

STRUKTUR DATA
Tugas List Berkait 1



NAMA: Bagas Diatama Wardoyo
NPM: 140810230061

Dikumpulkan tanggal :
13 Maret 2024

Universitas Padjadjaran
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
Program Studi S-1 Teknik Informatika
2024

1. Soal 1

a. Source Code

```
/* Nama program : soal_1
   Nama          : Bagas Diatama Wardoyo
   NPM           : 140810230061
   Tanggal buat  : 01/04/2024
   Deskripsi     : Revisi kuis
   *****/

#include <iostream>
using namespace std;

//menyimpan informasi mahasiswa (NPM, nama, nilai)
struct Mahasiswa
{
    char NPM[8];
    char nama[20];
    int nilai;
};

//struktur data untuk node dalam Linked List
struct node
{
    Mahasiswa dataMahasiswa;
    node *nextNode;
};

/*pointerToList: pointer yang menunjuk ke struktur node
list: alias untuk pointerToList, digunakan untuk menunjuk ke head dari Linked
List*/
typedef node* pointerToList;
typedef pointerToList list;

//menginisialisasi head dari linked list menjadi NULL
void createHead(list &head){
    head = new node;
    head->nextNode = nullptr;
}

/*membuat elemen baru (node) dan meminta input data mahasiswa (nama, NPM,
nilai),
lalu mengembalikan pointer ke elemen baru tersebut*/
void createList(list &newNode){
    newNode = new node;
    cout << "Masukkan Nama   : "; cin >> newNode->dataMahasiswa.nama;
    cout << "Masukkan NPM    : "; cin >> newNode->dataMahasiswa.NPM;
    cout << "Masukkan nilai  : "; cin >> newNode->dataMahasiswa.nilai;
    newNode->nextNode=nullptr;
    cout << endl;
}
```

```

//menampilkan seluruh elemen dalam linked list (head) dengan format tabel
void trafsersal(pointerToList head){
    pointerToList ptrBantu = head;
    do
    {
        cout << "Nama      : " << ptrBantu->dataMahasiswa.nama << endl;
        cout << "NPM       : " << ptrBantu->dataMahasiswa.NPM << endl;
        cout << "NILAI    : " << ptrBantu->dataMahasiswa.nilai << endl;
        cout << endl;
        ptrBantu = ptrBantu->nextNode;
    } while (ptrBantu != nullptr);

    cout << endl;
}

//menghitung dan mengembalikan nilai rata-rata nilai mahasiswa dalam linked
list (head)
float average(pointerToList head){
    float jumlah, rata_rata;
    int n = 0;
    pointerToList ptrBantu = head;
    do
    {
        jumlah += ptrBantu->dataMahasiswa.nilai;
        ptrBantu = ptrBantu->nextNode;
        n++;
    } while (ptrBantu != nullptr);
    rata_rata = jumlah / n;
    return rata_rata;
}

int main(){

    //Membentuk list
    list head, p1, p2, p3; //Inisialisasi head, p1, p2, p3 sebagai pointer ke
node
    createHead(head); //Memanggil createHead(head) untuk menginisialisasi head
    createList(p1);
    createList(p2); //Memanggil createList(p1), createList(p2), createList(p3)
untuk membuat elemen baru dan menyimpannya di p1, p2, p3
    createList(p3);
    head = p1;
    p1->nextNode = p2;
    p2->nextNode = p3;

    trafsersal(head);
    cout << "Nilai rata - rata : " << average(head) << endl;

    //Mencetak B dengan 2 cara
    cout << "Cara 1 cetak B (Lewat p2) : " << p2->dataMahasiswa.nama << "
" << p2->dataMahasiswa.nilai << " " << p2->dataMahasiswa.NPM << endl;
    cout << "Cara 2 cetak B (Lewat p1) : " <<

```

```

p1->nextNode->dataMahasiswa.nama << " " << p1->nextNode->dataMahasiswa.nilai
<< " " << p1->nextNode->dataMahasiswa.NPM << endl;

//Mencetak C melalui p1
cout << "Cetak C : " <<p1->nextNode->nextNode->dataMahasiswa.nama <<
" " << p1->nextNode->nextNode->dataMahasiswa.nilai << " " <<
p1->nextNode->nextNode->dataMahasiswa.NPM << endl;

return 0;
}

```

b. Hasil Run

Masukkan Nama : **Bagas**
 Masukkan NPM : **002**
 Masukkan nilai : **80**

Masukkan Nama : **Diatama**
 Masukkan NPM : **003**
 Masukkan nilai : **100**

Masukkan Nama : **Wardoyo**
 Masukkan NPM : **008**
 Masukkan nilai : **35**

Nama : **Bagas**
 NPM : **002**
 NILAI : **80**

Nama : **Diatama**
 NPM : **003**
 NILAI : **100**

Nama : **Wardoyo**
 NPM : **008**
 NILAI : **35**

Nilai rata - rata : **71.6667**

Cara **1** cetak B (Lewat p2) : **Diatama 100 003**

Cara **2** cetak B (Lewat p1) : **Diatama 100 003**

Cetak C : **Wardoyo 35 008**

2. Soal 2

a. Source Code

```
/* Nama program : soal_2
   Nama          : Bagus Diatama Wardoyo
   NPM           : 140810230061
   Tanggal buat  : 01/04/2024
   Deskripsi     : Tugas List Berkait
   *****/

#include <iostream>
#include <string>

//menyimpan informasi dosen (NIP, nama, golongan)
struct dosen
{
    char NIP[3];
    std::string nama;
    int golongan;
};

//struktur data untuk node dalam Linked List
struct node
{
    dosen dataDosen;
    node *nextNode;
};

typedef node *pointerToNode; //pointer yang menunjuk ke struktur node
typedef pointerToNode list; //alias untuk pointerToNode, digunakan untuk
menunjuk ke head dari Linked List

//menginisialisasi head dari Linked List menjadi NULL
void createHead(list &head)
{
    head = nullptr;
}

//membuat elemen baru (node) dan meminta input data dosen (NIP, nama,
golongan), lalu mengembalikan pointer ke elemen baru tersebut
void createElement(pointerToNode &newNode)
{
    newNode = new node;
    std::cout << "Masukkan NIP      : ";
    std::cin >> newNode->dataDosen.NIP;
    std::cout << "Masukkan Nama      : ";
    std::cin.ignore();
    std::getline(std::cin, newNode->dataDosen.nama);
    std::cout << "Masukkan Golongan : ";
    std::cin >> newNode->dataDosen.golongan;
```

```

        newNode->nextNode = nullptr;
    }

    //menyisipkan elemen baru (newNode) di depan Linked List (head)
    void insertFirst(list &head, pointerToNode newNode)
    {
        if (head == nullptr)
        {
            head = newNode;
        }
        else
        {
            newNode->nextNode = head;
            head = newNode;
        }
    }

    //menghapus elemen pertama dari Linked List (head) dan mengembalikan pointer
    ke elemen selanjutnya
    void deleteFirst(list &head)
    {
        pointerToNode ptrHelp;
        ptrHelp = head;
        if (head == nullptr)
        {
            return;
        }
        else
        {
            head = ptrHelp->nextNode;
            ptrHelp = nullptr;
            delete ptrHelp;
        }
    }

    //mengembalikan string gaji berdasarkan golongan (if-else untuk pengecekan)
    std::string gajiGol(int gol)
    {
        std::string gaji;
        if (gol == 1)
        {
            gaji = "2.5 juta";
        }
        else if (gol == 2)
        {
            gaji = "3.5 juta";
        }
        else if (gol == 3)
        {
            gaji = "5 juta";
        }
        else if (gol == 4)

```

```

    {
        gaji = "7.5 juta";
    }
    return gaji;
}

//menampilkan seluruh elemen dalam Linked List (head) dengan format tabel
void traversal(list head)
{
    pointerToNode pHelp;
    int no = 1;
    std::string gaji;
    if (head == nullptr)
    {
        return;
    }
    else
    {
        pHelp = head;
        std::cout << "NO\tNIP\tNama\t\t\tGol\tGaji\n";
        std::cout <<
        "-----\n";
        do
        {
            gaji = gajiGol(pHelp->dataDosen.golongan);
            std::cout << no << "\t" << pHelp->dataDosen.NIP << "\t" <<
            pHelp->dataDosen.nama << "\t\t" << pHelp->dataDosen.golongan << "\t" << gaji
            << "\n";
            no++;
            pHelp = pHelp->nextNode;
        } while (pHelp != nullptr);
        std::cout <<
        "-----\n";
    }
}

int main()
{
    /*Inisialisasi head (pointer ke head linked list) menjadi NULL
    dan newNode (pointer untuk elemen baru)*/
    pointerToNode head = nullptr;
    list newNode;

    int pilihan;
    bool menu = 1;

    //Membuat loop while untuk menampilkan menu program
    while (menu)
    {
        std::cout << "\nMenu:" << std::endl;
        std::cout << "1. Create List" << std::endl;
        std::cout << "2. Create Elemen" << std::endl;
    }
}

```

```

        std::cout << "3. Insert First" << std::endl; //Menampilkan menu dengan
        pilihan (Create List, Create Elemen, Insert First, Delete First, Traversal,
        Keluar)
        std::cout << "4. Delete First" << std::endl;
        std::cout << "5. Traversal " << std::endl;
        std::cout << "0. Keluar" << std::endl;

        std::cout << "Masukkan pilihan: ";
        std::cin >> pilihan;
        //Membaca input pilihan pengguna (pilihan)
        switch (pilihan)
        {
        case 1:
            createHead(head); //membuat head linked list menjadi NULL
            (inisialisasi)
            std::cout << "List berhasil dibuat" << std::endl;
            break;
        case 2:
            createElement(newNode); //membuat elemen baru dosen, meminta input
            data, dan menyimpannya di newNode
            std::cout << "Elemen berhasil dibuat" << std::endl;
            break;
        case 3:
            insertFirst(head, newNode); //menyisipkan elemen baru (newNode) di
            depan linked list (head)
            std::cout << "Elemen berhasil disisipkan di depan" << std::endl;
            break;
        case 4:
            deleteFirst(head); //menghapus elemen pertama dari linked list (head)
            std::cout << "Elemen pertama berhasil dihapus" << std::endl;
            break;
        case 5:
            traversal(head); //menampilkan seluruh elemen dalam linked list
            (head) dengan format tabel
            break;
        default:
            menu = 0; //Pilihan lain (default): mengakhiri loop (menu menjadi 0)
        }
    }
}

```

b. Hasil Run

Menu:

1. Create List
2. Create Elemen
3. Insert First
4. Delete First
5. Traversal
0. Keluar

Masukkan pilihan: 1
List berhasil dibuat

Menu:

1. Create List
2. Create Elemen
3. Insert First
4. Delete First
5. Traversal
0. Keluar

Masukkan pilihan: 2

Masukkan NIP : 001
Masukkan Nama : Bagus
Masukkan Golongan : 2
Elemen berhasil dibuat

Menu:

1. Create List
2. Create Elemen
3. Insert First
4. Delete First
5. Traversal
0. Keluar

Masukkan pilihan: 3

Elemen berhasil disisipkan di depan

Menu:

1. Create List
2. Create Elemen
3. Insert First
4. Delete First
5. Traversal
0. Keluar

Masukkan pilihan: 2

Masukkan NIP : 002
Masukkan Nama : Diatama
Masukkan Golongan : 2
Elemen berhasil dibuat

Menu:

1. Create List
2. Create Elemen
3. Insert First
4. Delete First
5. Traversal
0. Keluar

Masukkan pilihan: 3

Elemen berhasil disisipkan di depan

Menu:

1. Create List
2. Create Elemen
3. Insert First
4. Delete First
5. Traversal
0. Keluar

Masukkan pilihan: 5

NO	NIP	Nama	Gol	Gaji
1	002	Diatama	2	3.5 juta
2	001	Bagas	2	3.5 juta

Menu:

1. Create List
2. Create Elemen
3. Insert First
4. Delete First
5. Traversal
0. Keluar

Masukkan pilihan: 4

Elemen pertama berhasil dihapus

Menu:

1. Create List
2. Create Elemen
3. Insert First
4. Delete First
5. Traversal
0. Keluar

Masukkan pilihan: 5

NO	NIP	Nama	Gol	Gaji
1	001	Bagas	2	3.5 juta

Menu:

1. Create List
2. Create Elemen
3. Insert First
4. Delete First
5. Traversal
0. Keluar

Masukkan pilihan: 0

