

STRUKTUR DATA

(Tugas5)



Nama : Prames Ray Lopian

NPM : 140810210059

Dikumpulkan tanggal :

27 Maret 2022

UNIVERSITAS PADJADJARAN

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM

Program Studi INFORMATIKA

2022

```

#include <iostream>
#include <string.h>
#include <iomanip>
using namespace std;

struct Pegawai
{
    char NIP[10];
    char Nama[30];
    char Alamat[30];
    string gol;
    float gaji;
};

struct Node
{
    Pegawai info;
    Node* next;
};

typedef Node *pointer;
typedef pointer List;

void createElement (pointer& pBaru);
void insertFirst (List& first, pointer pBaru);
void insertLast (List& first, pointer pBaru);
void insertAfter (List& first, pointer pBaru, pointer pCari);
void deleteFirst (List& first, pointer& pHapus);
void deleteLast (List& first, pointer& pHapus);
void deleteAfter (List& first, pointer& pHapus, pointer pCari);
void deleteByKey (List& first, pointer& pHapus, pointer pCari);
void traversal (List first);
void linearSearch (List first, pointer& pCari, char NIP[], int& found);
void gajiMaksimal (List first);
void gajiRata (List first);
string golPegawai (int gaji);
char menu();

int main()
{
    List first = NULL;
    pointer pBaru, pHapus, pCari;
    char keyNIP[10];
    int opsi;
    int found = 0;
    bool program = true;

```

```
while (program)
{
    int pil = menu();

    switch (pil)
    {
        case 1:
            createElement(pBaru);
            insertFirst(first, pBaru);
            traversal(first);
            break;

        case 2:
            createElement(pBaru);
            insertLast(first, pBaru);
            traversal(first);
            break;

        case 3:
            cout << "\nMasukkan NIP pencarian : "; cin.get(keyNIP,10);
            cin.ignore();
            linearSearch(first, pCari, keyNIP, found);

            if (found)
            {
                cout << "\nData yang dicari telah ditemukan!" << endl;
                createElement(pBaru);
                insertAfter(first, pBaru, pCari);
                traversal(first);
            }
            else
            {
                cout << "\nData yang dicari tidak ditemukan." << endl;
            }
            break;

        case 4:
            deleteFirst(first, pHapus);
            cout << endl;
            traversal(first);
            break;

        case 5:
            deleteLast(first, pHapus);
            cout << endl;
            traversal(first);
            break;
    }
}
```

```

case 6:
    cout << "\nMasukkan NIP Pencarian\t: "; cin.get(keyNIP,10);
    cin.ignore();
    linearSearch(first, pCari, keyNIP, found);

    if (found)
    {
        deleteByKey(first, pHapus, pCari);
        cout << endl;
        traversal(first);
    }
    else
    {
        cout << "\nData Tidak Ditemukan!" << endl;
    }

    break;

case 7:
    cout << "\nMasukkan NIP Pencarian\t: "; cin.get(keyNIP,10);
    cin.ignore();
    linearSearch(first, pCari, keyNIP, found);

    if (found)
    {
        deleteAfter(first, pHapus, pCari);
        cout << endl;
        traversal(first);
    }
    else
    {
        cout << "\nData Tidak Ditemukan!" << endl;
    }
    break;

case 8:
    traversal(first);
    break;

case 9:
    gajiMaksimal(first);
    break;

case 10:
    gajiRata(first);
    break;

```

```

        default:
            cout << "\nPilihan Tidak Tersedia." << endl;
            break;
    }

    cout << "\nIngin terus menggunakan program?" << endl
         << "1.YA" << endl
         << "2.TIDAK" << endl
         << "Pilihan\t: "; cin >> opsi; cin.ignore();

    if (opsi == 1)
    {
        program = true;
    }
    else if (opsi == 2)
    {
        program = false;
        cout << "\nTerima kasih!" << endl;
    }
    else
    {
        program = false;
        cout << "\nPilihan Tidak Tersedia" << endl;
    }
}

}

void createElement (pointer& pBaru)
{
    pBaru = new Node;

    cout << "\nData pegawai yang ingin ditambahkan:" << endl;
    cout << "NIP    : "; cin.get(pBaru -> info.NIP,10); cin.ignore();
    cout << "Nama    : "; cin.get(pBaru -> info>Nama,30); cin.ignore();
    cout << "Alamat  : "; cin.get(pBaru -> info.Alat,30); cin.ignore();
    cout << "Gaji    : "; cin >> pBaru -> info.gaji; cin.ignore();

    pBaru -> info.gol = golPegawai(pBaru -> info.gaji);
    pBaru -> next = NULL;
}

void insertFirst (List& first, pointer pBaru)
{
    if (first == NULL)
    {
        first = pBaru;
    }
}

```

```

        else
        {
            pBaru -> next = first;
            first = pBaru;
        }
    }

void insertLast (List& first, pointer pBaru)
{
    if (first == NULL)
    {
        first = pBaru;
    }
    else
    {
        pointer last = first;

        while (last -> next != NULL)
        {
            last = last -> next;
        }

        last -> next = pBaru;
    }
}

void insertAfter (List& first, pointer pBaru, pointer pCari)
{
    if (pCari -> next == NULL)
    {
        pCari -> next = pBaru;
    }
    else
    {
        pBaru -> next = pCari -> next;
        pCari -> next = pBaru;
    }
}

void deleteFirst (List& first, pointer& pHapus)
{
    if (first == NULL)
    {
        pHapus = NULL;
    }
    else if (first -> next == NULL)
    {
        pHapus = first;
    }
}

```

```

        first = NULL;
    }
    else
    {
        pHapus = first;
        first = first -> next;
        pHapus -> next = NULL;
    }
}

void deleteLast (List& first, pointer& pHapus)
{
    if (first == NULL)
    {
        pHapus = NULL;
    }
    else if (first -> next == NULL)
    {
        pHapus = first;
        first = NULL;
    }
    else
    {
        pointer last = first;
        pointer preLast = NULL;

        while (last -> next != NULL)
        {
            preLast = last;
            last = last -> next;
        }

        pHapus = last;
        preLast -> next = NULL;
    }
}

void deleteAfter (List& first, pointer& pHapus, pointer pCari)
{
    if (pCari -> next == NULL)
    {
        pHapus = NULL;
        cout << "Tidak ada yang dihapus" << endl;
    }
    else
    {
        pHapus = pCari -> next;
        pCari -> next = pHapus -> next;
    }
}

```

```

        pHapus -> next = NULL;
    }
}

void deleteByKey (List& first, pointer& pHapus, pointer pCari)
{
    if (pCari == first)
    {
        deleteFirst(first, pHapus);
    }
    else if (pCari -> next == NULL)
    {
        deleteLast(first, pHapus);
    }
    else
    {
        deleteAfter(first, pHapus, pCari);
    }
}

void traversal (List first)
{
    if (first == NULL)
    {
        cout << "\nList kosong!" << endl;
    }
    else
    {
        pointer pBantu = first;
        cout << endl;
        cout << setw(10) << "NIP" << setw(30) << "NAMA" << setw(30) <<
"ALAMAT" << setw(10) << "GOL" << setw(15) << "GAJI" << endl;

        do
        {
            cout << setw(10) << pBantu->info.NIP << setw(30) << pBantu-
>info>Nama << setw(30) << pBantu->info.Alamat;
            cout << setw(10) << pBantu->info.gol << setw(15) << fixed <<
setprecision(0) << pBantu->info.gaji << endl;

            pBantu = pBantu -> next;
        }
        while (pBantu != NULL);
    }
}

void linearSearch (List first, pointer& pCari, char NIP[], int& found)
{

```



```

found = 0;
pCari = first;

while (found == 0 && pCari != NULL)
{
    if (strcmp(pCari->info.NIP, NIP) == 0)
    {
        found = 1;
    }
    else
    {
        pCari = pCari -> next;
    }
}

}

void gajiMaksimal (List first)
{
    pointer pBantu;
    float maksimal = 0;

    if (first == NULL)
    {
        cout << "\nList kosong!" << endl;
    }
    else
    {
        pBantu = first;

        do
        {
            if (pBantu->info.gaji > maksimal)
            {
                maksimal = pBantu->info.gaji;
            }

            pBantu = pBantu -> next;
        }
        while (pBantu != NULL);

        cout << "\nGaji Maksimum\t: Rp" << maksimal << endl;
    }
}

void gajiRata (List first)
{
    pointer pBantu;
    float hasil, rata = 0;

```

```

int i = 0;

if (first == NULL)
{
    cout << "\nList kosong!" << endl;
}
else
{
    pBantu = first;

    do
    {
        rata += pBantu->info.gaji;
        i++;
        pBantu = pBantu -> next;
        hasil = rata/i;
    }
    while (pBantu != NULL);

    cout << "\nRata-Rata Gaji\t: Rp" << hasil << endl;
}
}

string golPegawai (int gaji)
{
    string gol;

    if (gaji <= 1000000)
    {
        gol = "1A";
    }
    else if (gaji > 1000000 && gaji <= 2000000)
    {
        gol = "1B";
    }
    else if (gaji > 2000000 && gaji <= 3000000)
    {
        gol = "2A";
    }
    else if (gaji > 3000000 && gaji <= 4000000)
    {
        gol = "2B";
    }
    else if (gaji > 4000000 && gaji <= 5000000)
    {
        gol = "3A";
    }
    else if (gaji > 5000000 && gaji <= 6000000)

```

```

{
    gol = "3B";
}
else if (gaji > 6000000 && gaji <= 7000000)
{
    gol = "4A";
}
else if (gaji > 7000000)
{
    gol = "4B";
}

return gol;
}

char menu()
{
    int opsi;
    cout << "===== " << endl
        << "          MENU PROGRAM PEGAWAI          " << endl
        << "===== " << endl
        << "1. Input Data Pertama Pegawai          " << endl
        << "2. Input Data Terakhir Pegawai         " << endl
        << "3. Input Data Pegawai (After Key)       " << endl
        << "4. Hapus Data Pertama Pegawai           " << endl
        << "5. Hapus Data Terakhir Pegawai          " << endl
        << "6. Hapus Data Pegawai (By Key)          " << endl
        << "7. Hapus Data Pegawai (After Key)       " << endl
        << "8. Tampilkan Seluruh List Data Pegawai" << endl
        << "9. Tampilkan Gaji Maksimum Pegawai      " << endl
        << "10.Tampilkan Rata-Rata Gaji Pegawai     " << endl << endl
        << "Pilihan\t: "; cin >> opsi; cin.ignore();

    return opsi;
}

```

Output Program:

1. Kondisi Traversal Ketika tidak ada list:

```

=====
MENU PROGRAM PEGAWAI
=====
1. Input Data Pertama Pegawai
2. Input Data Terakhir Pegawai
3. Input Data Pegawai (After Key)
4. Hapus Data Pertama Pegawai
5. Hapus Data Terakhir Pegawai
6. Hapus Data Pegawai (By Key)
7. Hapus Data Pegawai (After Key)
8. Tampilkan Seluruh List Data Pegawai
9. Tampilkan Gaji Maksimum Pegawai
10. Tampilkan Rata-Rata Gaji Pegawai

Pilihan : 8

List kosong!

```

2. Kondisi meng-input data di posisi awal:

```

=====
MENU PROGRAM PEGAWAI
=====
1. Input Data Pertama Pegawai
2. Input Data Terakhir Pegawai
3. Input Data Pegawai (After Key)
4. Hapus Data Pertama Pegawai
5. Hapus Data Terakhir Pegawai
6. Hapus Data Pegawai (By Key)
7. Hapus Data Pegawai (After Key)
8. Tampilkan Seluruh List Data Pegawai
9. Tampilkan Gaji Maksimum Pegawai
10. Tampilkan Rata-Rata Gaji Pegawai

Pilihan : 1

Data pegawai yang ingin ditambahkan:
NIP      : 00001
Nama     : Prames
Alamat   : Kebon Baru
Gaji     : 4000000

      NIP              NAMA              ALAMAT              GOL              GAJI
      00001              Prames              Kebon Baru              2B              4000000

```

3. Kondisi meng-input data di posisi akhir:

```
=====
MENU PROGRAM PEGAWAI
=====
1. Input Data Pertama Pegawai
2. Input Data Terakhir Pegawai
3. Input Data Pegawai (After Key)
4. Hapus Data Pertama Pegawai
5. Hapus Data Terakhir Pegawai
6. Hapus Data Pegawai (By Key)
7. Hapus Data Pegawai (After Key)
8. Tampilkan Seluruh List Data Pegawai
9. Tampilkan Gaji Maksimum Pegawai
10.Tampilkan Rata-Rata Gaji Pegawai

Pilihan : 2

Data pegawai yang ingin ditambahkan:
NIP    : 00002
Nama   : Ray
Alamat : Kebon Bawang
Gaji   : 3000000

      NIP                NAMA                ALAMAT                GOL                GAJI
      00001              Prames              Kebon Baru            2B                4000000
      00002              Ray                Kebon Bawang          2A                3000000
```

4. Kondisi meng-input data di posisi setelah yang data yang ditentukan:

```
=====
MENU PROGRAM PEGAWAI
=====
1. Input Data Pertama Pegawai
2. Input Data Terakhir Pegawai
3. Input Data Pegawai (After Key)
4. Hapus Data Pertama Pegawai
5. Hapus Data Terakhir Pegawai
6. Hapus Data Pegawai (By Key)
7. Hapus Data Pegawai (After Key)
8. Tampilkan Seluruh List Data Pegawai
9. Tampilkan Gaji Maksimum Pegawai
10.Tampilkan Rata-Rata Gaji Pegawai

Pilihan : 3

Masukkan NIP pencarian : 00002

Data yang dicari telah ditemukan!

Data pegawai yang ingin ditambahkan:
NIP    : 00003
Nama   : Lopian
Alamat : Kebon Kacang
Gaji   : 3500000

      NIP                NAMA                ALAMAT                GOL                GAJI
      00001              Prames              Kebon Baru            2B                4000000
      00002              Ray                Kebon Bawang          2A                3000000
      00003              Lopian              Kebon Kacang          2B                3500000
```

5. Kondisi menghapus data di posisi awal:

```
=====
MENU PROGRAM PEGAWAI
=====
1. Input Data Pertama Pegawai
2. Input Data Terakhir Pegawai
3. Input Data Pegawai (After Key)
4. Hapus Data Pertama Pegawai
5. Hapus Data Terakhir Pegawai
6. Hapus Data Pegawai (By Key)
7. Hapus Data Pegawai (After Key)
8. Tampilkan Seluruh List Data Pegawai
9. Tampilkan Gaji Maksimum Pegawai
10.Tampilkan Rata-Rata Gaji Pegawai

Pilihan : 4

      NIP                NAMA                ALAMAT                GOL                GAJI
      00002                Ray                Kebon Bawang                2A                3000000
      00003                Lopian                Kebon Kacang                2B                3500000
```

6. Kondisi menghapus data di posisi akhir:

```
=====
MENU PROGRAM PEGAWAI
=====
1. Input Data Pertama Pegawai
2. Input Data Terakhir Pegawai
3. Input Data Pegawai (After Key)
4. Hapus Data Pertama Pegawai
5. Hapus Data Terakhir Pegawai
6. Hapus Data Pegawai (By Key)
7. Hapus Data Pegawai (After Key)
8. Tampilkan Seluruh List Data Pegawai
9. Tampilkan Gaji Maksimum Pegawai
10.Tampilkan Rata-Rata Gaji Pegawai

Pilihan : 5

      NIP                NAMA                ALAMAT                GOL                GAJI
      00001                Prames                Kebon Baru                2B                4000000
      00002                Ray                Kebon Bawang                2A                3000000
```

7. Kondisi menghapus data di posisi setelah yang ditentukan:

```
=====
MENU PROGRAM PEGAWAI
=====
1. Input Data Pertama Pegawai
2. Input Data Terakhir Pegawai
3. Input Data Pegawai (After Key)
4. Hapus Data Pertama Pegawai
5. Hapus Data Terakhir Pegawai
6. Hapus Data Pegawai (By Key)
7. Hapus Data Pegawai (After Key)
8. Tampilkan Seluruh List Data Pegawai
9. Tampilkan Gaji Maksimum Pegawai
10.Tampilkan Rata-Rata Gaji Pegawai

Pilihan : 7

Masukkan NIP Pencarian : 00001

      NIP                NAMA                ALAMAT                GOL                GAJI
      00001                Prames                Kebon Baru                2B                4000000
      00003                Lopian                Kebon Bawang                2A                3000000
```

8. Kondisi menghapus data yang dipilih:

```
=====
MENU PROGRAM PEGAWAI
=====
1. Input Data Pertama Pegawai
2. Input Data Terakhir Pegawai
3. Input Data Pegawai (After Key)
4. Hapus Data Pertama Pegawai
5. Hapus Data Terakhir Pegawai
6. Hapus Data Pegawai (By Key)
7. Hapus Data Pegawai (After Key)
8. Tampilkan Seluruh List Data Pegawai
9. Tampilkan Gaji Maksimum Pegawai
10.Tampilkan Rata-Rata Gaji Pegawai

Pilihan : 6

Masukkan NIP Pencarian : 00003
```

| NIP | NAMA | ALAMAT | GOL | GAJI |
|-------|--------|--------------|-----|---------|
| 00001 | Prames | Kebon Baru | 2B | 4000000 |
| 00002 | Ray | Kebon Kacang | 2B | 3500000 |

9. Kondisi menampilkan rata-rata gaji pegawai:

```
=====
MENU PROGRAM PEGAWAI
=====
1. Input Data Pertama Pegawai
2. Input Data Terakhir Pegawai
3. Input Data Pegawai (After Key)
4. Hapus Data Pertama Pegawai
5. Hapus Data Terakhir Pegawai
6. Hapus Data Pegawai (By Key)
7. Hapus Data Pegawai (After Key)
8. Tampilkan Seluruh List Data Pegawai
9. Tampilkan Gaji Maksimum Pegawai
10.Tampilkan Rata-Rata Gaji Pegawai

Pilihan : 10

Rata-Rata Gaji : Rp3500000
```

10. Kondisi menampilkan gaji terbesar pegawai:

```
=====
                        MENU PROGRAM PEGAWAI
=====
1. Input Data Pertama Pegawai
2. Input Data Terakhir Pegawai
3. Input Data Pegawai (After Key)
4. Hapus Data Pertama Pegawai
5. Hapus Data Terakhir Pegawai
6. Hapus Data Pegawai (By Key)
7. Hapus Data Pegawai (After Key)
8. Tampilkan Seluruh List Data Pegawai
9. Tampilkan Gaji Maksimum Pegawai
10.Tampilkan Rata-Rata Gaji Pegawai

Pilihan : 9

Gaji Maksimum      : Rp4000000
```

11. Kondisi menampilkan seluruh data terbaru pegawai:

```
=====
                        MENU PROGRAM PEGAWAI
=====
1. Input Data Pertama Pegawai
2. Input Data Terakhir Pegawai
3. Input Data Pegawai (After Key)
4. Hapus Data Pertama Pegawai
5. Hapus Data Terakhir Pegawai
6. Hapus Data Pegawai (By Key)
7. Hapus Data Pegawai (After Key)
8. Tampilkan Seluruh List Data Pegawai
9. Tampilkan Gaji Maksimum Pegawai
10.Tampilkan Rata-Rata Gaji Pegawai

Pilihan : 8
```

| NIP | NAMA | ALAMAT | GOL | GAJI |
|-------|--------|--------------|-----|---------|
| 00001 | Prames | Kebon Baru | 2B | 4000000 |
| 00002 | Ray | Kebon Kacang | 2B | 3500000 |
| 00003 | Lapian | Kebon Bawang | 2A | 3000000 |