

LAPORAN PRAKTIKUM STRUKTUR DATA

LAPORAN AWAL PRAKTIKUM

Pertemuan ke-03

Structure



Disusun Oleh:

Nama Lengkap : Nova Ardiansyah

NIM : 211011401309

Kelas : 04-TPLE008

TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PAMULANG

Jl. Raya Puspitek No. 11 Buaran, Serpong Telp. (021) 7412566, Fax. (021) 7412566
Tangerang Selatan - Banten

A. RANGKUMAN MATERI

C++ adalah bahasa pemrograman yang kuat dan efisien, dan salah satu fitur utama yang membuatnya begitu berguna adalah struktur. Struktur adalah jenis data yang memungkinkan programmer untuk mengelompokkan beberapa tipe data bersama-sama dalam satu objek. Dalam bahasa C++, struktur terdiri dari beberapa variabel dengan jenis yang berbeda, yang dapat diakses dan dimanipulasi secara bersamaan. Sebagai contoh, struktur bisa digunakan untuk merepresentasikan sebuah buku, dimana struktur bisa memiliki variabel-variabel seperti judul, penulis, dan nomor ISBN.

Dalam C++, struktur didefinisikan menggunakan kata kunci 'struct'. Setiap variabel dalam struktur disebut anggota struktur atau 'member'. Anggota struktur dapat diakses menggunakan operator titik (.) setelah nama variabel struktur. Sebagai contoh, jika Anda memiliki sebuah struktur yang disebut 'Mahasiswa', dan Anda ingin mengakses anggota 'Nama' dari objek mahasiswa tersebut, maka Anda dapat menggunakan sintaks 'mahasiswa.Nama'.

Satu fitur menarik dari struktur dalam C++ adalah kemampuannya untuk disarangkan. Ini berarti bahwa struktur dapat memiliki struktur lain di dalamnya, yang memungkinkan programmer untuk membuat struktur yang kompleks dan terorganisir dengan baik. Misalnya, jika Anda memiliki struktur 'Alamat' dengan anggota seperti nomor jalan, kota, dan kode pos, Anda dapat menyematkan struktur ini ke dalam struktur 'Mahasiswa' untuk merepresentasikan alamat mahasiswa. Ini membuat program lebih mudah dibaca dan dipahami, dan memudahkan pengelolaan data secara umum.

B. TUGAS PENDAHULUAN

1. Apa yang dimaksud dengan Stucture!

Jawab :

Struktur dalam pemrograman adalah jenis data yang memungkinkan pengelompokan beberapa variabel dengan tipe data yang berbeda menjadi satu objek. Struktur memungkinkan programmer untuk mengelola data dengan lebih terstruktur dan terorganisir, sehingga memudahkan pembacaan dan pemeliharaan kode.

2. Jelaskan kelebihan sebuah program menggunakan Structure!

Jawab :

Kelebihan penggunaan struktur dalam sebuah program adalah memungkinkan programmer untuk mengelompokkan data terkait dalam satu objek, sehingga program lebih mudah dipahami dan dikelola. Selain itu, struktur juga dapat disarangkan, sehingga memungkinkan pembuatan struktur yang kompleks dan terorganisir dengan baik. Struktur juga memudahkan programmer untuk memanipulasi data secara bersamaan, serta menghindari kesalahan penulisan variabel yang serupa.

3. Buatlah contoh program sederhana menggunakan Structure!

Jawab :

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

struct Mahasiswa {
    string nama;
    int usia;
    float ipk;
};

int main() {
    cout << "Nama \t: Nova Ardiansyah\n";
    cout << "NIM \t: 211011401309\n";
    cout << "=====\n\n";

    Mahasiswa mhs1;
    mhs1.nama = "Nova Ardiansyah";
    mhs1.usia = 22;
    mhs1.ipk = 3.85;

    cout << "Nama\t: " << mhs1.nama << endl;
    cout << "Usia\t: " << mhs1.usia << endl;
    cout << "IPK\t: " << mhs1.ipk << endl;

    return 0;
}
```

```

Nama      : Nova Ardiansyah
NIM       : 211011401309
=====
Nama      : Nova Ardiansyah
Usia      : 22
IPK       : 3.85

```

4. Buatlah contoh program sederhana kombinasi Array dan Structure!

Jawab :

```

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

struct Mahasiswa {
    string nama;
    int nilai[3];
};

int main() {
    cout << "Nama \t: Nova Ardiansyah\n";
    cout << "NIM \t: 211011401309\n";
    cout << "=====\n\n";

    Mahasiswa mhs[2];

    for(int i = 0; i < 2; i++) {
        cout << "Masukkan nama mahasiswa " << i+1 << ": ";
        cin >> mhs[i].nama;

        for(int j = 0; j < 3; j++) {
            cout << "Masukkan nilai ke-" << j+1 << ": ";
            cin >> mhs[i].nilai[j];
        }
    }

    cout << endl;
    cout << "Data mahasiswa:" << endl;

    for(int i = 0; i < 2; i++) {
        cout << "Mahasiswa " << i+1 << ": " << mhs[i].nama << endl;
        cout << "Nilai: ";
        for(int j = 0; j < 3; j++) {
            cout << mhs[i].nilai[j] << " ";
        }
        cout << endl;
    }

    return 0;
}

```

```
Nama      : Nova Ardiansyah
NIM       : 211011401309
=====

Masukkan nama mahasiswa 1: Nova
Masukkan nilai ke-1: 89
Masukkan nilai ke-2: 75
Masukkan nilai ke-3: 80
Masukkan nama mahasiswa 2: Alex
Masukkan nilai ke-1: 80
Masukkan nilai ke-2: 75
Masukkan nilai ke-3: 89

Data mahasiswa:
Mahasiswa 1: Nova
Nilai: 89 75 80
Mahasiswa 2: Alex
Nilai: 80 75 89
```

C. TUGAS PRAKTIKUM

- Lat3_1

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

using namespace std;

struct Mahasiswa
{
    char NIM[15];
    char Nama[25];
    char Alamat[40];
    short Umur;
};

int main()
{
    cout << "Nama \t: Nova Ardiansyah\n";
    cout << "NIM \t: 211011401309\n";
    cout << "=====\n\n";

    Mahasiswa Mhs;

    cout << "Masukkan NIM \t: ";
    cin.getline(Mhs.NIM, 15);

    cout << "Masukkan Nama \t: ";
    cin.getline(Mhs.Nama, 25);

    cout << "Masukkan Alamat : ";
    cin.getline(Mhs.Alat, 40);

    cout << "Masukkan Umur \t: ";
    cin >> Mhs.Umur;

    cout << "=====\n\n";

    cout << "Berikut adalah data yang telah dimasukkan: \n";
    cout << "NIM \t: " << Mhs.NIM << endl;
    cout << "Nama \t: " << Mhs.Nama << endl;
    cout << "Alamat \t: " << Mhs.Alat << endl;
    cout << "Umur \t: " << Mhs.Umur << endl;

    return 0;
}
```

```
Output

/tmp/8kFTszGjmS.o
Nama      : Nova Ardiansyah
NIM       : 211011401309
=====

Masukkan NIM      : 211011401309
Masukkan Nama     : Nova Ardiansyah
Masukkan Alamat  : Jl. Jalur Sutera Kav 29A
Masukkan Umur     : 22
=====

Berikut adalah data yang telah dimasukkan:
NIM      : 211011401309
Nama     : Nova Ardiansyah
Alamat  : Jl. Jalur Sutera Kav 29A
Umur    : 22
```

- Lat3_2

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

using namespace std;

struct tinggal
{
    char Jalan[40];
    char Kota[40];
    char Pos[5];
};

struct TglLahir
{
    char Tgl[4];
    char Bln[4];
    char Thn[5];
};

struct Mahasiswa
{
    char NIM[15];
    char Nama[25];
};

int main()
{
    cout << "Nama \t: Nova Ardiansyah\n";
    cout << "NIM \t: 211011401309\n";
    cout << "=====\n\n";

    Mahasiswa Mhs;

    cout << "Masukkan NIM \t\t\t: ";
    cin.getline(Mhs.NIM, 15);
```

```

cout << "Masukkan Nama \t\t\t: ";
cin.getline(Mhs>Nama, 25);

TglLahir Mhs_TglLahir;

cout << "Masukkan Tanggal Lahir \t: ";
cin.getline(Mhs_TglLahir.Tgl, 4);

cout << "Masukkan Bulan Lahir \t: ";
cin.getline(Mhs_TglLahir.Bln, 4);

cout << "Masukkan Tahun Lahir \t: ";
cin.getline(Mhs_TglLahir.Thn, 5);

tinggal Mhs_tinggal;

cout << "Masukkan Jalan \t\t\t: ";
cin.getline(Mhs_tinggal.Jalan, 40);

cout << "Masukkan Kota \t\t\t: ";
cin.getline(Mhs_tinggal.Kota, 40);

cout << "Masukkan Kode Pos \t\t: ";
cin.getline(Mhs_tinggal.Pos, 5);

cout << "=====\n\n";

cout << "Berikut adalah data yang telah dimasukkan: \n";
cout << "NIM \t\t\t: " << Mhs.NIM << endl;
cout << "Nama \t\t\t: " << Mhs>Nama << endl;
cout << "Tanggal Lahir \t: " << Mhs_TglLahir.Tgl << "/" << Mhs_TglLahir.Bln << "/" <<
Mhs_TglLahir.Thn << endl;
cout << "Alamat \t\t\t: " << Mhs_tinggal.Jalan << ", " << Mhs_tinggal.Kota << ", " <<
Mhs_tinggal.Pos << "." << endl;

return 0;
}

```

```

/tmp/8kFTszGjmS.o
Nama      : Nova Ardiansyah
NIM       : 211011401309
=====

Masukkan NIM      : 211011401309
Masukkan Nama     : Nova Ardiansyah
Masukkan Tanggal Lahir : 11
Masukkan Bulan Lahir   : 11
Masukkan Tahun Lahir  : 2001
Masukkan Jalan     : Jl. Jalur Sutera Kav 29A
Masukkan Kota      : Tangerang Selatan
Masukkan Kode Pos   : 34131
=====

Berikut adalah data yang telah dimasukkan:
NIM      : 211011401309
Nama     : Nova Ardiansyah
Tanggal Lahir : 11/11/2001
Alamat   : Jl. Jalur Sutera Kav 29A, Tangerang Selatan, 3413.
|

```

- Lat3_3


```

#include <iostream>
#include <string>

using namespace std;

struct Mahasiswa
{
    char NIM[30];
    char Nama[25];
    char Alamat[40];
    char Umur[5];
};

int main()
{
    cout << "Nama \t: Nova Ardiansyah\n";
    cout << "NIM \t: 211011401309\n";
    cout << "=====\n\n";

    Mahasiswa Mhs[2];

    for (int i = 0; i < 2; i++)
    {
        cout << "Masukkan NIM \t: ";
        cin.getline(Mhs[i].NIM, 30);

        cout << "Masukkan Nama \t: ";
        cin.getline(Mhs[i].Nama, 25);

        cout << "Masukkan Alamat : ";
        cin.getline(Mhs[i].Alamat, 40);

        cout << "Masukkan Umur \t: ";
        cin.getline(Mhs[i].Umur, 5);

        cout << "=====\n\n";
    }

    cout << "Berikut adalah data yang telah dimasukkan: \n\n";
    for (int i = 0; i < 2; i++)
    {
        cout << "NIM \t: " << Mhs[i].NIM << endl;
        cout << "Nama \t: " << Mhs[i].Nama << endl;
        cout << "Alamat \t: " << Mhs[i].Alamat << endl;
        cout << "Umur \t: " << Mhs[i].Umur << endl;

        cout << "=====\n\n";
    }

    return 0;
}

```

```

/tmp/vKCcou4WJX.o
Nama      : Nova Ardiansyah
NIM       : 211011401309
=====

Masukkan NIM      : 211011401309
Masukkan Nama     : Nova Ardiansyah
Masukkan Alamat   : Jl. Jalur Sutra Kav 29A
Masukkan Umur     : 22
=====

Masukkan NIM      : 211011401310
Masukkan Nama     : Nova Ardiansyah 2
Masukkan Alamat   : Jl. Jalur Sutra Kav 29B
Masukkan Umur     : 22
=====

Berikut adalah data yang telah dimasukkan:

NIM      : 211011401309
Nama     : Nova Ardiansyah
Alamat   : Jl. Jalur Sutra Kav 29A
Umur     : 22
=====

NIM      : 211011401310
Nama     : Nova Ardiansyah 2
Alamat   : Jl. Jalur Sutra Kav 29B
Umur     : 22
=====

```

- Lat3_4

```

#include <iostream>
#include <string>

using namespace std;

struct Mahasiswa
{
    char NIM[30];
    char Nama[25];
    char Jalan[40];
    char Kota[40];
    char KodePos[40];
    char TanggalLahir[40];
};

int main()
{
    cout << "Nama \t: Nova Ardiansyah\n";
    cout << "NIM \t: 211011401309\n";
    cout << "=====\n\n";

    Mahasiswa Mhs[2];
}

```

```

for (int i = 0; i < 2; i++)
{
    cout << "Masukkan NIM \t\t: ";
    cin.getline(Mhs[i].NIM, 30);

    cout << "Masukkan Nama \t\t: ";
    cin.getline(Mhs[i].Nama, 25);

    cout << "Masukkan Jalan \t\t: ";
    cin.getline(Mhs[i].Jalan, 40);

    cout << "Masukkan Kota \t\t: ";
    cin.getline(Mhs[i].Kota, 40);

    cout << "Masukkan Kode Pos \t: ";
    cin.getline(Mhs[i].KodePos, 40);

    cout << "Masukkan Tgl Lahir \t: ";
    cin.getline(Mhs[i].TanggalLahir, 40);

    cout << "=====\n\n";
}

cout << "Berikut adalah data yang telah dimasukkan: \n\n";
for (int i = 0; i < 2; i++)
{
    cout << "NIM \t\t\t: " << Mhs[i].NIM << endl;
    cout << "Nama \t\t\t: " << Mhs[i].Nama << endl;
    cout << "Alamat \t\t\t: " << Mhs[i].Jalan << ", " << Mhs[i].Kota << ", " << Mhs[i].KodePos
    << ". " << endl;
    cout << "Tanggal Lahir \t: " << Mhs[i].TanggalLahir << endl;

    cout << "=====\n\n";
}

return 0;
}

```

Berikut adalah data yang telah dimasukkan:

```

NIM           : Nova Ardiansyah
Nama          : 211011401309
Alamat        : Jl. Jalur Sutera Kav 29A, Tangerang Selatan, 15131.
Tanggal Lahir : 11/11/2001
=====

```

```

NIM           : 211011401310
Nama          : Nova Ardiansyah 2
Alamat        : Jl. Jalur Sutera Kav 29B, Tangerang Selatan, 15131.
Tanggal Lahir : 11/11/2001
=====

```

- Lat3_5

```

#include <iostream>
#include <string>

using namespace std;

struct Mahasiswa

```

```

{
    char NIM[30];
    char Nama[25];
    int NilaiTugas;
    int NilaiUTS;
    int NilaiUAS;
    int NilaiAkhir;
    string NilaiHuruf;
};

int main()
{
    cout << "Nama \t: Nova Ardiansyah\n";
    cout << "NIM \t: 211011401309\n";
    cout << "=====\n\n";

    Mahasiswa Mhs;

    cout << "Masukkan NIM \t\t: ";
    cin.getline(Mhs.NIM, 30);

    cout << "Masukkan Nama \t\t: ";
    cin.getline(Mhs.Nama, 25);

    cout << "Masukkan Nilai Tugas \t: ";
    cin >> Mhs.NilaiTugas;

    cout << "Masukkan Nilai UTS \t\t: ";
    cin >> Mhs.NilaiUTS;

    cout << "Masukkan Nilai UAS \t\t: ";
    cin >> Mhs.NilaiUAS;

    cout << "=====\n\n";

    Mhs.NilaiAkhir = (Mhs.NilaiTugas * 0.2) + (Mhs.NilaiUTS * 0.35) + (Mhs.NilaiUAS * 0.45);

    if (Mhs.NilaiAkhir > 85 && Mhs.NilaiAkhir <= 100) {
        Mhs.NilaiHuruf = "A";
    } else if (Mhs.NilaiAkhir > 70 && Mhs.NilaiAkhir <= 85) {
        Mhs.NilaiHuruf = "B";
    } else if (Mhs.NilaiAkhir > 55 && Mhs.NilaiAkhir <= 70) {
        Mhs.NilaiHuruf = "C";
    } else if (Mhs.NilaiAkhir > 40 && Mhs.NilaiAkhir <= 55) {
        Mhs.NilaiHuruf = "D";
    } else if (Mhs.NilaiAkhir <= 40) {
        Mhs.NilaiHuruf = "E";
    }

    cout << "Berikut adalah hasil perhitungan nilai akhir Anda :\n\n";

    cout << "NIM \t\t\t: " << Mhs.NIM << endl;
    cout << "Nama \t\t\t: " << Mhs.Nama << endl;
    cout << "Nilai Tugas \t: " << Mhs.NilaiTugas << endl;
    cout << "Nilai UTS \t\t: " << Mhs.NilaiUTS << endl;
    cout << "Nilai UAS \t\t: " << Mhs.NilaiUAS << endl;
    cout << "Nilai Akhir \t: " << Mhs.NilaiAkhir << endl;
    cout << "Peringkat \t\t: " << Mhs.NilaiHuruf << endl;

    return 0;
}

```

```
}  
  
/tmp/icZaRkdvL4.o  
Nama      : Nova Ardiansyah  
NIM       : 211011401309  
=====
```

Masukkan NIM : 211011401309
Masukkan Nama : Nova Ardiansyah
Masukkan Nilai Tugas : 80
Masukkan Nilai UTS : 85
Masukkan Nilai UAS : 90
=====

Berikut adalah hasil perhitungan nilai akhir Anda :

NIM : 211011401309
Nama : Nova Ardiansyah
Nilai Tugas : 80
Nilai UTS : 85
Nilai UAS : 90
Nilai Akhir : 86
Peringkat : A

LAPORAN PRAKTIKUM STRUKTUR DATA

LAPORAN AKHIR PRAKTIKUM

Pertemuan ke-03

Structure



Disusun Oleh:

Nama Lengkap : Nova Ardiansyah

NIM : 211011401309

Kelas : 04-TPLE008

TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PAMULANG

Jl. Raya Puspitek No. 11 Buaran, Serpong Telp. (021) 7412566, Fax. (021) 7412566
Tangerang Selatan - Banten

A. TUGAS AKHIR

1. Buatlah program untuk membaca dan mencetak biodata pegawai, dengan menggunakan :
 - a) Stuktur BIODATA yang terdiri dari NIP, Nama, Alamat, Jabatan, Agama, Tanggal_Lahir, Tanggal_Mulai_Kerja, Unit_Kerja
 - b) Struktur TINGGAL yang terdiri dari Jalan, Kode_Pos, Kota dan
 - c) Stuktur TANGGAL yang terdiri dari Tanggal, Bulan, Tahun.Gunakan Array of Struct

Jawab :

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

struct TANGGAL {
    int tanggal;
    int bulan;
    int tahun;
};

struct TINGGAL {
    string jalan;
    string kode_pos;
    string kota;
};

struct BIODATA {
    string nip;
    string nama;
    string alamat;
    string jabatan;
    string agama;
    TANGGAL tanggal_lahir;
    TANGGAL tanggal_mulai_kerja;
    TINGGAL unit_kerja;
};

int main() {
    const int MAX_PEGAWAI = 2;
    BIODATA pegawai[MAX_PEGAWAI];

    cout << "Nama \t: Nova Ardiansyah\n";
    cout << "NIM \t: 211011401309\n";
    cout << "=====\n\n";

    int n;
    cout << "Masukkan jumlah pegawai: ";
    cin >> n;

    for (int i = 0; i < n; i++) {
        cout << "\nBiodata Pegawai " << i+1 << ":\n";
```

```

    cout << "NIP: ";
    cin >> pegawai[i].nip;

    cout << "Nama: ";
    cin >> pegawai[i].nama;

    cout << "Alamat: ";
    cin.ignore();
    getline(cin, pegawai[i].alamat);

    cout << "Jabatan: ";
    getline(cin, pegawai[i].jabatan);

    cout << "Agama: ";
    getline(cin, pegawai[i].agama);

    cout << "Tanggal Lahir (dd mm yyyy): ";
    cin >> pegawai[i].tanggal_lahir.tanggal >> pegawai[i].tanggal_lahir.bulan >>
    pegawai[i].tanggal_lahir.tahun;

    cout << "Tanggal Mulai Kerja (dd mm yyyy): ";
    cin >> pegawai[i].tanggal_mulai_kerja.tanggal >> pegawai[i].tanggal_mulai_kerja.bulan
    >> pegawai[i].tanggal_mulai_kerja.tahun;

    cout << "Unit Kerja:\n";
    cout << "Jalan: ";

    cin.ignore();
    getline(cin, pegawai[i].unit_kerja.jalan);

    cout << "Kode Pos: ";
    getline(cin, pegawai[i].unit_kerja.kode_pos);

    cout << "Kota: ";
    getline(cin, pegawai[i].unit_kerja.kota);
}

cout << "\nData Pegawai:\n";
for (int i = 0; i < n; i++) {
    cout << "\nBiodata Pegawai " << i+1 << ":\n";
    cout << "NIP: " << pegawai[i].nip << endl;
    cout << "Nama: " << pegawai[i].nama << endl;
    cout << "Alamat: " << pegawai[i].alamat << endl;
    cout << "Jabatan: " << pegawai[i].jabatan << endl;
    cout << "Agama: " << pegawai[i].agama << endl;
    cout << "Tanggal Lahir: " << pegawai[i].tanggal_lahir.tanggal << "/" <<
    pegawai[i].tanggal_lahir.bulan << "/" << pegawai[i].tanggal_lahir.tahun << endl;
    cout << "Tanggal Mulai Kerja: " << pegawai[i].tanggal_mulai_kerja.tanggal << "/" <<
    pegawai[i].tanggal_mulai_kerja.bulan << "/" << pegawai[i].tanggal_mulai_kerja.tahun <<
    endl;

    cout << "Unit Kerja:\n";
    cout << "Jalan: " << pegawai[i].unit_kerja.jalan << endl;
    cout << "Kode Pos: " << pegawai[i].unit_kerja.kode_pos << endl;
    cout << "Kota: " << pegawai[i].unit_kerja.kota << endl;
}

return 0;
}

```



```
Biodata Pegawai 1:  
NIP: 12345  
Nama: Nova  
Alamat: Lampung  
Jabatan: HRD  
Agama: Islam  
Tanggal Lahir: 11/11/2001  
Tanggal Mulai Kerja: 1/1/2021  
Unit Kerja:  
Jalan: Ahmad Yani  
Kode Pos: 15131  
Kota: Tangerang
```

2. KESIMPULAN

Dalam bahasa pemrograman C++, struktur merupakan tipe data yang memungkinkan programmer untuk mengelompokkan beberapa variabel dengan tipe data yang berbeda menjadi satu objek. Dengan menggunakan struktur, programmer dapat mengelola data dengan lebih terstruktur dan terorganisir, sehingga memudahkan pembacaan dan pemeliharaan kode. Kelebihan penggunaan struktur adalah memungkinkan programmer untuk mengelompokkan data terkait dalam satu objek, sehingga program lebih mudah dipahami dan dikelola. Struktur juga dapat disarangkan, sehingga memungkinkan pembuatan struktur yang kompleks dan terorganisir dengan baik. Struktur juga memudahkan programmer untuk memanipulasi data secara bersamaan, serta menghindari kesalahan penulisan variabel yang serupa.

Selain itu, struktur juga dapat dikombinasikan dengan array untuk membentuk struktur data yang lebih kompleks. Misalnya, struktur Mahasiswa dapat memiliki array nilai dengan beberapa elemen untuk menyimpan nilai-nilai dari beberapa mata kuliah. Program tersebut dapat meminta pengguna untuk memasukkan data mahasiswa, termasuk nama dan nilai-nilai, dan kemudian menampilkan data tersebut dalam bentuk tabel. Dengan menggunakan struktur dan array, programmer dapat membuat program yang lebih terstruktur dan mudah dipelihara. Namun, sebelum menggunakan struktur dalam program, programmer harus memahami konsep struktur dengan baik dan membuat desain struktur yang tepat untuk program yang dibuat.