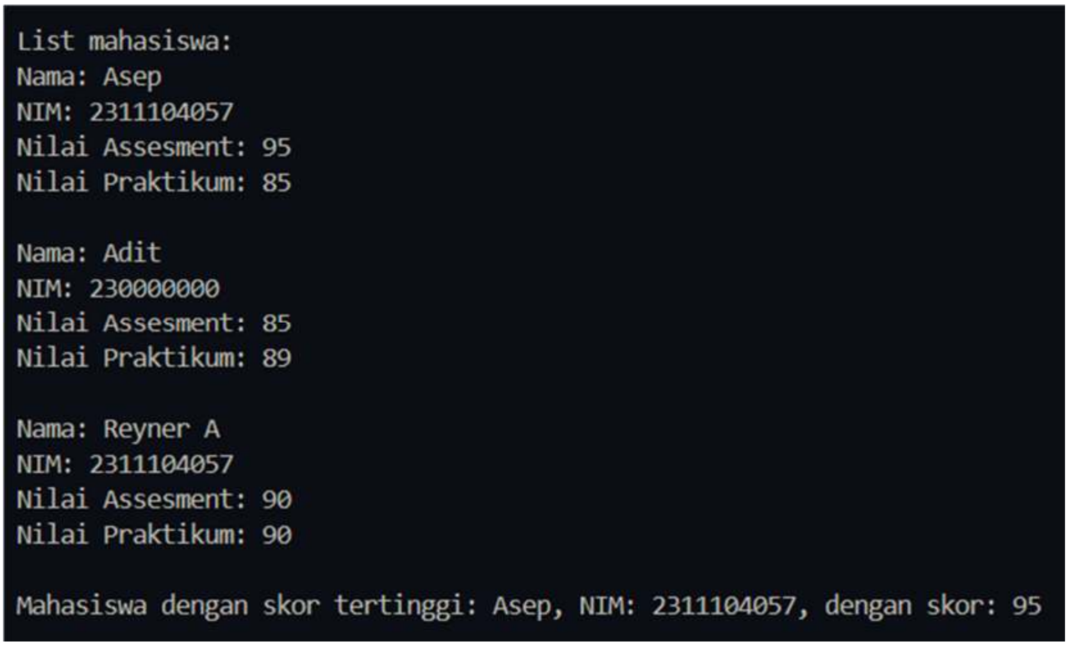
Output :

1. insertStudent



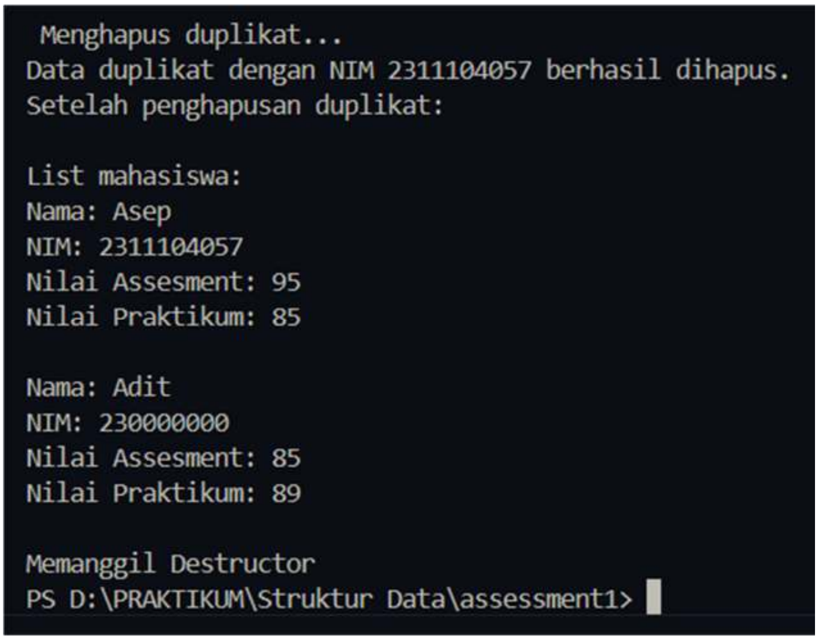
```
=Microsoft-MIEngine-Pid-af
Enter number of students:
Enter name: Reyner A
Enter NIM: 2311104057
Enter nilai assesment: 90
Enter nilai praktikum: 90
Enter name: Adit
Enter NIM: 230000000
Enter nilai assesment: 85
Enter nilai praktikum: 89
Enter name: Asep
Enter NIM: 2311104057
Enter nilai assesment: 95
Enter nilai praktikum: 85
```

2. print info



```
List mahasiswa:
Nama: Asep
NIM: 2311104057
Nilai Assesment: 95
Nilai Praktikum: 85

Nama: Adit
NIM: 230000000
Nilai Assesment: 85
Nilai Praktikum: 89

Nama: Reyner A
NIM: 2311104057
Nilai Assesment: 90
Nilai Praktikum: 90

Mahasiswa dengan skor tertinggi: Asep, NIM: 2311104057, dengan skor: 95
```

3. delete duplicate dan print info setelah delete



```
 Menghapus duplikat...
Data duplikat dengan NIM 2311104057 berhasil dihapus.
Setelah penghapusan duplikat:

List mahasiswa:
Nama: Asep
NIM: 2311104057
Nilai Assesment: 95
Nilai Praktikum: 85

Nama: Adit
NIM: 230000000
Nilai Assesment: 85
Nilai Praktikum: 89

Memanggil Destructor
PS D:\PRAKTIKUM\Struktur Data\assessment1>
```