LAPORAN PRAKTIKUM
Tugas Pendahuluan Modul 05
“Single Linked List”

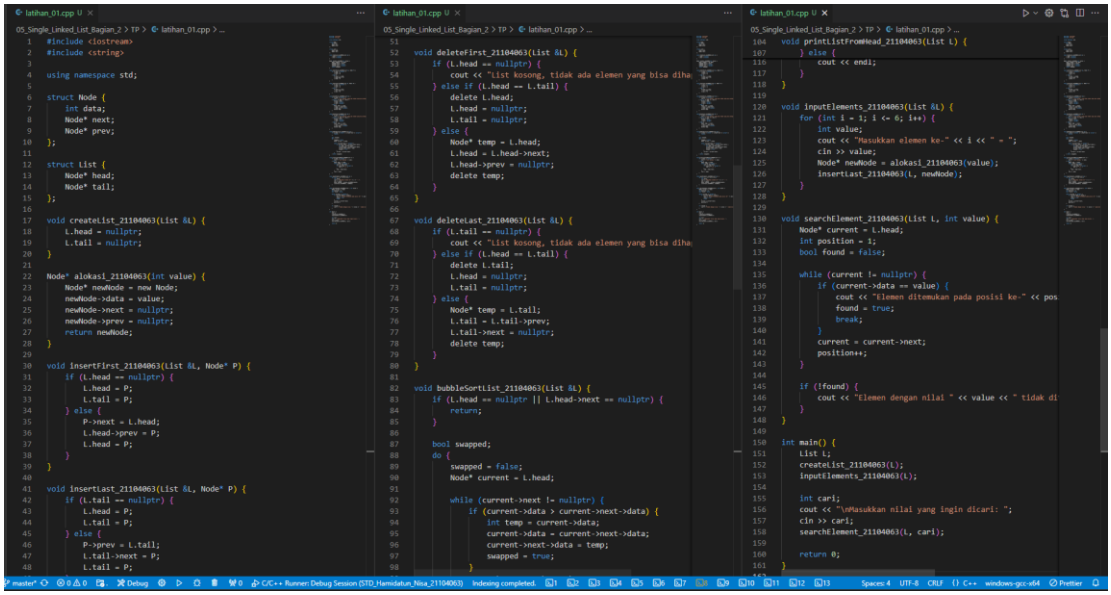
Disusun Oleh:
Hamidatun Nisa - 21104063
Struktur Data SE07-01

Dosen :
Yudha Islami Sulistya

PROGRAM STUDI S1 SOFTWARE ENGINEERING
FAKULTAS INFORMATIKA
UNIVERSITAS TELKOM PURWOKERTO
2024

1. Mencari Elemen Tertentu dalam SLL

   Code :



```cpp
#include <iostream>
#include <string>

using namespace std;

struct Node {
    int data;
    Node* next;
    Node* prev;
};

struct List {
    Node* head;
    Node* tail;
};

void createList_21104063(List &L) {
    L.head = nullptr;
    L.tail = nullptr;
}

Node* alokasi_21104063(int value) {
    Node* newNode = new Node;
    newNode->data = value;
    newNode->next = nullptr;
    newNode->prev = nullptr;
    return newNode;
}

void insertFirst_21104063(List &L, Node* P) {
    if (L.head == nullptr) {
        L.head = P;
        L.tail = P;
    } else {
        P->next = L.head;
        L.head->prev = P;
        L.head = P;
    }
}

void insertLast_21104063(List &L, Node* P) {
    if (L.tail == nullptr) {
        L.head = P;
        L.tail = P;
    } else {
        P->prev = L.tail;
        L.tail->next = P;
        L.tail = P;
    }
}

void deleteFirst_21104063(List &L) {
    if (L.head == nullptr) {
        cout << "List kosong, tidak ada elemen yang bisa dihapus.\n";
    } else if (L.head == L.tail) {
        delete L.head;
        L.head = nullptr;
        L.tail = nullptr;
    } else {
        Node* temp = L.head;
        L.head = L.head->next;
        L.head->prev = nullptr;
        delete temp;
    }
}

void deleteLast_21104063(List &L) {
    if (L.tail == nullptr) {
        cout << "List kosong, tidak ada elemen yang bisa dihapus.\n";
    } else if (L.head == L.tail) {
        delete L.tail;
        L.head = nullptr;
        L.tail = nullptr;
    } else {
        Node* temp = L.tail;
        L.tail = L.tail->prev;
        L.tail->next = nullptr;
        delete temp;
    }
}

void bubbleSortList_21104063(List &L) {
    if (L.head == nullptr || L.head->next == nullptr) {
        return;
    }

    bool swapped;
    do {
        swapped = false;
        Node* current = L.head;

        while (current->next != nullptr) {
            if (current->data > current->next->data) {
                int temp = current->data;
                current->data = current->next->data;
                current->next->data = temp;
                swapped = true;
            }
            current = current->next;
        }
    } while (swapped);
}

void printListFromHead_21104063(List L) {
    } else {
        cout << endl;
    }
}

void inputElements_21104063(List &L) {
    for (int i = 1; i <= 6; i++) {
        int value;
        cout << "Masukkan elemen ke-" << i << " = ";
        cin >> value;
        Node* newNode = alokasi_21104063(value);
        insertLast_21104063(L, newNode);
    }
}

void searchElement_21104063(List L, int value) {
    Node* current = L.head;
    int position = 1;
    bool found = false;

    while (current != nullptr) {
        if (current->data == value) {
            cout << "Elemen ditemukan pada posisi ke-" << pos
            found = true;
            break;
        }
        current = current->next;
        position++;
    }

    if (!found) {
        cout << "Elemen dengan nilai " << value << " tidak di
    }
}

int main() {
    List L;
    createList_21104063(L);
    inputElements_21104063(L);

    int cari;
    cout << "\nMasukkan nilai yang ingin dicari: ";
    cin >> cari;
    searchElement_21104063(L, cari);

    return 0;
}
```

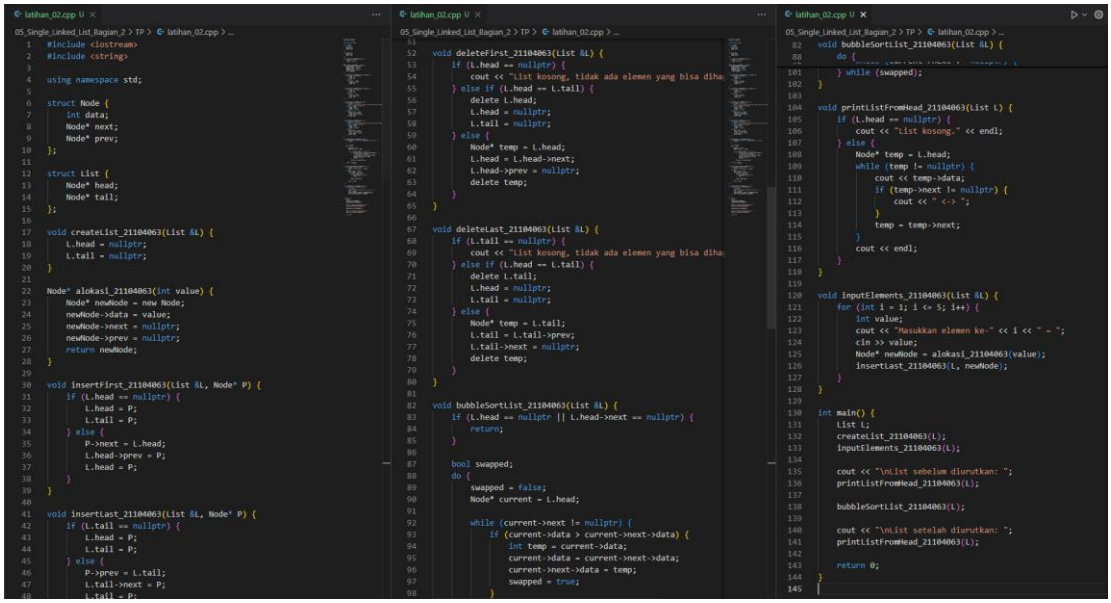
   Output :

```
Masukkan elemen ke-1: 30
Masukkan elemen ke-2: 45
Masukkan elemen ke-3: 21
Masukkan elemen ke-4: 40
Masukkan elemen ke-5: 89
Masukkan elemen ke-6: 76

Masukkan nilai yang ingin dicari: 76
Elemen ditemukan pada posisi ke-6 dengan alamat: 0x73edd0
PS D:\= Collage\Semester 7\Praktikum Struktur Data\STD_Hamidatun_Nisa_21104063>
```

```
Masukkan elemen ke-1: 60
Masukkan elemen ke-2: 32
Masukkan elemen ke-3: 56
Masukkan elemen ke-4: 86
Masukkan elemen ke-5: 46
Masukkan elemen ke-6: 33

Masukkan nilai yang ingin dicari: 10
Elemen dengan nilai 10 tidak ditemukan dalam list.
PS D:\= Collage\Semester 7\Praktikum Struktur Data\STD_Hamidatun_Nisa_21104063>
```

2. Mengurutkan List Menggunakan Bubble Sort

Code :

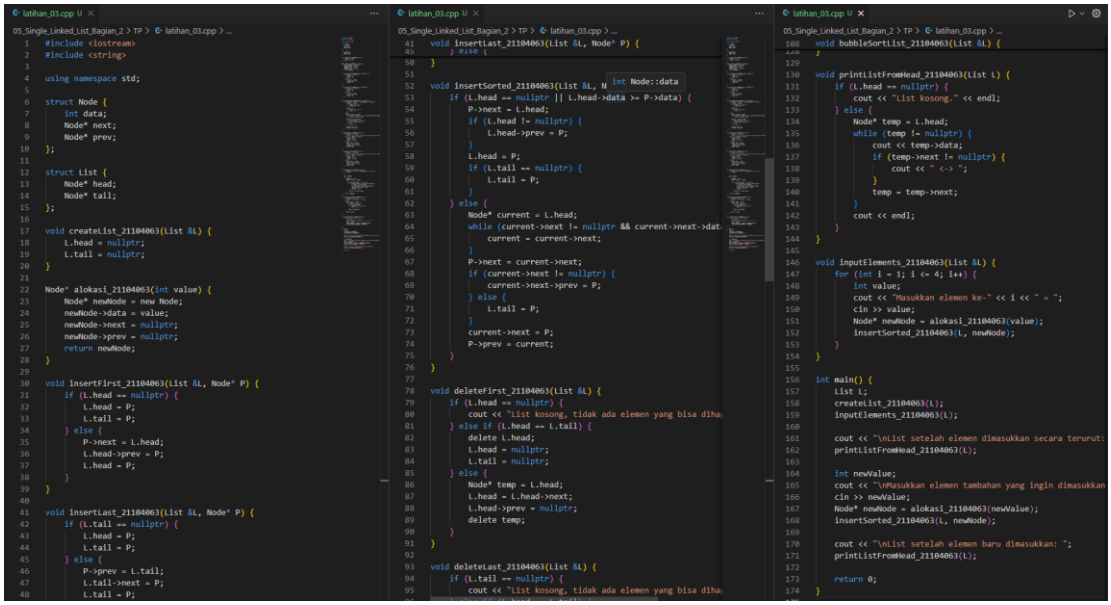


Output :

```
Masukkan elemen ke-1 = 54
Masukkan elemen ke-2 = 32
Masukkan elemen ke-3 = 67
Masukkan elemen ke-4 = 53
Masukkan elemen ke-5 = 66

List sebelum diurutkan: 54 <-> 32 <-> 67 <-> 53 <-> 66

List setelah diurutkan: 32 <-> 53 <-> 54 <-> 66 <-> 67
PS D:\= Collage\Semester 7\Praktikum Struktur Data\STD_Hamidatun_Nisa_21104063>
```

3. Menambahkan Elemen Secara Terurut

Code :



Output :

```
Masukkan elemen ke-1: 89
Masukkan elemen ke-2: 65
Masukkan elemen ke-3: 34
Masukkan elemen ke-4: 77

List setelah elemen dimasukkan secara terurut:
34 65 77 89

Masukkan elemen tambahan yang ingin dimasukkan secara terurut: 76

List setelah elemen baru dimasukkan:
34 65 76 77 89
PS D:\= Collage\Semester 7\Praktikum Struktur Data\STD_Hamidatun_Nisa_21104063>
```