Nama : Damar Galih

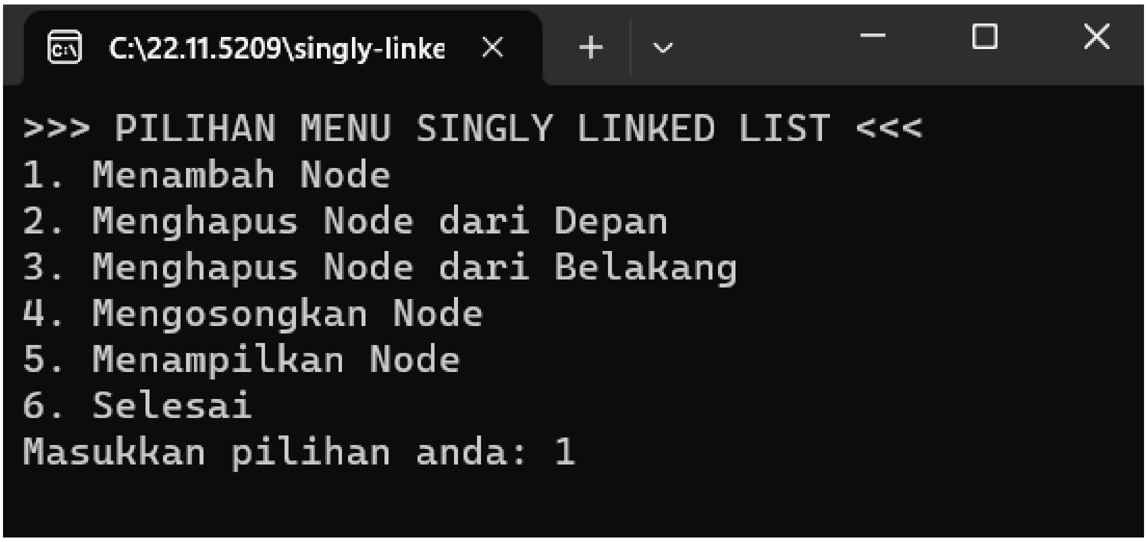
NIM : 22.11.5209

- **Singly-linked-list1-5209**

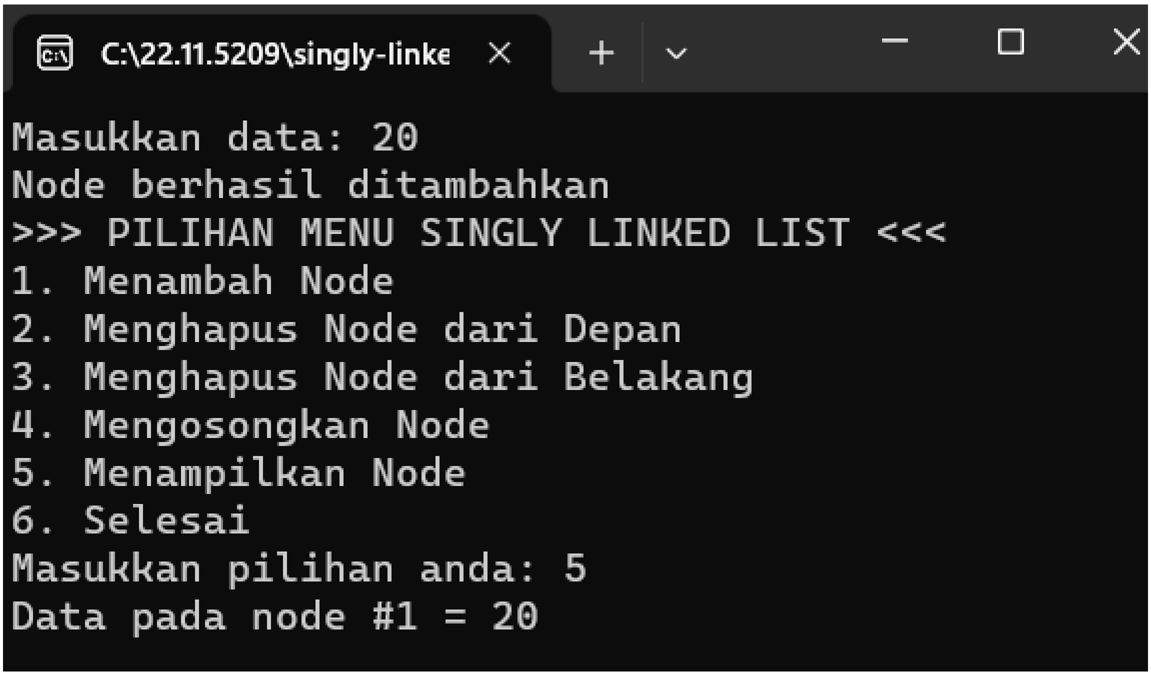
```
#include <iostream>
using namespace std;

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

Node* headNode = NULL;
Node* currentNode = NULL;

bool isEmpty() {
    return (headNode == NULL);
}

void insertNode(int data) {
    Node* newNode = new Node;
    newNode->data = data;
    newNode->next = NULL;
    if (isEmpty()) {
        headNode = newNode;
        currentNode = newNode;
    }
    else {
        currentNode->next = newNode;
        currentNode = newNode;
    }
    cout << "Node berhasil ditambahkan" << endl;
}

void deleteFromFront() {
    if (!isEmpty()) {
        Node* temp = headNode;
        headNode = headNode->next;
        delete temp;
        cout << "Node berhasil dihapus" << endl;
    }
}

void deleteFromEnd() {
    if (!isEmpty()) {
        if (headNode == currentNode) {
            delete headNode;
            headNode = NULL;
            currentNode = NULL;
        }
        else {
            Node* temp = headNode;
            while (temp->next != currentNode) {
                temp = temp->next;
            }

            delete currentNode;
            currentNode = temp;
            currentNode->next = NULL;
        }
        cout << "Node berhasil dihapus" << endl;
    }
}
```

```
void deleteAll() {
    if (!isEmpty()) {
        Node* temp = headNode;
        while (temp != NULL) {
            headNode = headNode->next;
            delete temp;
            temp = headNode;
        }

        currentNode = NULL;
        cout << "Semua node berhasil dihapus" << endl;
    }
}

void printNode() {
    if (!isEmpty()) {
        int i = 1;
        Node* temp = headNode;
        while (temp != NULL) {
            cout << "Data pada node #" << i << " = " << temp->data << endl;
            temp = temp->next;
            i++;
        }
    }
}

int main() {
    int choice, input;
    menu:
    cout << ">>> PILIHAN MENU SINGLY LINKED LIST <<<" << endl;
    cout << "1. Menambah Node" << endl;
    cout << "2. Menghapus Node dari Depan" << endl;
    cout << "3. Menghapus Node dari Belakang" << endl;
    cout << "4. Mengosongkan Node" << endl;
    cout << "5. Menampilkan Node" << endl;
    cout << "6. Selesai" << endl;
    cout << "Masukkan pilihan anda: ";
    cin >> choice;
    switch (choice) {
    case 1:
        system("cls");
        cout << "Masukkan data: ";
        cin >> input;
        insertNode(input);
        goto menu;
        break;
    case 2:
        deleteFromFront();
        goto menu;
         break;
    case 3:
        deleteFromEnd();
        goto menu;
        break;
    case 4:
        deleteAll();
        goto menu;
        break;
    case 5:
        printNode();
        break;
    case 6:
         system("cls");
         break;
    }
}
```

- **Output**



```
>>> PILIHAN MENU SINGLY LINKED LIST <<<
1. Menambah Node
2. Menghapus Node dari Depan
3. Menghapus Node dari Belakang
4. Mengosongkan Node
5. Menampilkan Node
6. Selesai
Masukkan pilihan anda: 1
```



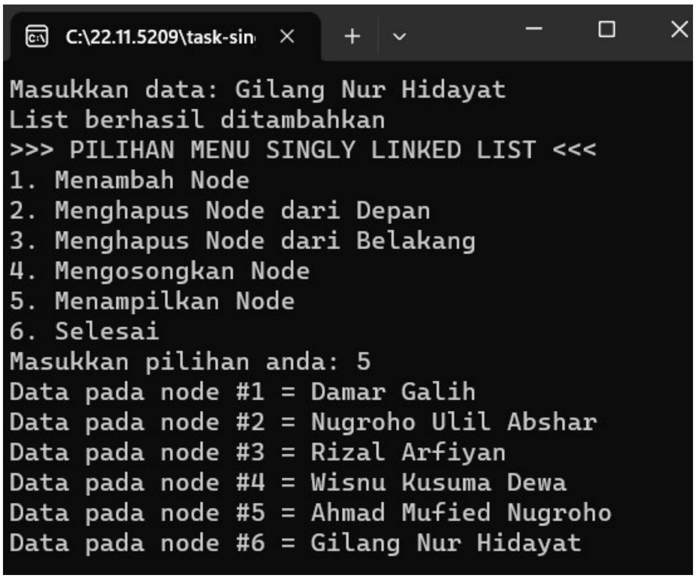
```
Masukkan data: 20
Node berhasil ditambahkan
>>> PILIHAN MENU SINGLY LINKED LIST <<<
1. Menambah Node
2. Menghapus Node dari Depan
3. Menghapus Node dari Belakang
4. Mengosongkan Node
5. Menampilkan Node
6. Selesai
Masukkan pilihan anda: 5
Data pada node #1 = 20
```

- **Task-singly-linked-list1-5209**

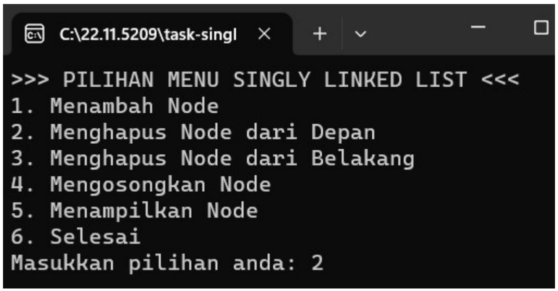
```
#include <iostream>
using namespace std;

struct Node {
    string data;
    Node* next;
};

Node* headNode = NULL;
Node* currentNode = NULL;

bool isEmpty() {
    return (headNode == NULL);
}

void insertNode(string data) {
    Node* newNode = new Node;
    newNode->data = data;
    newNode->next = NULL;
    if (isEmpty()) {
        headNode = newNode;
        currentNode = newNode;
    }
    else {
        currentNode->next = newNode;
        currentNode = newNode;
    }
    cout << "List berhasil ditambahkan" << endl;
}

void deleteFromFront() {
    if (!isEmpty()) {
        Node* temp = headNode;
        headNode = headNode->next;
        delete temp;
        cout << "Node berhasil dihapus" << endl;
    }
    else {
        cout << "List masih kosong" << endl;
    }
}

void deleteFromEnd() {
    if (!isEmpty()) {
        if (headNode == currentNode) {
            delete headNode;
            headNode = NULL;
            currentNode = NULL;
        }
        else {
            Node* temp = headNode;
            while (temp->next != currentNode) {
                temp = temp->next;
            }
            delete currentNode;
            currentNode = temp;
            currentNode->next = NULL;
        }
        cout << "Node berhasil dihapus" << endl;
    }
    else {
        cout << "List masih kosong" << endl;
    }
}
```

```
void deleteAll() {
    if (!isEmpty()) {
        Node* temp = headNode;
        while (temp != NULL) {
            headNode = headNode->next;
            delete temp;
            temp = headNode;
        }
        currentNode = NULL;
        cout << "Semua node berhasil dihapus" << endl;
    }
    else {
        cout << "List masih kosong" << endl;
    }
}

void printNode() {
    if (!isEmpty()) {
        int i = 1;
        Node* temp = headNode;
        while (temp != NULL) {
            cout << "Data pada node #" << i << " = " << temp->data << endl;
            temp = temp->next;
            i++;
        }
    }
    else {
        cout << "List masih kosong" << endl;
    }
}

int main() {
    int choice, sum;
    string input;
    menu:
    cout << ">>> PILIHAN MENU SINGLY LINKED LIST <<<" << endl;
    cout << "1. Menambah Node" << endl;
    cout << "2. Menghapus Node dari Depan" << endl;
    cout << "3. Menghapus Node dari Belakang" << endl;
    cout << "4. Mengosongkan Node" << endl;
    cout << "5. Menampilkan Node" << endl;
    cout << "6. Selesai" << endl;

    cout << "Masukkan pilihan anda: ";
    cin >> choice;

    switch (choice) {
    case 1:
        system("cls");
        cout << "Masukkan data: ";
        cin.ignore();
        getline(cin, input);
        insertNode(input);
        goto menu;
        break;
    case 2:
        system("cls");
        deleteFromFront();
        goto menu;
               break;
    case 3:
        system("cls");
        deleteFromEnd();
        goto menu;
        break;

    case 4:
        system("cls");
        deleteAll();
        goto menu;
        break;
    case 5:
        printNode();
        break;
    case 6:
        system("cls");
        break;
    default:
        cout << "Menu yang anda pilih tidak terdaftar !!!!" << endl;
    }
}
```

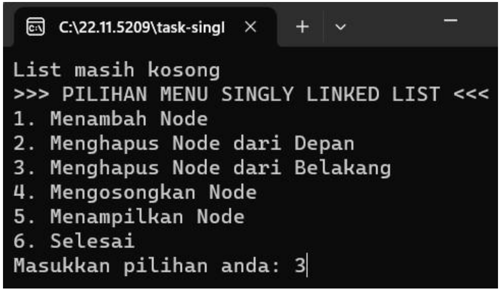
- **Output**



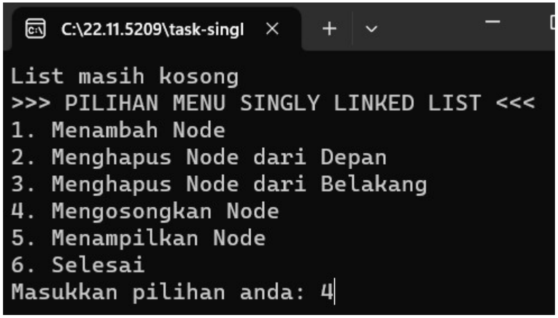
```
Masukkan data: Gilang Nur Hidayat
List berhasil ditambahkan
>>> PILIHAN MENU SINGLY LINKED LIST <<<
1. Menambah Node
2. Menghapus Node dari Depan
3. Menghapus Node dari Belakang
4. Mengosongkan Node
5. Menampilkan Node
6. Selesai
Masukkan pilihan anda: 5
Data pada node #1 = Damar Galih
Data pada node #2 = Nugroho Ulil Abshar
Data pada node #3 = Rizal Arfiyan
Data pada node #4 = Wisnu Kusuma Dewa
Data pada node #5 = Ahmad Mufied Nugroho
Data pada node #6 = Gilang Nur Hidayat
```



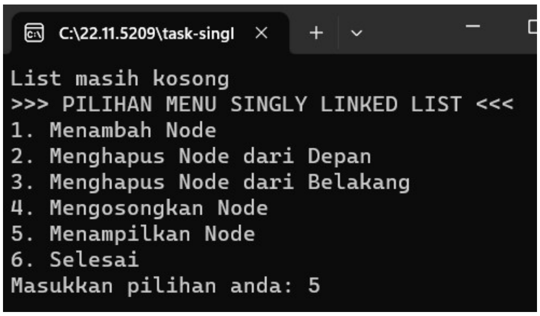
```
>>> PILIHAN MENU SINGLY LINKED LIST <<<
1. Menambah Node
2. Menghapus Node dari Depan
3. Menghapus Node dari Belakang
4. Mengosongkan Node
5. Menampilkan Node
6. Selesai
Masukkan pilihan anda: 2
```



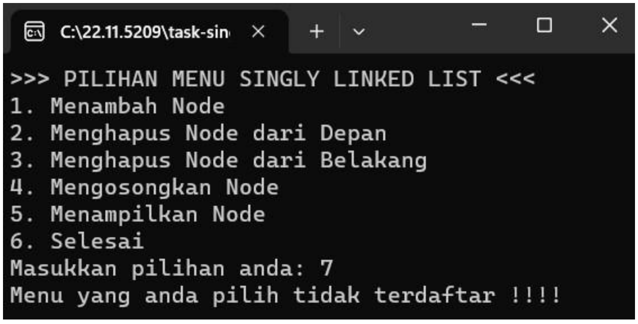
```
List masih kosong
>>> PILIHAN MENU SINGLY LINKED LIST <<<
1. Menambah Node
2. Menghapus Node dari Depan
3. Menghapus Node dari Belakang
4. Mengosongkan Node
5. Menampilkan Node
6. Selesai
Masukkan pilihan anda: 3
```



```
List masih kosong
>>> PILIHAN MENU SINGLY LINKED LIST <<<
1. Menambah Node
2. Menghapus Node dari Depan
3. Menghapus Node dari Belakang
4. Mengosongkan Node
5. Menampilkan Node
6. Selesai
Masukkan pilihan anda: 4
```



```
List masih kosong
>>> PILIHAN MENU SINGLY LINKED LIST <<<
1. Menambah Node
2. Menghapus Node dari Depan
3. Menghapus Node dari Belakang
4. Mengosongkan Node
5. Menampilkan Node
6. Selesai
Masukkan pilihan anda: 5
```



```
>>> PILIHAN MENU SINGLY LINKED LIST <<<
1. Menambah Node
2. Menghapus Node dari Depan
3. Menghapus Node dari Belakang
4. Mengosongkan Node
5. Menampilkan Node
6. Selesai
Masukkan pilihan anda: 7
Menu yang anda pilih tidak terdaftar !!!!
```