**STRUKTUR DATA**

(UTS)

****

Nama : Prames Ray Lapian

NPM : 140810210059

Dikumpulkan tanggal :

3 April 2022

UNIVERSITAS PADJADJARAN

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM

Program Studi INFORMATIKA

2022

1. Soal 1:

/\*  Nama Program    : 210059\_UTSStrukDat\_Soal1

    Nama            : Prames Ray Lapian

    NPM             : 140810210059

    Tanggal Buat    : 10 April 2022

    Deskripsi       : Soal 1 UTS

    Lokasi File     : C:\Users\prame\Documents\PRAMES\PERKULIAHAN\SEMESTER 2\Struktur Data\TUGAS

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <iostream>

using namespace std;

typedef int Larik[15];

void inputBanyakData(int& n)

{

    cout << "Input banyak data : "; cin >> n;

}

void inputData(Larik& data, int n)

{

    for (int i = 0; i < n; i++)

    {

        cout << "\nMasukan data mahasiswa ke- " << (i+1) << endl

             << "Nilai : "; cin >> data[i];

    }

}

void cetakData(Larik data, int n)

{

    cout << "=========" << endl

         << "  DATA   " << endl

         << "=========" << endl;

    for (int i = 0; i < n; i++)

    {

        cout << data[i] << endl;

    }

}

int deleteLarik(Larik& data, int& n, int key, int status)

{

    status = 0;

    cout << "Data yang ingin dihapus : "; cin >> key; cin.ignore();

    for (int i = 0; i < n; i++)

    {

        if (data[i] == key)

        {

            status = 1;

            data[i] = data[i + 1];

            n -= 1;

            cout << "Data berhasil dihapus" << endl;

            break;

        }

        else if (i == n)

        {

            cout << "Data tidak ditemukan" << endl;

            break;

        }

    }

    return 0;

}

int main()

{

    Larik data;

    int n, key, help;

    inputBanyakData(n);

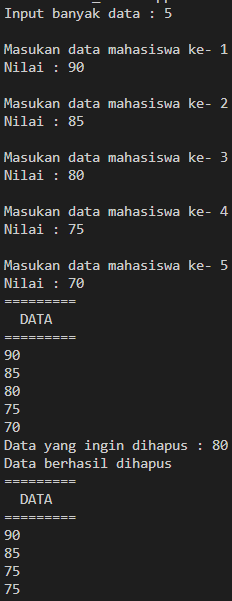
    inputData(data, n);

    cetakData(data, n);

    deleteLarik(data, n, key, help);

    cetakData(data, n);

}



1. Soal 2:

/\*  Nama Program    : 210059\_UTSStrukDat\_Soal2

    Nama            : Prames Ray Lapian

    NPM             : 140810210059

    Tanggal Buat    : 10 April 2022

    Deskripsi       : Soal 2 UTS

    Lokasi File     : C:\Users\prame\Documents\PRAMES\PERKULIAHAN\SEMESTER 2\Struktur Data\TUGAS

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <iostream>

#include <iomanip>

using namespace std;

struct Mahasiswa

{

    string nama;

    string npm;

    float nilai;

    Mahasiswa \*next;

};

struct Node

{

    Mahasiswa info;

    Node \*next;

};

typedef Node\* pointer;

typedef pointer List;

void createElement(pointer &pBaru)

{

    pBaru = new Node;

    cout << "Nama   : "; cin >> pBaru -> info.nama; cin.ignore();

    cout << "NPM    : "; cin >> pBaru -> info.npm; cin.ignore();

    cout << "Nilai  : "; cin >> pBaru -> info.nilai; cin.ignore();

    pBaru -> next = NULL;

}

void insertFirst(List &first, pointer pBaru)

{

    cout << endl;

    if (first == NULL)

    {

        first = pBaru;

    }

    else

    {

        pBaru -> next = first;

        first = pBaru;

    }

}

void rataRata(float &rata, List &first)

{

    float jumlah = 0;

    int loop = 0;

    if (first == NULL)

    {

        cout << "List kosong!\n";

    }

    else

    {

        pointer pBantu = first;

        do

        {

            jumlah += pBantu -> info.nilai;

            pBantu = pBantu -> next;

            loop += 1;

        }

        while (pBantu != NULL);

        rata = jumlah / loop;

        cout << "Rata - rata nilai : " << rata << endl;

    }

}

char nilaiMutu(float nilai)

{

    char nilaiMutu;

    if (nilai >= 80 && nilai <= 100)

    {

        nilaiMutu = 'A';

    }

    else if (nilai >= 68 && nilai <= 80)

    {

        nilaiMutu = 'B';

    }

    else if (nilai >= 55 && nilai <= 68)

    {

        nilaiMutu = 'C';

    }

    else if (nilai >= 45 && nilai <= 55)

    {

        nilaiMutu = 'D';

    }

    else if (nilai >= 0 && nilai <= 45)

    {

        nilaiMutu = 'E';

    }

    return nilaiMutu;

}

string kelulusan(char nilaiMutu)

{

    string status;

    if (nilaiMutu == 'A' || nilaiMutu == 'B' || nilaiMutu == 'C')

    {

        status = "lulus";

    }

    else

    {

        status = "tidak lulus";

    }

    return status;

}

void traversal(List first, int number)

{

    if (first == NULL)

    {

        cout << "\nList kosong!" << endl;

    }

    else

    {

        pointer pBantu = first;

        cout << endl;

        cout << "-------------------------------------------------------------------------" << endl;

        cout << setw(5) << "NO" << setw(10) << "NPM" << setw(30) << "NAMA"  << setw(10) << "NILAI" << setw(5) << "HM" << setw(10) << "STATUS" << endl;

        cout << "-------------------------------------------------------------------------" << endl;

        do

        {

            cout << setw(5) << number << setw(10) << pBantu -> info.npm << setw(30) << pBantu -> info.nama  << setw(10) << pBantu -> info.nilai << setw(5) << nilaiMutu(pBantu -> info.nilai) << setw(10)  << kelulusan(nilaiMutu(pBantu -> info.nilai)) << endl;

            number += 1;

            pBantu = pBantu -> next;

        }

        while (pBantu != NULL);

        cout << "-------------------------------------------------------------------------" << endl;

    }

}

int main()

{

    List first = NULL;

    pointer pBaru;

    float rata;

    int number = 1;

    bool program = true;

    int opsi;

    while (program)

    {

        createElement(pBaru);

        insertFirst(first, pBaru);

        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;

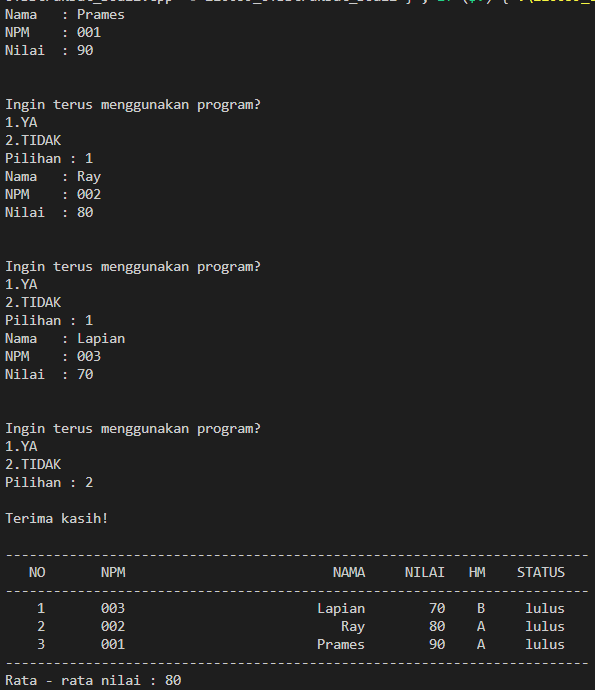
        }

    }

    traversal(first, number);

    rataRata(rata, first);

}



1. Soal 3:

/\*  Nama Program    : 210059\_UTSStrukDat\_Soal3

    Nama            : Prames Ray Lapian

    NPM             : 140810210059

    Tanggal Buat    : 10 April 2022

    Deskripsi       : Soal 3 UTS

    Lokasi File     : C:\Users\prame\Documents\PRAMES\PERKULIAHAN\SEMESTER 2\Struktur Data\TUGAS

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include<iostream>

#include <iomanip>

using namespace std;

struct pegawai

{

    int NIP;

    char nama[30];

    int gol;

};

struct node

{

    node \*next;

    node \*prev;

    pegawai info;

};

typedef node\* pointer;

typedef pointer List;

void createElement(pointer &pBaru)

{

    pBaru = new node;

    cout << "Nama     : "; cin.get(pBaru -> info.nama, 30); cin.ignore();

    cout << "NIP      : "; cin >> pBaru -> info.NIP;

    cout << "Golongan : "; cin >> pBaru -> info.gol;

    pBaru -> next = NULL;

    pBaru -> prev = NULL;

}

void search(List& first, int& key)

{

    if (first != NULL)

    {

        cout << "Masukkan NIP yang dicari : "; cin >> key; cin.ignore();

    }

    else

    {

        key = 0;

    }

}

void linearSearch(List &first, pointer &pCari, int key, pointer& pBantu)

{

    pCari = first;

    while (pCari != NULL)

    {

        if (pCari -> info.NIP == key)

        {

            break;

        }

        pBantu = pCari;

        pCari = pCari -> next;

    }

}

void insertBefore(List& first, pointer pCari, pointer pBantu, pointer pBaru)

{

    if (first == NULL)

    {

        first = pBaru;

    }

    else if (pCari == first)

    {

        pBaru -> next = first;

        first -> prev = pBaru;

        pBaru -> prev = NULL;

        first = pBaru;

    }

    else

    {

        pBantu -> next = pBaru;

        pBaru -> prev = pBantu;

        pCari -> prev = pBaru;

        pBaru -> next = pCari;

    }

}

void golGaji(int k)

{

    int gaji;

    if(k == 1)

    {

        gaji = 1000000;

    }

    else if(k == 2)

    {

        gaji = 2000000;

    }

    else if(k == 3)

    {

        gaji = 3000000;

    }

    cout << setw(15) << gaji;

}

void traversal(pointer first)

{

    if (first == NULL)

    {

        cout << "list kosong" << endl;

    }

    else

    {

        pointer pBantu = first;

        while (pBantu->next != NULL)

        {

            pBantu = pBantu->next;

        }

        cout << setw(5) << "NO" << setw(10) << "NIP" << setw(20) << "NAMA" << setw(15) << "GOLONGAN" << setw(15) << "GAJI" << endl;

        int number = 1;

        while (pBantu != NULL)

        {

            cout << setw(5) << number << setw(10) << pBantu->info.NIP << setw(20) << pBantu->info.nama << setw(15) << pBantu->info.gol;

            golGaji(pBantu->info.gol);

            cout << endl;

            pBantu = pBantu->prev;

            number++;

        }

    }

}

void edit(pointer pCari)

{

    cout << "nama baru : "; cin.get(pCari->info.nama, 30); cin.ignore();

    cout << "golongan baru : "; cin >> pCari->info.gol;

}

void menu(int &opsi)

{

    cout << endl;

    cout << "========================" << endl

         << "      LIST PEGAWAI      " << endl

         << "========================" << endl

         << "1. Input data           " << endl

         << "2. Cetak data           " << endl

         << "3. Edit data            " << endl

         << "4. keluar               " << endl

         << "Pilihan : "; cin >> opsi; cin.ignore();

    cout << endl;

}

int main()

{

    List list = NULL;

    pointer pBaru, pHapus, pCari, pBantu;

    int nilai, opsi;

    program:

    menu(opsi);

    switch (opsi)

    {

        case 1 :

            search(list, nilai);

            linearSearch(list, pCari, nilai, pBantu);

            createElement(pBaru);

            insertBefore(list, pCari, pBantu, pBaru);

            goto program;

        break;

        case 2 :

            traversal(list);

            goto program;

        break;

        case 3 :

            search(list, nilai);

            linearSearch(list, pCari, nilai, pBantu);

            edit(pCari);

            goto program;

        break;

        default :

        break;

    }

}

