

LAPORAN AKHIR PRAKTIKUM
MATA KULIAH
MODUL 5
JUDUL MODUL : RECORD DAN STRUCTURE



Nama : Kurnia Fajar Rahyudi Putra
No. BP : 2211512013
Hari/Tanggal : Senin/ 5 Juni 2023
Shift : 1

Dosen : Dodon Yendri, M.Kom

LABORATORIUM KOMPUTER DAN JARINGAN
DEPARTEMEN TEKNIK KOMPUTER
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS ANDALAS
PADANG
2023

BAB I

PENDAHULUAN

1.1 Tujuan

Mempraktekkan konsep record/structure pada bahasa pemrograman C++

1.2 Landasan Teori

1.2.1 Pengertian Struct

Structure (struktur) adalah kumpulan elemen data yang digabungkan menjadi satu kesatuan. Dengan kata lain, structure merupakan bentuk struktur data yang dapat menyimpan variabel-variabel dalam satu nama. Masing-masing elemen data dikenal dengan sebuah *field*. Masing-masing field dapat memiliki tipe data yang sama ataupun berbeda. Variabel-variabel tersebut memiliki kaitan satu sama lain. Walaupun field-field tersebut berada dalam satu kesatuan, namun masing-masing field tersebut tetap dapat diakses secara individual atau satu persatu. Field-field digabungkan menjadi satu dengan tujuan untuk kemudahan dalam operasinya. Struktur membantu mengatur data yang rumit, khususnya dalam program besar, karena struktur akan membuat sekelompok elemen data diperlakukan sebagai suatu unit.

Struktur (Structure) C++ adalah koleksi variabel dibawah sebuah nama, variabel – variabel ini dapat berbentuk berbagai type, yaitu sebagai berikut:

- Int
- Float
- Char
- Dan lain-lain

Perbedaan utama antara struktur dan array adalah bahwa dalam array memiliki tipe data yang sama, sedangkan struktur adalah sebuah koleksi dari variabel – variabel dibawah nama yang sama, dimana setiap elemen dapat saja memiliki tipe yang berbeda.

Struct mirip dengan class. Keduanya sama – sama mengelompokkan data dan fungsi. Perbedaan antara keduanya, diantaranya: default member access struct adalah public sedangkan default member access class adalah private,

struct tidak dapat digunakan sebagai parent di pewarisan sedangkan class dapat digunakan sebagai parent di pewarisan, dan masih banyak lagi.

1.2.2 Kelebihan Struct Pada Bahasa Pemrograman C++

- Kompatibel dengan bahasa pemrograman C,
- Lebih ringan dibandingkan class di c++,
- Inisialisasinya mudah,
- Dapat digunakan untuk mengelompokkan data dan fungsi.

1.2.3 Deklarasi Struct

Pendeklarasian structure selalu diawali kata baku **struct** diikuti nama structure dan deklarasi field-field yang membangun structure di antara pasangan tanda kurung kurawal buka dan kurung kurawal tutup yang diakhiri dengan tanda titik koma (;). Jika terdapat field dengan tipe data yang sama, dapat dideklarasikan bersama dengan dipisahkan oleh tanda koma (,) sehingga tidak perlu menuliskan tipe datanya berulang-ulang.

```
struct namaRecord {  
    tipe_atribut1 atribut1;  
    tipe_atribut2 atribut2;  
    tipe_atribut3 atribut3;  
};
```

Contoh:

```
struct dataMhs {  
    string nama;  
    int NIM;  
    string Departemen;  
    float IPK;  
};
```

1.2.4 Membuat Objek Struct Secara Langsung

Kita juga dapat membuat objek struct secara langsung pada saat struct diinisialisasi. Setelah membuat objeknya secara langsung, kita dapat mengaksesnya seperti biasa, seperti pada objek struct umumnya. Caranya, nama objek – objek tersebut kita letakkan di antara tutup kurung kurawal dan titik koma. Berikut ini contohnya:

```
struct dataMhs {  
    string nama;  
    int NIM;  
    string Departemen;  
    float IPK;  
} Mahasiswa;
```

1.2.5 Cara Mengakses Variabel di Dalam Struct

Cara mengakses variabel di dalam record/struct adalah dengan operator dot (.). Misalkan terdapat nama record/struct mhs, dan variabel yang akan diakses di dalamnya adalah nim, maka cara mengaksesnya adalah “Mahasiswa.NIM” yang artinya kita mengakses nim yang merupakan satu atribut dari data Mahasiswa.

1.2.6 Array dalam Struct

Suatu structure juga dideklarasikan menjadi sebuah array apabila hendak menggunakan satu struct untuk beberapa kali. Sebenarnya sama dengan struktur tunggal, perbedaannya adalah hanya pada saat pendeklarasian variabelnya saja. Bentuk umum pendeklarasian array dari suatu struktur, yaitu:

```
struct namaStrcut {  
    tipeAtribut1 Atribut1[batas Array];  
    tipeAtribut2 Atribut2[batas Array];  
    .....  
    tipeAtributN AtributN[batas Array];  
};
```

1.2.7 Nested Struct

Suatu struktur juga dapat mengandung suatu struktur yang lain. Artinya field-field dalam suatu structure merupakan suatu structure juga. Misalkan biodata Mahasiswa yang terdiri dari NIM, Nama, Alamat, dan Tanggal Lahir. Alamat terdiri dari Nama Jalan, Kota, dan Kode Pos. Demikian juga halnya dengan Tanggal terdiri dari Tanggal, Bulan, dan Tahun. Dengan demikian maka struktur dibagi dalam tiga struktur, yaitu:

```
struct adres {  
    string jalan, Kota;  
    int kodePos;  
};
```

```
struct born {  
    int tanggal, tahun;  
    string bulan;  
};
```

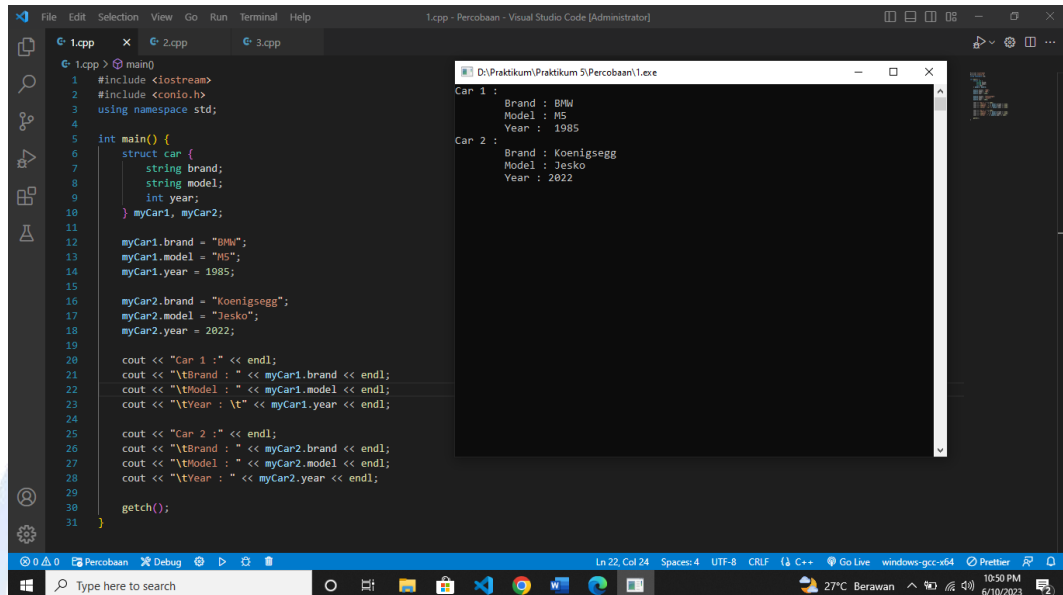
```
struct dataMhs {  
    string Nama;  
    int NIM;  
    born TTL;  
    adres Alamat;  
} Mahasiswa;
```

Perlu diingat bahwa jika suatu struktur dengan field bertipe struktur yang lain, maka struktur tersebut harus berada di atas struktur yang menggunakannya. Misalnya, perhatikan ketiga struktur di atas. Struktur Mahasiswa letaknya harus di bawah dari struktur Tanggal dan Tgl_Lahir. Sedangkan Tanggal dan struktur Tgl_Lahir boleh saling dipertukarkan posisinya, karena kedua struktur tidak saling terkait.

BAB II

PERCOBAAN

2.1 Percobaan 5.1



The screenshot shows the Visual Studio Code editor with a C++ file named 1.cpp. The code defines a struct 'car' with members 'brand', 'model', and 'year'. It then declares two variables, 'myCar1' and 'myCar2', of type 'car'. 'myCar1' is initialized with 'BMW', 'M5', and '1985'. 'myCar2' is initialized with 'Koenigsegg', 'Jesko', and '2022'. The program uses 'cout' to print the details of both cars. The output window on the right shows the execution results: 'Car 1 : Brand : BMW Model : M5 Year : 1985' and 'Car 2 : Brand : Koenigsegg Model : Jesko Year : 2022'.

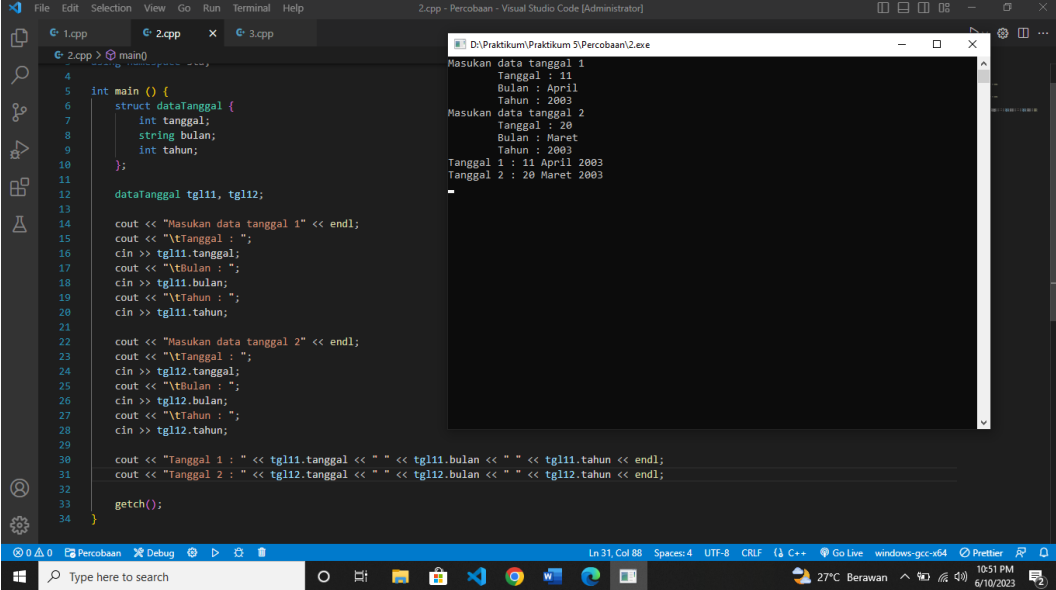
```
1.cpp > main()
1 #include <iostream>
2 #include <conio.h>
3 using namespace std;
4
5 int main() {
6     struct car {
7         string brand;
8         string model;
9         int year;
10    } myCar1, myCar2;
11
12    myCar1.brand = "BMW";
13    myCar1.model = "M5";
14    myCar1.year = 1985;
15
16    myCar2.brand = "Koenigsegg";
17    myCar2.model = "Jesko";
18    myCar2.year = 2022;
19
20    cout << "Car 1 : " << endl;
21    cout << "\tBrand : " << myCar1.brand << endl;
22    cout << "\tModel : " << myCar1.model << endl;
23    cout << "\tYear : " << myCar1.year << endl;
24
25    cout << "Car 2 : " << endl;
26    cout << "\tBrand : " << myCar2.brand << endl;
27    cout << "\tModel : " << myCar2.model << endl;
28    cout << "\tYear : " << myCar2.year << endl;
29
30    getch();
31 }
```

```
D:\Praktikum\Praktikum 5\percobaan1.exe
Car 1 :
Brand : BMW
Model : M5
Year : 1985
Car 2 :
Brand : Koenigsegg
Model : Jesko
Year : 2022
```

Pertama-tama, program mendefinisikan struktur car yang memiliki tiga anggota yaitu brand, model, dan year. Anggota-anggota tersebut menyimpan informasi tentang merek mobil, model mobil, dan tahun produksi mobil. Setelah mendefinisikan struktur, program mendeklarasikan dua variabel myCar1 dan myCar2 yang memiliki tipe car. Variabel-variabel ini akan digunakan untuk menyimpan data mobil yang diinputkan.

Selanjutnya, program mengisi nilai anggota-anggota dari myCar1 dengan menggunakan operator dot (titik) dan memberikan nilai spesifik untuk masing-masing anggota seperti merek "BMW", model "M5", dan tahun produksi 1985. Program juga mengisi nilai anggota-anggota dari myCar2 dengan merek "Koenigsegg", model "Jesko", dan tahun produksi 2022. Setelah mengisi nilai anggota dari kedua variabel, program mencetak informasi tentang mobil pertama (myCar1) dan mobil kedua (myCar2) menggunakan fungsi cout. Informasi yang dicetak meliputi merek, model, dan tahun produksi mobil.

2.2 Percobaan 5.2



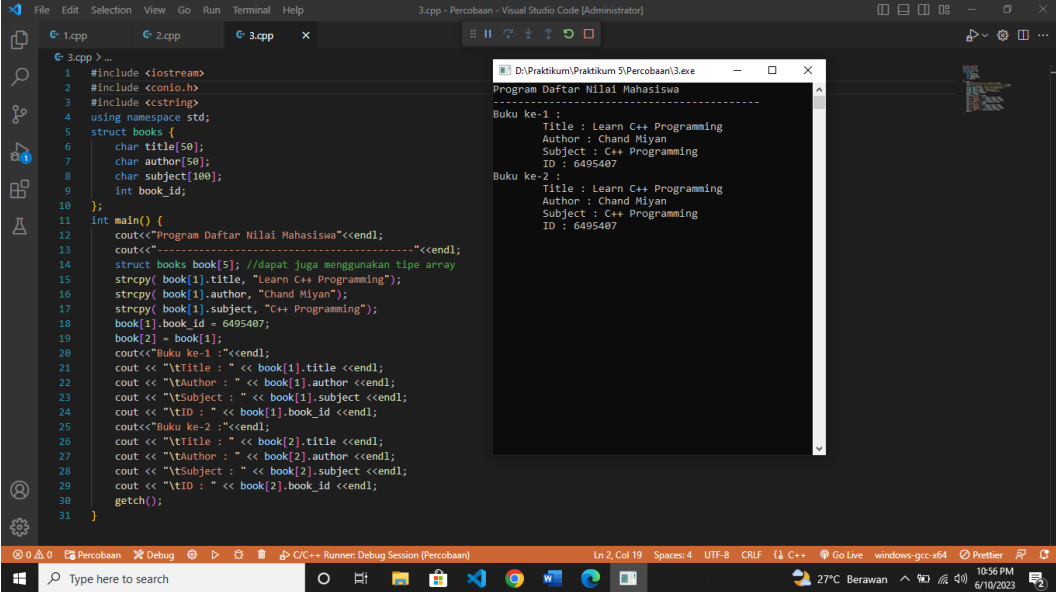
The screenshot shows the Visual Studio Code editor with a C++ file named `2.cpp`. The code defines a `dataTanggal` struct with `tanggal` (int), `bulan` (string), and `tahun` (int) members. It declares two variables, `tgl11` and `tgl12`, of type `dataTanggal`. The `main` function prompts the user to enter data for two dates. For each date, it asks for the day, month, and year, reads the input using `cin`, and then prints the stored data using `cout`. The output window shows the program's execution with the following input and output:

```
Masukan data tanggal 1
Tanggal : 11
Bulan : April
Tahun : 2003
Masukan data tanggal 2
Tanggal : 20
Bulan : Maret
Tahun : 2003
Tanggal 1 : 11 April 2003
Tanggal 2 : 20 Maret 2003
```

Program dimulai dengan mendefinisikan struktur `dataTanggal` yang memiliki tiga anggota yaitu `tanggal` (bertipe `int`), `bulan` (bertipe `string`), dan `tahun` (bertipe `int`). Struktur ini digunakan untuk menyimpan informasi tanggal dengan format tanggal, bulan, dan tahun. Setelah mendefinisikan struktur, program mendeklarasikan dua variabel `tgl11` dan `tgl12` yang memiliki tipe `dataTanggal`. Variabel-variabel ini akan digunakan untuk menyimpan data tanggal yang diinputkan.

Selanjutnya, program meminta pengguna memasukkan data tanggal 1 dan data tanggal 2 melalui fungsi `cin`. Pengguna diminta untuk memasukkan tanggal, bulan, dan tahun secara terpisah untuk kedua tanggal. Setelah semua data dimasukkan, program mencetak kembali informasi tanggal yang telah diinputkan menggunakan fungsi `cout`. Informasi tanggal yang dicetak mencakup tanggal, bulan, dan tahun untuk kedua tanggal.

2.3 Percobaan 5.3



The screenshot shows the Visual Studio Code editor with a C++ file named 3.cpp. The code defines a struct 'books' with members 'title', 'author', 'subject', and 'book_id'. In the main function, it prints the program title, defines an array of 5 'books' structures, and populates the first element with 'Learn C++ Programming', 'Chand Miyan', 'C++ Programming', and ID 6495407. It then prints the details of the first book and assigns the first book's data to the second book (index 2). Finally, it prints the details of the second book. The output window on the right shows the execution results, confirming that the second book's data is identical to the first.

```
1 #include <iostream>
2 #include <conio.h>
3 #include <cstring>
4 using namespace std;
5 struct books {
6     char title[50];
7     char author[50];
8     char subject[100];
9     int book_id;
10 };
11 int main() {
12     cout<<"Program Daftar Nilai Mahasiswa"<<endl;
13     cout<<"-----"<<endl;
14     struct books book[5]; //dapat juga menggunakan tipe array
15     strcpy( book[1].title, "Learn C++ Programming");
16     strcpy( book[1].author, "Chand Miyan");
17     strcpy( book[1].subject, "C++ Programming");
18     book[1].book_id = 6495407;
19     book[2] = book[1];
20     cout<<"Buku ke-1 : "<<endl;
21     cout << "\ttitle : " << book[1].title <<endl;
22     cout << "\tauthor : " << book[1].author <<endl;
23     cout << "\tsubject : " << book[1].subject <<endl;
24     cout << "\tid : " << book[1].book_id <<endl;
25     cout<<"Buku ke-2 : "<<endl;
26     cout << "\ttitle : " << book[2].title <<endl;
27     cout << "\tauthor : " << book[2].author <<endl;
28     cout << "\tsubject : " << book[2].subject <<endl;
29     cout << "\tid : " << book[2].book_id <<endl;
30     getch();
31 }
```

Program Daftar Nilai Mahasiswa

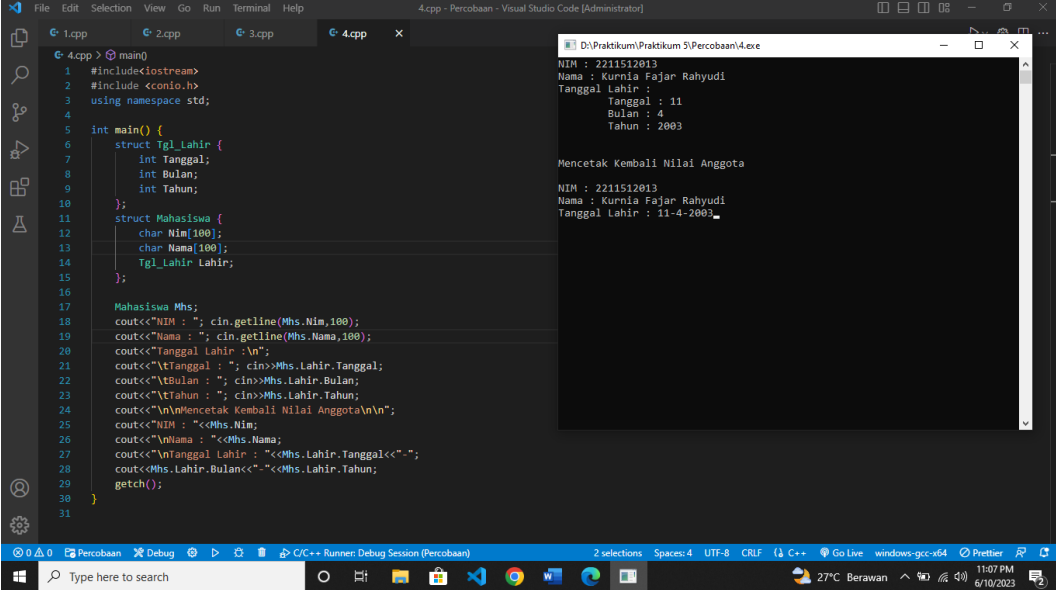
Buku ke-1 :
Title : Learn C++ Programming
Author : Chand Miyan
Subject : C++ Programming
ID : 6495407
Buku ke-2 :
Title : Learn C++ Programming
Author : Chand Miyan
Subject : C++ Programming
ID : 6495407

Program dimulai dengan mencetak judul program "Program Daftar Nilai Mahasiswa" dan garis pembatas menggunakan fungsi cout. Selanjutnya, program mendefinisikan struktur books yang memiliki empat anggota yaitu title (array char), author (array char), subject (array char), dan book_id (int). Struktur ini digunakan untuk menyimpan informasi tentang buku, seperti judul, penulis, subjek, dan ID buku.

Program mendeklarasikan sebuah array book dengan tipe books dan ukuran 5. Array ini akan digunakan untuk menyimpan data buku. Program menggunakan fungsi strcpy dari library <cstring> untuk menyalin nilai ke dalam anggota-anggota buku dengan indeks 1. Data buku tersebut meliputi judul "Learn C++ Programming", penulis "Chand Miyan", subjek "C++ Programming", dan ID buku 6495407. Selanjutnya, program menyalin buku dengan indeks 1 ke buku dengan indeks 2 menggunakan operator assignment (=).

Program mencetak informasi buku dengan indeks 1 menggunakan fungsi cout, termasuk judul, penulis, subjek, dan ID buku. Program mencetak informasi buku dengan indeks 2 menggunakan fungsi cout, yang sebelumnya telah disalin dari buku dengan indeks 1. Dalam hal ini, informasi buku dengan indeks 2 akan sama dengan informasi buku dengan indeks 1.

2.4 Percobaan 5.4



The screenshot shows the Visual Studio Code editor with a C++ file named 4.cpp. The code defines two structures: Tgl_Lahir and Mahasiswa. The Tgl_Lahir structure contains variables for Tanggal, Bulan, and Tahun. The Mahasiswa structure contains variables for Nim, Nama, and a Tgl_Lahir object. A main function declares a Mahasiswa variable Mhs, prompts the user to input NIM, Nama, and birth date (month and year), and then prints the stored data. The output window shows the program's execution with the input NIM: 2211512013, Nama: Kurnia Fajar Rahyudi, and birth date 11-4-2003, followed by the printed output.

```
1 #include<iostream>
2 #include <conio.h>
3 using namespace std;
4
5 int main() {
6     struct Tgl_Lahir {
7         int Tanggal;
8         int Bulan;
9         int Tahun;
10    };
11    struct Mahasiswa {
12        char Nim[100];
13        char Nama[100];
14        Tgl_Lahir Lahir;
15    };
16
17    Mahasiswa Mhs;
18    cout<<"NIM : "; cin.getline(Mhs.Nim,100);
19    cout<<"Nama : "; cin.getline(Mhs.Nama,100);
20    cout<<"Tanggal Lahir : \n";
21    cout<<"\tTanggal : "; cin>>Mhs.Lahir.Tanggal;
22    cout<<"\tBulan : "; cin>>Mhs.Lahir.Bulan;
23    cout<<"\tTahun : "; cin>>Mhs.Lahir.Tahun;
24    cout<<"\nMencetak Kembali Nilai Anggota\n\n";
25    cout<<"NIM : "<<Mhs.Nim;
26    cout<<"\nNama : "<<Mhs.Nama;
27    cout<<"\nTanggal Lahir : "<<Mhs.Lahir.Tanggal<<"-";
28    cout<<Mhs.Lahir.Bulan<<"- "<<Mhs.Lahir.Tahun;
29    getch();
30 }
31
```

Output:

```
NIM : 2211512013
Nama : Kurnia Fajar Rahyudi
Tanggal Lahir :
    Tanggal : 11
    Bulan : 4
    Tahun : 2003

Mencetak Kembali Nilai Anggota

NIM : 2211512013
Nama : Kurnia Fajar Rahyudi
Tanggal Lahir : 11-4-2003_
```

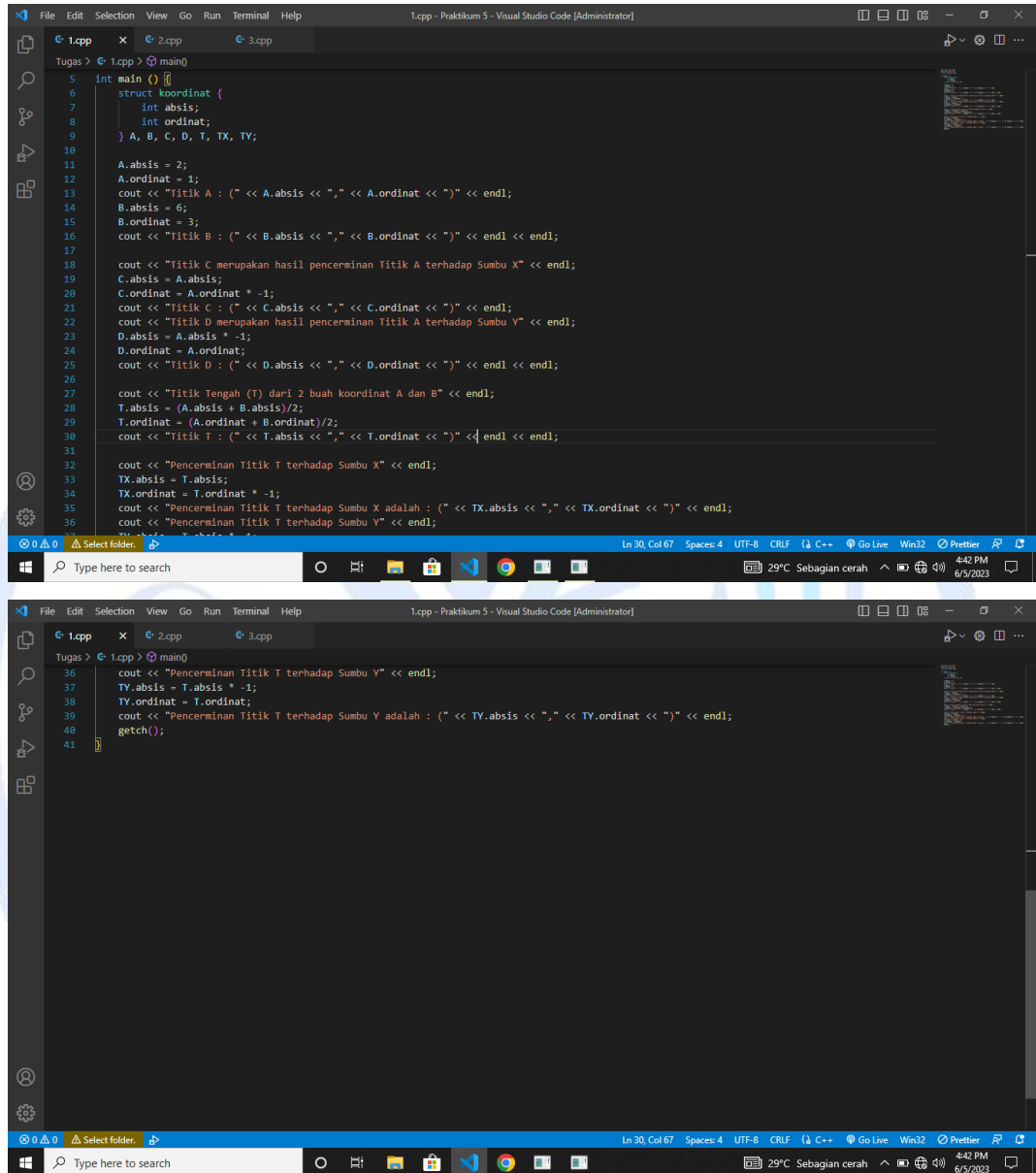
Program dimulai dengan mendefinisikan dua struktur (struct) yaitu Tgl_Lahir dan Mahasiswa. Struktur Tgl_Lahir digunakan untuk menyimpan informasi tanggal lahir, sedangkan struktur Mahasiswa digunakan untuk menyimpan data mahasiswa, termasuk NIM, Nama, dan tanggal lahir. Setelah mendefinisikan struktur, variabel Mhs dari tipe Mahasiswa dideklarasikan. Variabel ini akan digunakan untuk menyimpan data mahasiswa yang diinputkan.

Selanjutnya, program akan meminta pengguna memasukkan data mahasiswa melalui fungsi cin dan getline. Pengguna diminta untuk memasukkan NIM, Nama, dan tanggal lahir (tanggal, bulan, dan tahun). Setelah semua data dimasukkan, program mencetak kembali nilai anggota dari variabel Mhs menggunakan fungsi cout. Data yang dicetak adalah NIM, Nama, dan tanggal lahir dengan format "Tanggal-Bulan-Tahun".

BAB III

TUGAS PRAKTIKUM

3.1 Tugas 1



The image displays two screenshots of a Visual Studio Code editor window, showing C++ code for a program that calculates the reflection of points across the X and Y axes.

Top Screenshot: The code defines a `koordinat` struct with `absis` and `ordinat` members. It initializes points A, B, C, D, T, TX, and TY. It then calculates the reflection of point A across the Y-axis to get point C, and the reflection of point B across the X-axis to get point D. It also calculates the midpoint T of segment AB and its reflection TX across the Y-axis. The code uses `cout` to display the coordinates of these points.

```
5 int main() {
6     struct koordinat {
7         int absis;
8         int ordinat;
9     } A, B, C, D, T, TX, TY;
10
11     A.absis = 2;
12     A.ordinat = 1;
13     cout << "Titik A : (" << A.absis << ", " << A.ordinat << ")" << endl;
14     B.absis = 6;
15     B.ordinat = 3;
16     cout << "Titik B : (" << B.absis << ", " << B.ordinat << ")" << endl << endl;
17
18     cout << "Titik C merupakan hasil pencerminan Titik A terhadap Sumbu X" << endl;
19     C.absis = A.absis;
20     C.ordinat = A.ordinat * -1;
21     cout << "Titik C : (" << C.absis << ", " << C.ordinat << ")" << endl;
22     cout << "Titik D merupakan hasil pencerminan Titik A terhadap Sumbu Y" << endl;
23     D.absis = A.absis * -1;
24     D.ordinat = A.ordinat;
25     cout << "Titik D : (" << D.absis << ", " << D.ordinat << ")" << endl << endl;
26
27     cout << "Titik Tengah (T) dari 2 buah koordinat A dan B" << endl;
28     T.absis = (A.absis + B.absis)/2;
29     T.ordinat = (A.ordinat + B.ordinat)/2;
30     cout << "Titik T : (" << T.absis << ", " << T.ordinat << ")" << endl << endl;
31
32     cout << "Pencerminan Titik T terhadap Sumbu X" << endl;
33     TX.absis = T.absis;
34     TX.ordinat = T.ordinat * -1;
35     cout << "Pencerminan Titik T terhadap Sumbu X adalah : (" << TX.absis << ", " << TX.ordinat << ")" << endl;
36     cout << "Pencerminan Titik T terhadap Sumbu Y" << endl;
37     TY.absis = T.absis * -1;
38     TY.ordinat = T.ordinat;
39     cout << "Pencerminan Titik T terhadap Sumbu Y adalah : (" << TY.absis << ", " << TY.ordinat << ")" << endl;
40     getch();
41 }
```

Bottom Screenshot: This screenshot shows the same code as the top one, but with the `cout` statement for the reflection of point T across the Y-axis (line 39) commented out, and the `getch()` statement (line 40) highlighted.

```
D:\Praktikum\Praktikum 5\Tugas1.exe
Titik A : (2,1)
Titik B : (6,3)

Titik C merupakan hasil pencerminan Titik A terhadap Sumbu X
Titik C : (2,-1)
Titik D merupakan hasil pencerminan Titik A terhadap Sumbu Y
Titik D : (-2,1)

Titik Tengah (T) dari 2 buah koordinat A dan B
Titik T : (4,2)

Pencerminan Titik T terhadap Sumbu X
Pencerminan Titik T terhadap Sumbu X adalah : (4,-2)
Pencerminan Titik T terhadap Sumbu Y
Pencerminan Titik T terhadap Sumbu Y adalah : (-4,2)
```

3.2 Tugas 2

```
File Edit Selection View Go Run Terminal Help 2.cpp - Praktikum 5 - Visual Studio Code [Administrator]
Tugas > 2.cpp > main()
5 int main()
6 {
7     struct time {
8         int jam;
9         int menit;
10        int detik;
11    } jamDatang, jamPulang, selisihJam;
12
13    cout << "Jam datang : ";
14    cin >> jamDatang.jam;
15    cout << "Menit datang : ";
16    cin >> jamDatang.menit;
17    cout << "Detik datang : ";
18    cin >> jamDatang.detik;
19    cout << endl;
20
21    cout << "Jam pulang : ";
22    cin >> jamPulang.jam;
23    cout << "Menit pulang : ";
24    cin >> jamPulang.menit;
25    cout << "Detik pulang : ";
26    cin >> jamPulang.detik;
27    cout << endl;
28
29    /* KONVERSI JAM KE DETIK */
30    int jamDatang_detik = (jamDatang.jam * 3600) + (jamDatang.menit * 60) + jamDatang.detik;
31    int jamPulang_detik = (jamPulang.jam * 3600) + (jamPulang.menit * 60) + jamPulang.detik;
32
33    /* SELISIH JAM */
34    int selisihJam_detik = jamPulang_detik - jamDatang_detik;
35    selisihJam.jam = selisihJam_detik/3600;
36    selisihJam.menit = (selisihJam_detik % 3600)/60;
```

The image displays a C++ program in Visual Studio Code and its execution output in a terminal window. The program calculates parking fees based on arrival and departure times.

```
2.cpp - Praktikum 5 - Visual Studio Code [Administrator]
Tugas > 2.cpp > main()
32
33 /* SELISIH JAM */
34 int selisihJam_detik = jamPulang_detik - jamDatang_detik;
35 selisihJam.jam = selisihJam_detik/3600;
36 selisihJam.menit = (selisihJam_detik % 3600)/60;
37 selisihJam.detik = selisihJam_detik % 60;
38
39 /* MENCARI BIAYA PARKIR */
40 int cost;
41 if (selisihJam_detik % 3600 == 0) {
42     cost = selisihJam_detik/3600 * 3000;
43 } else {
44     cost = (selisihJam_detik/3600 + 1) * 3000;
45 }
46
47 cout << "Jam Datang\t" << jamDatang.jam << ":" << jamDatang.menit << ":" << jamDatang.detik << endl;
48 cout << "Jam Pulang\t" << jamPulang.jam << ":" << jamPulang.menit << ":" << jamPulang.detik << endl;
49 cout << "Lama Parkir\t" << selisihJam.jam << ":" << selisihJam.menit << ":" << selisihJam.detik << endl;
50 cout << "Biaya\t\t" << cost << endl;
51 getch();
52 }
```

Output of the program (D:\Praktikum\Praktikum 5\Tugas2.exe):

```
Jam datang : 10
Menit datang : 10
Detik datang : 10

Jam pulang : 11
Menit pulang : 25
Detik pulang : 30

Jam Datang
= 10:10:10
Jam Pulang
= 11:25:30
Lama Parkir
= 1:15:20
Biaya
= Rp. 6000
```

3.3 Tugas 3

```
File Edit Selection View Go Run Terminal Help 3.cpp - Praktikum 5 - Visual Studio Code [Administrator]
Tugas > 3.cpp > ...
5 int main () {
6     struct ttl {
7         int tanggal;
8         int bulan;
9         int tahun;
10        string tempat;
11    };
12    struct alamat {
13        string jalan;
14        int RT;
15        int RW;
16        string desa;
17        string kecamatan;
18    };
19    struct dataKTP {
20        string Nama;
21        ttl TTL;
22        string Jenkel;
23        alamat Alamat;
24        string Agama;
25        string Status;
26        string Pekerjaan;
27    } KTP;
28
29    cout << "Masukan Data Penduduk" << endl;
30    for (int i = 1; i <= 3; i++) {
31        cout << "Data Penduduk ke-" << i << endl;
32        cout << "Nama : ";
33        getline (cin, KTP.Nama);
34        cout << "Tempat Lahir : ";
35        getline (cin, KTP.TTL.tempat);
36        cout << "Tanggal Lahir : ";
```

```
File Edit Selection View Go Run Terminal Help 3.cpp - Praktikum 5 - Visual Studio Code [Administrator]
Tugas > 3.cpp > ...
32        cout << "Nama : ";
33        getline (cin, KTP.Nama);
34        cout << "Tempat Lahir : ";
35        getline (cin, KTP.TTL.tempat);
36        cout << "Tanggal Lahir : ";
37        cin >> KTP.TTL.tanggal;
38        cout << "Bulan Lahir : ";
39        cin >> KTP.TTL.bulan;
40        cout << "Tahun Lahir : ";
41        cin >> KTP.TTL.tahun;
42        cout << "Jenis Kelamin : ";
43        cin >> KTP.Jenkel;
44        cout << "Alamat : " << endl;
45        cin.ignore();
46        cout << "Nama Jalan : ";
47        getline (cin, KTP.Alatamat.jalan);
48        cout << "RT : ";
49        cin >> KTP.Alatamat.RT;
50        cout << "RW : ";
51        cin >> KTP.Alatamat.RW;
52        cin.ignore();
53        cout << "Nama Desa : ";
54        getline (cin, KTP.Alatamat.desa);
55        cout << "Nama Kecamatan : ";
56        getline (cin, KTP.Alatamat.kecamatan);
57        cout << "Agama : ";
58        getline (cin, KTP.Alatamat.Agama);
59        cout << "Status : ";
60        getline (cin, KTP.Status);
61        cout << "Pekerjaan : ";
62        getline (cin, KTP.Pekerjaan);
63        cout << endl << endl;
```

```
File Edit Selection View Go Run Terminal Help 3.cpp - Praktikum 5 - Visual Studio Code [Administrator]
Tugas > 3.cpp > ...
64
65 cout << "
66 cout << "KARTU TANDA PENDUDUK" << endl;
67 cout << "Nama\t\t\t: " << KTP.Nama << endl;
68 cout << "Tempat, Tgl Lahir\t: " << KTP.TTL.tempat << ", " << KTP.TTL.tanggal << "-" << KTP.TTL.bulan << "-" << KTP.TTL.tahun << endl;
69 cout << "Jenis Kelamin\t\t: " << KTP.Jenkel << endl;
70 cout << "Alamat\t\t\t\t\t: Jl. " << KTP.Alat.jalan << endl;
71 cout << "\t\t\t\t\t\t\t: " << KTP.Alat.RT << "/" << KTP.Alat.RW << endl;
72 cout << "\tDesa\t\t\t: " << KTP.Alat.desa << endl;
73 cout << "\tKecamatan\t: " << KTP.Alat.kecamatan << endl;
74 cout << "Agama\t\t\t\t: " << KTP.Alat.Agama << endl;
75 cout << "Status Perkawinan\t: " << KTP.Status << endl;
76 cout << "Pekerjaan\t\t\t: " << KTP.Pekerjaan << endl;
77 cout << "
78 }
79 getch();
80 }
```

```
D:\Praktikum\Praktikum 5\Tugas3.exe
Data Penduduk ke-1
Nama : KURNIA FAJAR RAHYUDI PUTRA
Tempat Lahir : KAMPUNG BARU
Tanggal Lahir : 11
Bulan Lahir : 04
Tahun Lahir : 2003
Jenis Kelamin : LAKI-LAKI
Alamat :
Nama Jalan : KOTO TUO
RT : 1
RW : 1
Nama Desa : KAPALO KOTO
Nama Kecamatan : PAUH
Agama : ISLAM
Status : BELUM KAWIN
Pekerjaan : MAHASISWA

KARTU TANDA PENDUDUK
Nama : KURNIA FAJAR RAHYUDI PUTRA
Tempat, Tgl Lahir : KAMPUNG BARU, 11-4-2003
Jenis Kelamin : LAKI-LAKI
Alamat : Jl. KOTO TUO
RT/RW : 1/1
Desa : KAPALO KOTO
Kecamatan : PAUH
Agama : ISLAM
Status Perkawinan : BELUM KAWIN
Pekerjaan : MAHASISWA

Data Penduduk ke-2
Nama : ADITYA AULIA RAHMAN
Tempat Lahir : BATUSANGKAR
Tanggal Lahir : 24
Bulan Lahir : 01
Tahun Lahir : 2004
Jenis Kelamin : LAKI-LAKI
Alamat :
Nama Jalan : DR. MOH HATTA
RT : 02
RW : 01
Nama Desa : LIMAU MANIS
Nama Kecamatan : PAUH
```

```
D:\Praktikum\Praktikum 5\Tugas3.exe
Masukan Data Penduduk
Data Penduduk ke-1
Nama : KURNIA FAJAR RAHYUDI PUTRA
Tempat Lahir : KAMPUNG BARU
Tanggal Lahir : 11
Bulan Lahir : 04
Tahun Lahir : 2003
Jenis Kelamin : LAKI-LAKI
Alamat :
Nama Jalan : KOTO TUO
RT : 01
RW : 01
Nama Desa : KAPALO KOTO
Nama Kecamatan : PAUH
Agama : ISLAM
Status : BELUM KAWIN
Pekerjaan : MAHASISWA

KARTU TANDA PENDUDUK
Nama : KURNIA FAJAR RAHYUDI PUTRA
Tempat, Tgl Lahir : KAMPUNG BARU, 11-4-2003
Jenis Kelamin : LAKI-LAKI
Alamat : Jl. KOTO TUO
RT/RW : 1/1
Desa : KAPALO KOTO
Kecamatan : PAUH
Agama : ISLAM
Status Perkawinan : BELUM KAWIN
Pekerjaan : MAHASISWA

Data Penduduk ke-2
Nama : ADITYA AULIA RAHMAN
Tempat Lahir : BATUSANGKAR
Tanggal Lahir : 24
Bulan Lahir : 01
Tahun Lahir : 2004
Jenis Kelamin : LAKI-LAKI
Alamat :
Nama Jalan : DR. MOH HATTA
RT : 02
RW : 03
Nama Desa : LIMA MANIS
Nama Kecamatan : PAUH
Agama : ISLAM
Status : BELUM KAWIN
Pekerjaan : MAHASISWA
```

```
D:\Praktikum\Praktikum 5\Tugas3.exe
Agama : ISLAM
Status Perkawinan : BELUM KAWIN
Pekerjaan : MAHASISWA

Data Penduduk ke-2
Nama : ADITYA AULIA RAHMAN
Tempat Lahir : BATUSANGKAR
Tanggal Lahir : 24
Bulan Lahir : 01
Tahun Lahir : 2004
Jenis Kelamin : LAKI-LAKI
Alamat :
Nama Jalan : DR. MOH HATTA
RT : 02
RW : 03
Nama Desa : LIMA MANIS
Nama Kecamatan : PAUH
Agama : ISLAM
Status : BELUM KAWIN
Pekerjaan : MAHASISWA

KARTU TANDA PENDUDUK
Nama : ADITYA AULIA RAHMAN
Tempat, Tgl Lahir : BATUSANGKAR, 24-1-2004
Jenis Kelamin : LAKI-LAKI
Alamat : Jl. DR. MOH HATTA
RT/RW : 2/3
Desa : LIMA MANIS
Kecamatan : PAUH
Agama : ISLAM
Status Perkawinan : BELUM KAWIN
Pekerjaan : MAHASISWA

Data Penduduk ke-3
Nama :
```



```
D:\Praktikum\Praktikum 9\Tugas3.exe
Nama      : ADITYA AULIA RAHMAN
Tempat, Tgl Lahir : BATUSANGKAR, 24-1-2004
Jenis Kelamin : LAKI-LAKI
Alamat    : Jl. DR. MOH HATTA
           RT/RW : 2/3
           Desa : LIMAU MANIS
           Kecamatan : PAUH
Agama     : ISLAM
Status Perkawinan : BELUM KAWIN
Pekerjaan : MAHASISWA

Data Penduduk ke-3
Nama      : YULIA HANDAYANI
Tempat Lahir : PARIAMAN
Tanggal Lahir : 24
Bulan Lahir : 03
Tahun Lahir : 2004
Jenis Kelamin : PEREMPUAN
Alamat    : Nama Jalan : DR. MOH HATTA
           RT : 01
           RW : 01
           Nama Desa : LIMAU MANIS
           Nama Kecamatan : PAUH
Agama     : ISLAM
Status Perkawinan : BELUM KAWIN
Pekerjaan : MAHASISWA

KARTU TANDA PENDUDUK
Nama      : YULIA HANDAYANI
Tempat, Tgl Lahir : PARIAMAN, 24-3-2004
Jenis Kelamin : PEREMPUAN
Alamat    : Jl. DR. MOH HATTA
           RT/RW : 1/1
           Desa : LIMAU MANIS
           Kecamatan : PAUH
Agama     : ISLAM
Status Perkawinan : BELUM KAWIN
Pekerjaan : MAHASISWA
```



BAB IV

ANALISA

4.1 Tugas 1

4.1.1 Poin A

Program dimulai dengan mendefinisikan struktur koordinat yang memiliki dua anggota yaitu absis (int) dan ordinat (int). Struktur ini digunakan untuk menyimpan koordinat titik dengan nilai absis dan ordinat. Selanjutnya, program mendeklarasikan beberapa variabel titik A, B, C, D, T, TX, dan TY dengan tipe koordinat.

Program menginisialisasi koordinat titik A dengan absis 2 dan ordinat 1, kemudian mencetak koordinat titik A menggunakan fungsi cout. Program juga menginisialisasi koordinat titik B dengan absis 6 dan ordinat 3, kemudian mencetak koordinat titik B menggunakan fungsi cout.

4.1.2 Poin B

Program melakukan pencerminan titik A terhadap sumbu X dan menyimpannya dalam variabel titik C dengan cara mengganti tanda ordinat menjadi negatif (-1). Program mencetak koordinat titik C sebagai hasil pencerminan titik A terhadap sumbu X. Program juga melakukan pencerminan titik A terhadap sumbu Y dan menyimpannya dalam variabel titik D dengan cara mengganti tanda absis menjadi negatif (-1). Program mencetak koordinat titik D sebagai hasil pencerminan titik A terhadap sumbu Y.

4.1.3 Poin C

Program mencari titik tengah (T) dari dua koordinat A dan B dengan cara menjumlahkan absis dan ordinat titik A dan B, kemudian membaginya dengan 2. Program mencetak koordinat titik T sebagai hasil.

Program melakukan pencerminan titik T terhadap sumbu X dan menyimpannya dalam variabel titik TX dengan cara mengganti tanda ordinat menjadi negatif (-1). Program mencetak koordinat titik TX sebagai hasil pencerminan titik T terhadap sumbu X. Program juga melakukan pencerminan titik T terhadap sumbu Y dan menyimpannya dalam variabel titik TY dengan cara mengganti tanda absis menjadi negatif (-1). Program mencetak koordinat titik TY sebagai hasil pencerminan titik T terhadap sumbu Y.

4.2 Tugas 2

4.2.1 Poin A

Program dimulai dengan mendefinisikan struktur time yang memiliki tiga anggota yaitu jam (int), menit (int), dan detik (int). Struktur ini digunakan untuk menyimpan informasi waktu.

Selanjutnya, program mendeklarasikan beberapa variabel yaitu jamDatang, jamPulang, dan selisihJam dengan tipe time. Variabel ini akan digunakan untuk menyimpan jam datang, jam pulang, dan selisih waktu antara kedua jam tersebut.

Program meminta pengguna memasukkan jam datang dengan menggunakan cin dan menyimpannya pada anggota jam dari variabel jamDatang. Hal yang sama dilakukan untuk memasukkan menit dan detik. Program meminta pengguna memasukkan jam pulang dengan menggunakan cin dan menyimpannya pada anggota jam dari variabel jamPulang. Hal yang sama dilakukan untuk memasukkan menit dan detik.

4.2.2 Poin B

Program menghitung selisih waktu (dalam detik) antara jam pulang dan jam datang dengan mengurangi nilai detik pulang dengan detik datang. Program menghitung selisih waktu dalam jam, menit, dan detik dengan membagi selisih waktu detik dengan 3600 untuk mendapatkan jam, mengambil sisa hasil bagi dengan 3600 dan membaginya dengan 60 untuk mendapatkan menit, dan mengambil sisa hasil bagi dengan 60 untuk mendapatkan detik.

4.2.3 Poin C

Program menghitung biaya parkir dengan aturan sebagai berikut:

- Jika selisih waktu detik habis dibagi 3600, maka biaya parkir dihitung berdasarkan jumlah jam ($\text{selisih waktu detik} / 3600$) dikali 3000.
- Jika selisih waktu detik tidak habis dibagi 3600, maka biaya parkir dihitung berdasarkan jumlah jam ($\text{selisih waktu detik} / 3600 + 1$) dikali 3000.

Program mencetak jam datang, jam pulang, lama parkir, dan biaya parkir menggunakan fungsi cout.

4.3 Tugas 3

4.3.1 Poin A

Program dimulai dengan mendefinisikan beberapa struktur yaitu `ttl` (tanggal lahir), `adres` (alamat), dan `dataKTP` (data KTP). Struktur `ttl` digunakan untuk menyimpan informasi tanggal lahir, struktur `adres` digunakan untuk menyimpan informasi alamat, dan struktur `dataKTP` digunakan untuk menyimpan informasi KTP. Program mendeklarasikan variabel `KTP` dengan tipe `dataKTP`. Variabel ini akan digunakan untuk menyimpan data KTP individu.

4.3.2 Poin B

Program meminta pengguna memasukkan data penduduk sebanyak 3 kali menggunakan loop `for`. Setiap iterasi loop, program akan meminta pengguna memasukkan data individu.

Program menggunakan fungsi `getline(cin, variable)` untuk membaca input pengguna yang berupa string, termasuk nama, tempat lahir, dan alamat. Program menggunakan operasi `cin` dan `getline(cin, variable)` untuk membaca input pengguna yang berupa bilangan bulat, seperti tanggal lahir, bulan lahir, tahun lahir, RT, dan RW.

4.3.3 Poin C

Program mencetak ulang data yang telah dimasukkan oleh pengguna menggunakan fungsi `cout`. Data tersebut mencakup nama, tempat lahir, tanggal lahir, jenis kelamin, alamat, agama, status, dan pekerjaan. Setelah data individu selesai dimasukkan dan dicetak, program akan kembali ke awal loop untuk memasukkan data individu berikutnya.

BAB V

PENUTUP

5.1 Kesimpulan

Record adalah suatu tipe data yang merupakan kumpulan dari atribut-atribut (field) suatu objek. Pada record tipe elemen bisa berbeda-beda tidak seperti array yang mengharuskan mempunyai tipe elemen yang sama. Cara mengakses variabel di dalam record/struct adalah dengan operator dot (.). Fungsi record yaitu menyimpan sejumlah data dengan nilai dan tipe data yang berbeda dalam satu wadah. Selain itu, record juga merupakan suatu tipe data terstruktur yang dapat menampung data field bertipe berbeda. Tipe ini dapat dipadukan dengan tipe data terstruktur lainnya seperti array.

5.2 Saran

Sebelum memulai membuat program, hendaknya mempelajari dan memahami proses dan cara kerja dari setiap materi yang akan dipraktikkan agar tidak terjadi error, sehingga program dapat dijalankan dengan lancar.

DAFTAR PUSTAKA

Anonim. (2003). *MODUL 5 PRAKTIKUM ALGORITMA DAN PEMECAHAN MASALAH*. Laboratorium Komputer dan Jaringan.

Kie, J. (2023, Maret 2023). Retrieved from [https://josikie.com/apa-itu-struct-di-c/#:~:text=Structures%20\(struct\)%20pada%20C%2B%2B%20adalah,disebut%20dengan%20anggota%20dari%20struct](https://josikie.com/apa-itu-struct-di-c/#:~:text=Structures%20(struct)%20pada%20C%2B%2B%20adalah,disebut%20dengan%20anggota%20dari%20struct).

Pintarkom. (2023, Juni 10). Retrieved from <https://pintarkom.com/pengertian-dan-contoh-program-struct-pada-c/>

