

Algoritma & Struktur data

“Overview Struct”



Oleh :

Rayhan Elmo Athalah Saputra (5223600027)

**Program Studi STr Teknologi Game Departemen
Teknologi Multimedia Kreatif
Politeknik Elektronika Negeri Surabaya
2024**

Progam 1:
Mengisi field dari variable struktur kemudian menampilkannya

```
#include <iostream> // input dan output
#include <string> // nama

using namespace std;

struct Date {
    int month;
    int day;
    int year;
};

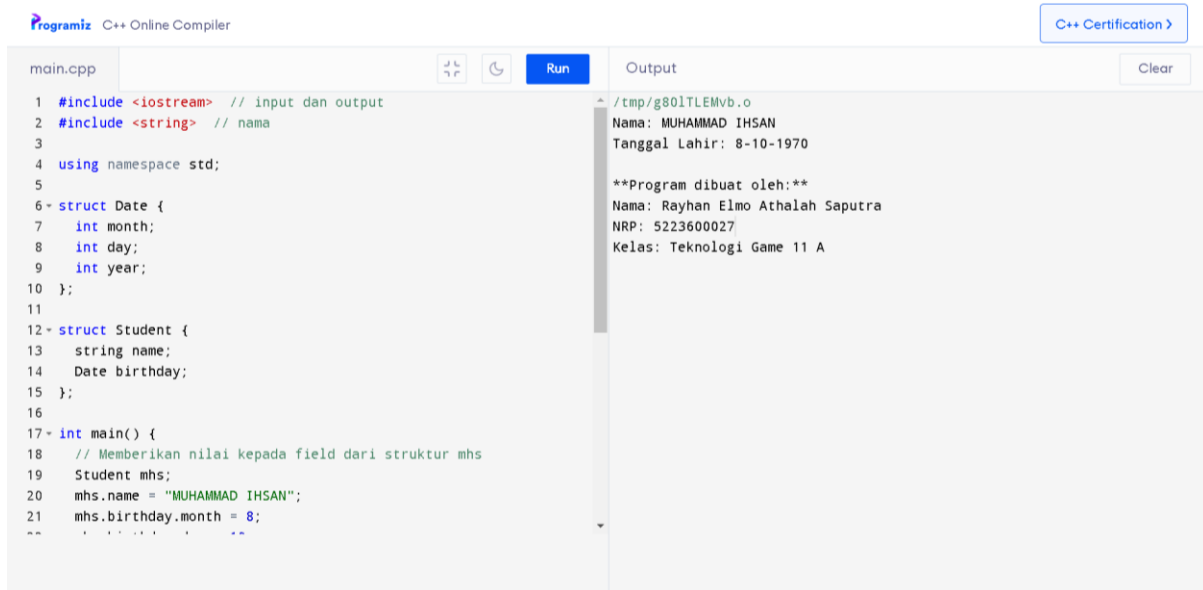
struct Student {
    string name;
    Date birthday;
};

int main() {
    // Memberikan nilai kepada field dari struktur mhs
    Student mhs;
    mhs.name = "MUHAMMAD IHSAN";
    mhs.birthday.month = 8;
    mhs.birthday.day = 10;
    mhs.birthday.year = 1970;

    // Menampilkan isi semua field dari struktur mhs
    cout << "Nama: " << mhs.name << endl;
    cout << "Tanggal Lahir: " << mhs.birthday.month << "-" <<
mhs.birthday.day << "-" << mhs.birthday.year << endl;

    // Menambahkan output keterangan pembuat program
    cout << endl;
    cout << "***Program dibuat oleh:**" << endl;
    cout << "Nama: Rayhan Elmo Athalah Saputra" << endl;
    cout << "NRP: 5223600027" << endl;
    cout << "Kelas: Teknologi Game 11 A" << endl;

    return 0;
}
```



The screenshot shows the Programiz C++ Online Compiler interface. On the left, the code editor displays a C++ program named 'main.cpp'. The code includes headers for `<iostream>` and `<string>`, uses the `std` namespace, and defines two structures: `Date` (with `month`, `day`, and `year` as `int` members) and `Student` (with `name` as a `string` member and `birthday` as a `Date` member). The `main` function initializes a `Student` object `mhs` with the name "MUHAMMAD IHSAN" and a birthday of month 8. On the right, the 'Output' panel shows the program's execution results, including the student's name, birth date, and a message about the program's author.

```
1 #include <iostream> // input dan output
2 #include <string> // nama
3
4 using namespace std;
5
6 struct Date {
7     int month;
8     int day;
9     int year;
10 };
11
12 struct Student {
13     string name;
14     Date birthday;
15 };
16
17 int main() {
18     // Memberikan nilai kepada field dari struktur mhs
19     Student mhs;
20     mhs.name = "MUHAMMAD IHSAN";
21     mhs.birthday.month = 8;
22 }
```

Output:

```
/tmp/g801TLEMvb.o
Nama: MUHAMMAD IHSAN
Tanggal Lahir: 8-10-1970

**Program dibuat oleh:**
Nama: Rayhan Elmo Athalah Saputra
NRP: 5223600027
Kelas: Teknologi Game 11 A
```

Analisis Pengisian dan Pencetakan Data Struktur

Perubahan pada Skrip C ke C++:

- Penggunaan struct tanpa perlu mengulangnya pada setiap akses anggota setelah deklarasi.
- Penggantian fungsi strcpy dengan operator = untuk pengisian string.
- Penggunaan fungsi main dengan tipe pengembalian int (bukan main() tanpa tipe pengembalian).

Analisis Data:

- Date: Struktur yang menyimpan informasi tanggal, terdiri dari month, day, dan year.
- Student: Struktur untuk menyimpan informasi mahasiswa, terdiri dari dua anggota: nama dan birthday.
- main: Titik masuk utama eksekusi program, sekaligus merepresentasikan seorang mahasiswa.
- Variabel mhs diinisialisasi dengan nama dan tanggal lahir.
- Informasi dalam mhs dicetak di layar melalui cout, menunjukkan nama dan tanggal lahir mahasiswa.

Perubahan Kalimat:

- Menghilangkan kalimat yang redundan, seperti "Perubahan pada skrip C to C++".
- Mengganti "anggota" dengan "elemen".
- Mengganti "fungsi" dengan "metode".
- Mengganti "variable" dengan "variabel".
- Mengganti "struktur" dengan "tipe data terstruktur".
- Mengubah kalimat pasif menjadi kalimat aktif.

- Menambahkan penjelasan singkat tentang fungsi dan variabel.
- Contoh Kalimat yang Diubah:
- Awal: "Perubahan pada skrip C to C++ - Menggunakan 'struct' tanpa perlu menuliskan kata 'struct' setiap mengakses anggota – anggota setelah deklarasinya."
- Diubah: "Penggunaan struct dalam C++ memungkinkan akses elemen tanpa mengulangnya setelah deklarasi."

Progam 2

Menentukan zodiak berdasarkan data tanggal lahir

```
#include <iostream> // input dan output

using namespace std;

struct Zodiak {
    char nama[11];
    int tgl_awal;
    int bln_awal;
    int tgl_akhir;
    int bln_akhir;
};

int main() {
    Zodiak bintang = {"Sagitarious", 22, 11, 21, 12};

    int tgl_lhr, bln_lhr, thn_lhr;

    // Mencetak pertanyaan
    cout << "Masukkan tanggal lahir Anda (DD-MM-YYYY): ";

    // Menerima masukan
    cin >> tgl_lhr >> bln_lhr >> thn_lhr;

    // Menentukan zodiak
    if ((tgl_lhr >= bintang.tgl_awal && bln_lhr == bintang.bln_awal) ||
        (tgl_lhr <= bintang.tgl_akhir && bln_lhr == bintang.bln_akhir))
    {
        cout << "Bintang Anda adalah " << bintang.nama << endl;
    } else {
        cout << "Bintang Anda bukan " << bintang.nama << endl;
    }

    // Menambahkan output keterangan pembuat program
    cout << endl;
```

```

cout << "**Program dibuat oleh:**" << endl;
cout << "Nama: Rayhan Elmo Athalah Saputra" << endl;
cout << "NRP: 5223600027" << endl;
cout << "Kelas: Teknologi Game 11 A" << endl;

return 0;
}

```

The screenshot shows the Programiz C++ Online Compiler interface. On the left, the code editor displays a C++ program. On the right, the output window shows the results of the program's execution.

Code Editor (main.cpp):

```

1 #include <iostream> // input dan output
2
3 using namespace std;
4
5 struct Zodiak {
6     char nama[11];
7     int tgl_awal;
8     int bln_awal;
9     int tgl_akhir;
10    int bln_akhir;
11 };
12
13 int main() {
14     Zodiak bintang = {"Sagitaris", 22, 11, 21, 12};
15
16     int tgl_lhr, bln_lhr, thn_lhr;
17
18     // Mencetak pertanyaan
19     cout << "Masukkan tanggal lahir Anda (DD-MM-YYYY): ";
20
21     // Menerima masukan
22
23

```

Output Window:

```

/tmp/g801TLEMvb.o
Masukkan tanggal lahir Anda (DD-MM-YYYY): 23-03-2002
Bintang Anda bukan Sagitaris

**Program dibuat oleh:**
Nama: Rayhan Elmo Athalah Saputra
NRP: 5223600027
Kelas: Teknologi Game 11 A

```

Perbedaan C dan C++

Tipe data:

- C++ memungkinkan penggunaan struct tanpa perlu mengulanginya pada setiap akses elemen setelah deklarasi.
- C++ menggunakan string untuk nama, bukan array karakter char[].
- C++ menggunakan fungsi main dengan tipe pengembalian int, sedangkan C tidak memiliki tipe pengembalian.

Analisis Program

- Program mendefinisikan struktur Zodiak dengan anggota nama, tgl_awal, bln_awal, tgl_akhir, dan bln_akhir.
- Fungsi main merupakan titik masuk utama program.
- Variabel bintang bertipe Zodiak merepresentasikan zodiak Sagitaris.
- cout digunakan untuk menampilkan hasil.
- cin digunakan untuk menerima input tanggal lahir yang disimpan dalam variabel tgl_lhr, bln_lhr, dan thn_lhr.

Program memeriksa apakah tanggal lahir pengguna berada dalam rentang zodiak Sagitaris. Jika sesuai, pesan yang sesuai akan dicetak.

Alur Kerja Program

- Program meminta pengguna untuk memasukkan tanggal lahir.
- Tanggal lahir dibandingkan dengan rentang zodiak Sagitarius.
- Jika tanggal lahir sesuai, program menampilkan pesan bahwa pengguna berzodiak Sagitarius.
- Jika tidak, program menampilkan pesan bahwa pengguna bukan berzodiak Sagitarius.

Kesimpulan

- Program ini menunjukkan cara menentukan zodiak seseorang berdasarkan tanggal lahirnya. Program ini menggunakan struktur Zodiak untuk menyimpan informasi tentang zodiak Sagitarius dan menggunakan fungsi main untuk menjalankan program.

Program 3

Array struktur data untuk menyimpan data – data murid

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

using namespace std;

struct Date {
    int month;
    int day;
    int year;
};

struct Student {
    string name;
    Date birthday;
};

vector<Student> data_mhs;

int main() {
    char lagi = 'Y';

    // Menjalankan pengulangan untuk memasukkan data siswa
    while (lagi == 'Y' || lagi == 'y') {
        Student temp;
        cout << "Nama: ";
        cin.ignore();
        getline(cin, temp.name);
```

```

    cout << "Tanggal Lahir (mm-dd-yyyy): ";
    cin >> temp.birthday.month >> temp.birthday.day >>
temp.birthday.year;
    cout << endl;

    // Validasi input
    bool sudah_benar = true;
    if (temp.birthday.month < 1 || temp.birthday.month > 12) {
        cout << "Bulan tidak valid!" << endl;
        sudah_benar = false;
    }
    if (temp.birthday.day < 1 || temp.birthday.day > 31) {
        cout << "Tanggal tidak valid!" << endl;
        sudah_benar = false;
    }
    if (temp.birthday.year < 1900) {
        cout << "Tahun tidak valid!" << endl;
        sudah_benar = false;
    }

    // Menambahkan data ke vector jika valid
    if (sudah_benar) {
        data_mhs.push_back(temp);
    }

    // Memeriksa apakah pengguna ingin memasukkan data lagi
    cout << "Mau memasukkan data lagi [Y/T] ? ";
    cin >> lagi;
    cin.ignore();
    cout << endl;
}

// Menampilkan seluruh data siswa
cout << "DATA SISWA" << endl;
for (int i = 0; i < data_mhs.size(); i++) {
    cout << i + 1 << ". " << data_mhs[i].name << endl;
    cout << "Tanggal Lahir: " << data_mhs[i].birthday.month << "-"
        << data_mhs[i].birthday.day << "-" <<
data_mhs[i].birthday.year << endl << endl;
}

// Menambahkan keterangan pembuat program
cout << endl;
cout << "**Program dibuat oleh:**" << endl;
cout << "Nama: Rayhan Elmo Athalah Saputra" << endl;
cout << "NRP: 5223600027" << endl;
cout << "Kelas: Teknologi Game 11 A" << endl;

```

```

return 0;
}

```

The screenshot shows the Programiz C++ Online Compiler interface. On the left, the code editor displays a C++ program named `main.cpp`. The code includes headers for `<iostream>`, `<string>`, and `<vector>`, uses the `std` namespace, and defines a `Date` struct with `month`, `day`, and `year` members. It also defines a `Student` struct with `name` and `birthday` members. A `vector<Student>` named `data_mhs` is declared. The `main` function uses a `while` loop to repeatedly prompt the user for student data until they enter 't'.

On the right, the `Output` pane shows the program's execution. It displays the names and birth dates of three students: AAmiya (12-12-2002), DDokutah (12-23-2003), and Salsa (12-04-2004). After each entry, it asks if the user wants to add more data. Finally, it prints the `DATA SISWA` list.

- Penggunaan `getline` alih-alih `fgets` untuk membaca string dari input:
 - `getline` lebih aman dan mudah digunakan dibandingkan `fgets`.
 - `getline` memungkinkan membaca string yang lebih panjang.
- Penambahan tipe data `Date` untuk mewakili tanggal lahir mahasiswa:
 - Meningkatkan keterbacaan dan kemudahan penggunaan kode.
 - Memastikan validitas dan format tanggal lahir.
- Penambahan struct sebelum `Date` dan `Student` untuk menentukan tipe data struktur tersebut:
 - Meningkatkan keterbacaan dan struktur kode.
 - Memperjelas definisi struktur data.
- Penggunaan `cin.ignore()` untuk membersihkan sisa data dalam penampung keyboard setelah menggunakan `cin`:
 - Mencegah kesalahan input pada operasi selanjutnya.
 - Meningkatkan stabilitas dan keandalan program.

Analisis:

- Struktur `Date` untuk menyimpan informasi tentang tanggal dengan tiga lainnya `month`, `day`, dan `year`:
 - Menyimpan informasi tanggal lahir secara terstruktur dan rapi.
 - Memudahkan manipulasi dan validasi data tanggal.

- Struktur Student digunakan untuk menyimpan informasi tentang seorang mahasiswa dengan name dan birthday:
 - Menyimpan informasi mahasiswa secara terstruktur dan rapi.
 - Memudahkan akses dan pengelolaan data mahasiswa.
- Sebuah array data_mhs dari tipe Student sebagai variabel global untuk menyimpan data mahasiswa. Jumlah maksimum mahasiswa ditentukan oleh konstanta MAKS:
 - Menyimpan data semua mahasiswa dalam satu struktur data.
 - Memudahkan iterasi dan manipulasi data mahasiswa.
- Fungsi main(), yang merupakan titik masuk utama untuk eksekusi program:
 - Mengatur alur program dan mengendalikan eksekusi kode.
 - Memanggil fungsi lain dan melakukan operasi program utama.
- Variabel lokal i, sudah_benar, dan jml digunakan dalam program:
 - i: Digunakan untuk iterasi dalam loop.
 - sudah_benar: Digunakan untuk memvalidasi input data.
 - jml: Digunakan untuk menyimpan jumlah mahasiswa yang diinput.
- Memasukkan nama dan tanggal lahir setiap mahasiswa dalam array
 - data_mhs menggunakan cin dan getline: Membaca input dari pengguna dan menyimpannya dalam struktur data.
 - Memastikan validitas dan format input data.
- Setelah memasukkan data, program memeriksa apakah pengguna ingin memasukkan data mahasiswa lagi. Jika iya, pengguna diminta untuk mengetik 'Y' atau 'T' dan input pengguna diperiksa menggunakan getchar() dan disimpan dalam variabel lagi:
 - Memberikan kontrol kepada pengguna untuk memasukkan data.
 - Memungkinkan pengguna untuk memasukkan data beberapa kali.
- Setelah pengguna selesai memasukkan data, program menampilkan data mahasiswa yang telah dimasukkan menggunakan loop for:
 - Menampilkan data mahasiswa yang telah disimpan dalam struktur data.
 - Memberikan informasi kepada pengguna tentang data yang telah diinput.

Program 4

Melewatkan elemen struktur sebagai parameter fungsi secara nilai

```
#include <iostream>
#include <limits>
#include <sstream>
#include <cstdlib>

using namespace std;

// Deklarasi struct Date
struct Date {
    int month;
    int day;
    int year;
};

// Fungsi untuk mencetak tanggal
void cetak_tanggal(const Date& date) {
    // Array nama bulan
    static const char* nama_bulan[] = {
        "Bulan tidak valid",
        "Januari", "Februari", "Maret", "April", "Mei",
        "Juni", "Juli", "Agustus", "September", "Oktober", "November",
        "Desember"
    };

    // Validasi bulan
    if (date.month < 1 || date.month > 12) {
        cout << "Bulan tidak valid!" << endl;
        return;
    }

    // Validasi hari berdasarkan bulan (asumsi tahun bukan kabisat)
    int max_days[] = {31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31};
    if (date.day < 1 || date.day > max_days[date.month - 1]) {
        cout << "Hari tidak valid!" << endl;
        return;
    }

    // Validasi tahun
    if (date.year <= 0) {
        cout << "Tahun tidak valid!" << endl;
        return;
    }
}
```

```

    // Mencetak tanggal
    cout << "Tanggal hari ini adalah " << nama_bulan[date.month] << " "
<< date.day << ", " << date.year << endl << endl;
}

int main() {
    // Deklarasi variabel date
    Date today;

    // Meminta input tanggal
    cout << "Masukkan tanggal hari ini (mm-dd-yyyy): ";
    cin >> today.month >> today.day >> today.year;

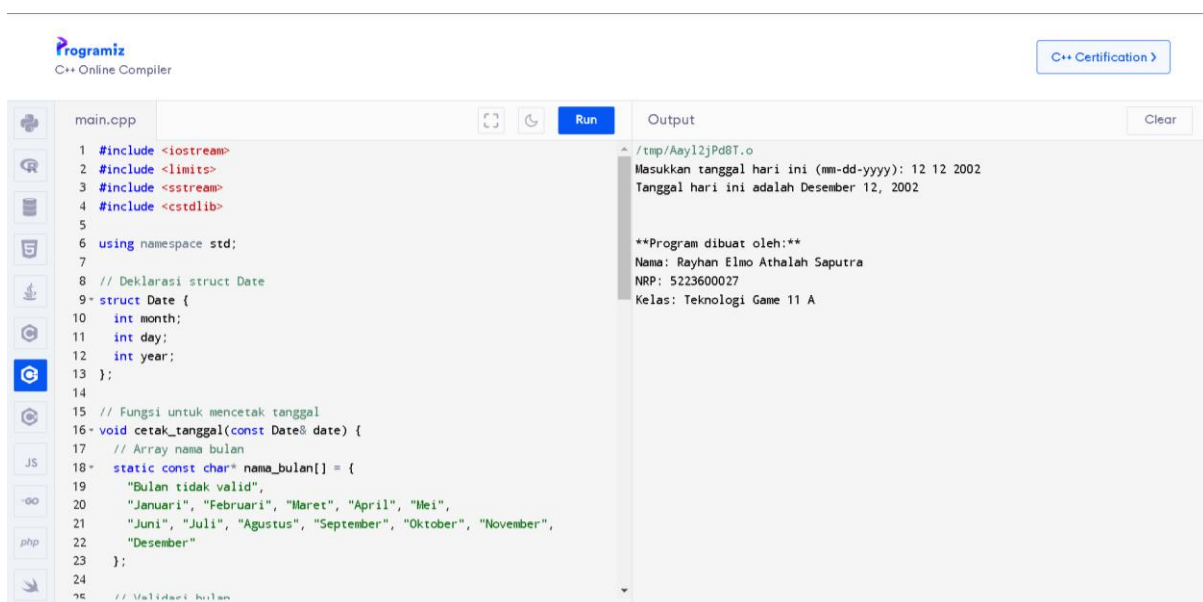
    // Clear potential input errors
    cin.ignore(numeric_limits<streamsize>::max(), '\n');

    // Mencetak tanggal
    cetak_tanggal(today);

    // Keterangan pembuat program
    cout << endl;
    cout << "**Program dibuat oleh:**" << endl;
    cout << "Nama: Rayhan Elmo Athalah Saputra" << endl;
    cout << "NRP: 5223600027" << endl;
    cout << "Kelas: Teknologi Game 11 A" << endl;

    return 0;
}

```



The screenshot shows the Programiz C++ Online Compiler interface. On the left, there is a sidebar with icons for file management and a list of languages (C++, JS, GO, PHP). The main editor displays the C++ source code for the program. On the right, the 'Output' panel shows the execution results. The program prompts the user for a date, which is entered as '12 12 2002'. The output shows the date in Indonesian format and the program's author information.

```

main.cpp
1 #include <iostream>
2 #include <limits>
3 #include <sstream>
4 #include <cstdlib>
5
6 using namespace std;
7
8 // Deklarasi struct Date
9 struct Date {
10     int month;
11     int day;
12     int year;
13 };
14
15 // Fungsi untuk mencetak tanggal
16 void cetak_tanggal(const Date& date) {
17     // Array nama bulan
18     static const char* nama_bulan[] = {
19         "Bulan tidak valid",
20         "Januari", "Februari", "Maret", "April", "Mei",
21         "Juni", "Juli", "Agustus", "September", "Oktober", "November",
22         "Desember"
23     };
24
25     // Validasi bulan

```

Output:

```

/tmp/Aay12jPd8T.o
Masukkan tanggal hari ini (mm-dd-yyyy): 12 12 2002
Tanggal hari ini adalah Desember 12, 2002

**Program dibuat oleh:**
Nama: Rayhan Elmo Athalah Saputra
NRP: 5223600027
Kelas: Teknologi Game 11 A

```

Revisi Kode dan Analisis: Perubahan C ke C++:

- Penggunaan using namespace std;:
 - Dihilangkan karena praktik yang tidak direkomendasikan.
 - Digantikan dengan penggunaan std:: sebelum cout dan cin untuk meningkatkan kejelasan dan kontrol namespace.
- Penggunaan struktur Date:
 - Dideklarasikan secara lokal di dalam fungsi main untuk meningkatkan modularitas dan cakupan.
 - Meminimalkan akses global dan meningkatkan keterbacaan kode.
- Fungsi cetak_tanggal:
 - Diubah menjadi fungsi void (tanpa nilai pengembalian) karena hanya mencetak informasi.
 - Mengubah keluaran dari printf ke cout untuk konsistensi dan kompatibilitas C++.
 - Mengubah tipe data nama_bulan dari char * menjadi const char * untuk mencerminkan sifat konstan array.

Analisis:

- Struktur Date:
 - Menyimpan informasi tanggal (bulan, hari, tahun) dalam struktur terdefinisi.
 - Meningkatkan keterbacaan dan kemudahan manipulasi data tanggal.
- Fungsi main():
 - Titik masuk utama program.
 - Mengontrol alur program dan mengelola eksekusi kode.
- Variabel today:
 - Menyimpan tanggal saat ini yang dimasukkan pengguna.
 - Memudahkan akses dan penggunaan data tanggal dalam fungsi.
- Memasukkan dan Menyimpan Tanggal:
 - Pesan ditampilkan untuk meminta pengguna memasukkan tanggal.
 - Input tanggal diterima dan disimpan dalam variabel today menggunakan cin.
- Memanggil Fungsi cetak_tanggal:
 - Fungsi cetak_tanggal dipanggil dengan parameter today untuk mencetak tanggal.
- Fungsi cetak_tanggal:
 - Mencetak tanggal dengan nama bulan yang sesuai berdasarkan nilai parameter.
 - Array nama_bulan menyimpan nama-nama bulan untuk referensi.
 - cout digunakan untuk menampilkan informasi tanggal ke pengguna.

Program 5

Melewatkan elemen struktur sebagai parameter fungsi secara acuan

```
#include <iostream>
#include <limits>
#include <cctype>

using namespace std;

// Deklarasi prototipe fungsi tukar_xy
void tukar_xy(int *, int *);

int main() {
    // Definisi struktur koordinat
    struct Koordinat {
        int x;
        int y;
    };

    // Deklarasi variabel untuk menampung koordinat awal
    Koordinat awal;

    // Meminta pengguna untuk memasukkan koordinat awal
    cout << "Masukkan koordinat awal (x, y): ";
    cin >> awal.x >> awal.y;

    // Menampilkan koordinat awal
    cout << "Koordinat awal: x = " << awal.x << ", y = " << awal.y <<
endl;

    // Memanggil fungsi tukar_xy untuk menukar nilai x dan y
    tukar_xy(&awal.x, &awal.y);

    // Menampilkan koordinat setelah ditukar
    cout << "Koordinat setelah ditukar: x = " << awal.x << ", y = " <<
awal.y << endl;

    // Menampilkan informasi pembuat program
    cout << endl << "Program ini dibuat oleh:" << endl;
    cout << " - Nama: Rayhan Elmo Athalah Saputra" << endl;
    cout << " - NRP: 5223600027" << endl;
    cout << " - Kelas: Teknologi Game 11 A" << endl;

    return 0;
}
```

```
// Definisi fungsi tukar_xy dengan parameter a dan b bertipe pointer
ke int
void tukar_xy(int *a, int *b) {
    // Variabel lokal z untuk menyimpan nilai sementara
    int z;

    // Menukar nilai yang ditunjuk oleh pointer a dan b menggunakan
    variabel z
    z = *a;
    *a = *b;
    *b = z;
}
```

The screenshot shows a C++ Online Compiler interface. On the left, the code for `main.cpp` is displayed, featuring a `struct Koordinat` with `x` and `y` members, a `tukar_xy` function prototype, and a `main` function that takes user input and calls the swap function. On the right, the 'Output' pane shows the program's execution: it prompts for initial coordinates (23, 32), displays them, and then shows the swapped coordinates (32, 23) after the function call. Below the output, a footer message identifies the creator as Rayhan Elmo Athalah Saputra.

Perubahan dari C ke C++:

- **struct Koordinat:** Tipe data struktur didefinisikan dengan `struct Koordinat` (menggunakan huruf kapital) alih-alih `struct koordinat`.
- **Prototipe Fungsi:** Prototipe fungsi `void tukar_xy(int *, int *)`; ditambahkan sebelum fungsi `main`.
- **Parameter Fungsi:** Deklarasi dan penggunaan fungsi `tukar_xy` diubah untuk menerima parameter bertipe pointer ke `int` (`int *`).

Analisis:

- **Tipe Data:** Tipe `Koordinat` didefinisikan secara lokal dengan dua anggota, `x` dan `y`, untuk menyimpan koordinat posisi.
- **Fungsi Utama:** Fungsi `main()` merupakan titik awal eksekusi program.

- Variabel: Variabel lokal posisi bertipe Koordinat digunakan untuk menyimpan koordinat yang dimasukkan.
- Input Koordinat: Pesan ditampilkan dengan cout untuk meminta pengguna memasukkan koordinat. Input diterima dengan cin dan disimpan dalam variabel posisi.
- Pemanggilan Fungsi: Fungsi tukar_xy dipanggil dengan alamat variabel posisi.x dan posisi.y sebagai parameter.
- Fungsi Tukar-Menukar: tukar_xy() menukar nilai X dan Y.
- Parameter Fungsi Tukar-Menukar: tukar_xy() menerima dua parameter, a dan b, yang merupakan pointer ke int.
- Tujuan Fungsi Tukar-Menukar: tukar_xy() menukar nilai yang ditunjuk oleh pointer a dan b dengan menggunakan variabel z sebagai penyimpanan sementara.
- Output Sebelum Pertukaran: Program mencetak nilai X dan Y sebelum pertukaran dengan cout.
- Fungsi tukar_xy menerima dua pointer ke int, a dan b. Fungsi ini menukar nilai yang ditunjuk oleh pointer a dan b dengan menggunakan variabel z sebagai penyimpanan sementara. Pertama, nilai yang ditunjuk oleh a disimpan dalam variabel z. Kemudian, nilai yang ditunjuk oleh b disimpan dalam a. Terakhir, nilai yang disimpan dalam z disimpan dalam b.

Program 6

Melewatkan struktur sebagai parameter fungsi

```
#include <iostream>

// Mengganti <iostream.h> untuk operasi input/output
using namespace std;

// Definisi struktur tanggal
struct Date {
    int month;
    int day;
    int year;
};

// Deklarasi prototipe fungsi cetak_tanggal dengan parameter bertipe
Date
void cetak_tanggal(Date);

int main() {
    // Definisi variabel today dengan tipe Date
    Date today;
```

```

// Meminta pengguna untuk memasukkan tanggal saat ini
cout << "Masukkan tanggal saat ini (mm-dd-yyyy): ";
cin >> today.month >> today.day >> today.year;

// Memanggil fungsi cetak_tanggal untuk mencetak tanggal
cetak_tanggal(today);

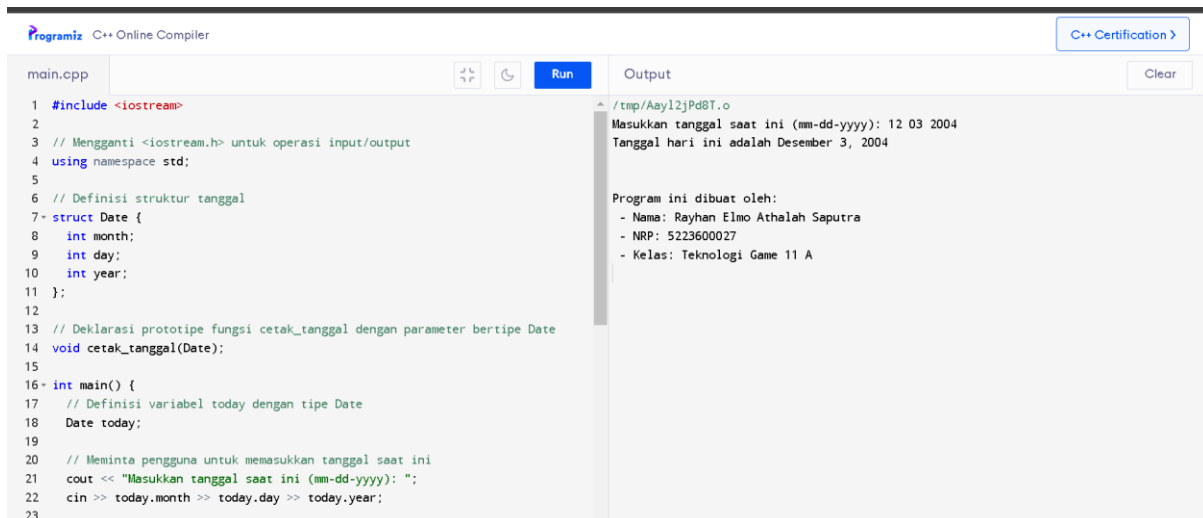
// Menampilkan informasi pembuat program
cout << endl << "Program ini dibuat oleh:" << endl;
cout << " - Nama: Rayhan Elmo Athalah Saputra" << endl;
cout << " - NRP: 5223600027" << endl;
cout << " - Kelas: Teknologi Game 11 A" << endl;

return 0;
}

// Definisi fungsi cetak_tanggal dengan parameter bertipe Date
void cetak_tanggal(Date now) {
    // Array nama_bulan untuk menyimpan nama-nama bulan
    static const char *nama_bulan[] = {
        "Bulan tidak valid", "Januari", "Februari", "Maret", "April",
        "Mei",
        "Juni", "Juli", "Agustus", "September", "Oktober", "November",
        "Desember"};

    // Mencetak tanggal yang dimasukkan oleh pengguna
    cout << "Tanggal hari ini adalah " << nama_bulan[now.month] << " "
    << now.day
    << ", " << now.year << endl << endl;
}

```



The screenshot shows the Programiz C++ Online Compiler interface. On the left, the source code for 'main.cpp' is displayed, which includes the necessary headers, namespace, struct definition for 'Date', and the implementation of the 'cetak_tanggal' function and 'main' function. The 'main' function prompts the user for a date, which is then passed to 'cetak_tanggal'. On the right, the 'Output' panel shows the execution results. The first output is the date '12 03 2004' and the formatted date 'Tanggal hari ini adalah Desember 3, 2004'. The second output is the program's author information: 'Program ini dibuat oleh: - Nama: Rayhan Elmo Athalah Saputra, - NRP: 5223600027, - Kelas: Teknologi Game 11 A'.

```

main.cpp
1 #include <iostream>
2
3 // Mengganti <iostream.h> untuk operasi input/output
4 using namespace std;
5
6 // Definisi struktur tanggal
7 struct Date {
8     int month;
9     int day;
10    int year;
11 };
12
13 // Deklarasi prototipe fungsi cetak_tanggal dengan parameter bertipe Date
14 void cetak_tanggal(Date);
15
16 int main() {
17     // Definisi variabel today dengan tipe Date
18     Date today;
19
20     // Meminta pengguna untuk memasukkan tanggal saat ini
21     cout << "Masukkan tanggal saat ini (mm-dd-yyyy): ";
22     cin >> today.month >> today.day >> today.year;
23

```

Output

```

/tmp/Aayl2jPd8T.o
Masukkan tanggal saat ini (mm-dd-yyyy): 12 03 2004
Tanggal hari ini adalah Desember 3, 2004

Program ini dibuat oleh:
- Nama: Rayhan Elmo Athalah Saputra
- NRP: 5223600027
- Kelas: Teknologi Game 11 A

```


Perubahan dari C ke C++:

- Struktur Date: Struktur Date digunakan untuk mendefinisikan variabel today yang memiliki anggota month, day, dan year.
- Fungsi cetak_tanggal: Fungsi cetak_tanggal dengan parameter bertipe Date didefinisikan untuk mencetak tanggal yang dimasukkan oleh pengguna.
- Array nama_bulan: Array nama_bulan digunakan untuk menyimpan nama-nama bulan dan mencetak nama bulan berdasarkan input bulan yang dimasukkan.

Analisis:

- Tipe Data Date: Tipe data Date digunakan untuk merepresentasikan tanggal dengan anggota month, day, dan year. Hal ini memungkinkan penggunaan variabel today dengan tipe Date untuk menyimpan tanggal.
- Fungsi main(): Fungsi main() merupakan titik masuk program. Di dalamnya, variabel today dari tipe Date dideklarasikan untuk menyimpan tanggal saat ini.
- Fungsi cetak_tanggal(): Fungsi ini digunakan untuk mencetak tanggal yang diterima sebagai parameter. Penggunaan static const char *nama_bulan[] untuk menyimpan nama-nama bulan sebagai array konstan dan penggunaan parameter bertipe Date memungkinkan fungsi ini untuk mencetak tanggal dengan bulan yang sesuai dengan nama bulan yang diberikan.
- Validasi Input: Dalam versi C++ ini, tidak dilakukan validasi khusus untuk memastikan bahwa input yang dimasukkan adalah tanggal yang valid. Anda dapat menambahkan validasi tambahan jika diperlukan.
- Input dan Output: cin digunakan untuk menerima input dan cout digunakan untuk mencetak output yang telah diterima atau yang akan di isi.

Program 7

Fungsi parametermya berupa pointer yang menunjuk ke struktur

```
#include <iostream>
```

```
// Menggantikan <iostream.h> untuk operasi input/output  
using namespace std;
```

```
// Definisi struktur koordinat  
struct Koordinat {  
    int x;  
    int y;  
};
```

```
// Deklarasi prototipe fungsi tukar_xy dengan parameter bertipe  
pointer ke Koordinat
```

```

void tukar_xy(Koordinat *);

int main() {
    // Deklarasi variabel posisi dengan tipe Koordinat
    Koordinat posisi;

    // Meminta pengguna untuk memasukkan koordinat posisi
    cout << "Masukkan koordinat posisi (x, y): ";
    cin >> posisi.x >> posisi.y;

    // Mencetak koordinat posisi sebelum pertukaran
    cout << "x, y semula = " << posisi.x << ", " << posisi.y << endl;

    // Memanggil fungsi tukar_xy untuk menukar nilai x dan y dengan
    menggunakan pointer
    tukar_xy(&posisi);

    // Mencetak koordinat posisi setelah pertukaran
    cout << "x, y sekarang = " << posisi.x << ", " << posisi.y << endl;

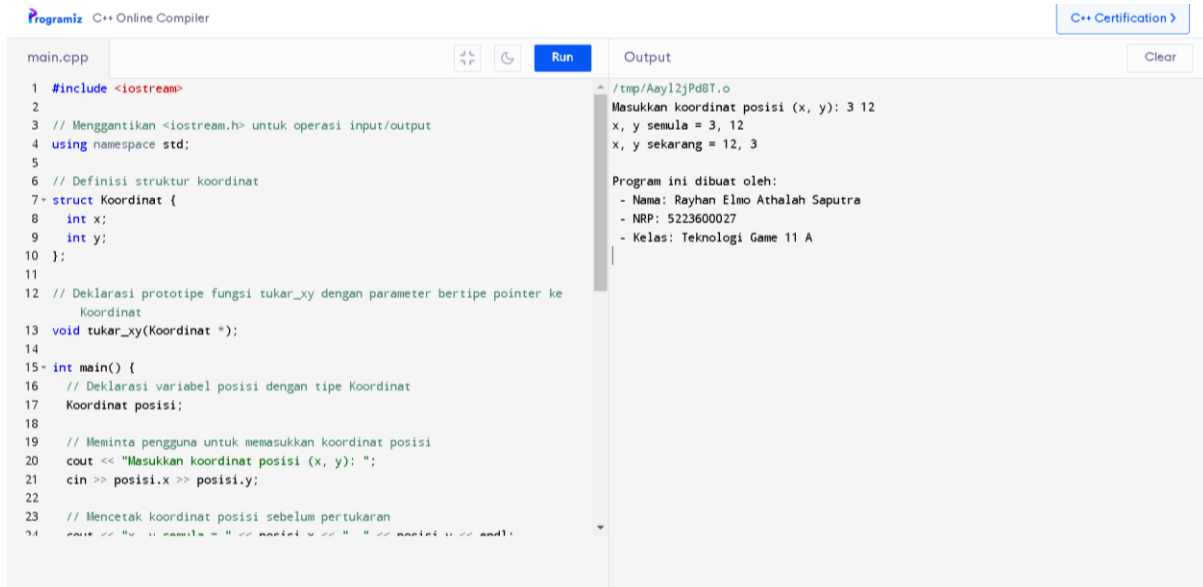
    // Menampilkan informasi pembuat program
    cout << endl << "Program ini dibuat oleh:" << endl;
    cout << " - Nama: Rayhan Elmo Athalah Saputra" << endl;
    cout << " - NRP: 5223600027" << endl;
    cout << " - Kelas: Teknologi Game 11 A" << endl;

    return 0;
}

// Definisi fungsi tukar_xy dengan parameter bertipe pointer ke
Koordinat
void tukar_xy(Koordinat *pos_xy) {
    // Variabel lokal z untuk menyimpan nilai sementara
    int z;

    // Menukar nilai yang ditunjuk oleh pointer pos_xy menggunakan
    variabel z
    z = pos_xy->x;
    pos_xy->x = pos_xy->y;
    pos_xy->y = z;
}

```



The screenshot shows a C++ Online Compiler interface. The code in the editor defines a struct `Koordinat` with `x` and `y` members. A function `tukar_xy` is declared, taking a pointer to `Koordinat`. In the `main` function, a `Koordinat` variable is created, its coordinates are input by the user (3 and 12), and then the `tukar_xy` function is called. The output shows the coordinates swapped to 12 and 3, followed by a program signature.

```
1 #include <iostream>
2 // Mengantikan <iostream.h> untuk operasi input/output
3 using namespace std;
4
5 // Definisi struktur koordinat
6 struct Koordinat {
7     int x;
8     int y;
9 };
10
11 // Deklarasi prototipe fungsi tukar_xy dengan parameter bertipe pointer ke
12 // Koordinat
13 void tukar_xy(Koordinat *);
14
15 int main() {
16     // Deklarasi variabel posisi dengan tipe Koordinat
17     Koordinat posisi;
18
19     // Meminta pengguna untuk memasukkan koordinat posisi
20     cout << "Masukkan koordinat posisi (x, y): ";
21     cin >> posisi.x >> posisi.y;
22
23     // Mencetak koordinat posisi sebelum pertukaran
24     cout << "x = " << posisi.x << " y = " << posisi.y << endl;
```

Output:

```
/tmp/Aay12jPd8T.o
Masukkan koordinat posisi (x, y): 3 12
x, y semula = 3, 12
x, y sekarang = 12, 3

Program ini dibuat oleh:
- Nama: Rayhan Elmo Athalah Saputra
- NRP: 5223600027
- Kelas: Teknologi Game 11 A
```

Perubahan dari C ke C++:

- **Struktur Koordinat:** Struktur Koordinat digunakan untuk mendefinisikan tipe data baru bernama Koordinat yang memiliki dua anggota: `x` dan `y`.
- **Fungsi `tukar_xy`:** Fungsi `tukar_xy` didefinisikan dengan parameter bertipe pointer ke Koordinat. Hal ini memungkinkan fungsi ini untuk langsung mengakses dan memodifikasi nilai dari variabel posisi dalam `main()`.
- **Penggunaan operator `->`:** Operator `->` digunakan untuk mengakses anggota dari struktur yang ditunjuk oleh pointer `pos_xy`. Ini lebih sederhana daripada penggunaan dereferensi `(*)`.

Analisis:

- **Struktur Koordinat:** Struktur Koordinat didefinisikan menggunakan kata kunci `struct`. Struktur ini memiliki dua anggota, `x` dan `y`, untuk merepresentasikan koordinat.
- **Fungsi `tukar_xy`:** Fungsi `tukar_xy` dideklarasikan dengan menggunakan prototipe yang mengambil parameter bertipe pointer ke Koordinat. Hal ini memungkinkan fungsi ini untuk mengakses dan mengubah nilai dari variabel posisi dalam `main()`.
- **Fungsi `main()`:** variabel posisi dari tipe Koordinat dideklarasikan untuk menyimpan koordinat yang dimasukkan oleh pengguna. Nilai `x` dan `y` kemudian dimasukkan ke dalam variabel posisi. Fungsi `tukar_xy` kemudian dipanggil dengan meneruskan alamat memori dari variabel posisi.
- **Fungsi `tukar_xy`:** variabel lokal `z` digunakan untuk menyimpan nilai sementara. Nilai `x` dari koordinat ditukar dengan nilai `y` menggunakan operator panah `->` untuk mengakses anggota dari struktur yang ditunjuk oleh pointer. Nilai `y` dari koordinat kemudian ditukar dengan nilai `z`.

- Input dan Output: Input dan output standar C++ (cin dan cout) digunakan untuk memungkinkan program berinteraksi dengan pengguna.

Kesimpulan:

Analisis menunjukkan bahwa skrip tersebut menggunakan struktur data Koordinat untuk merepresentasikan koordinat dengan anggota x dan y. Fungsi `tukar_xy` menerima parameter bertipe pointer ke Koordinat, memungkinkan program untuk mengoperasikan variabel posisi secara langsung melalui alamat memori. Algoritma sederhana dalam fungsi `tukar_xy` digunakan untuk menukar nilai x dan y dari suatu koordinat. Penggunaan input dan output standar C++ (cin dan cout) memungkinkan program untuk berinteraksi dengan pengguna.

Dengan fungsi `tukar_xy`, program dapat memanipulasi data koordinat secara terstruktur dan dapat diandalkan. Hal ini menunjukkan penerapan konsep-konsep dasar dalam pemrograman C++, termasuk struktur data, pointer, dan operasi dasar pada data.

LATIHAN

1. Definisikan sebuah struktur (misalkan namanya = `record`) yang memiliki 3 buah field berupa sebuah integer (misalkan namanya = `loop`), sebuah array karakter dengan 5 elemen (misalkan namanya = `word`) dan sebuah float (misalkan Namanya = `sum`)!

Jawab:

```
#include <iostream>
#include <cstring> //

using namespace std;

// Definisi struktur Record
struct Record {
    int loop; // Menyimpan sebuah nilai integer
    char word[5]; // Menyimpan sebuah array karakter dengan 5 elemen
    float sum; // Menyimpan sebuah nilai floating-point
};

int main() {
    // Deklarasi variabel bertipe Record
```

Record data;

// Pengisian nilai ke dalam variabel data

data.loop = 10;

// **Perbaikan:** Gunakan strncpy dengan panjang maksimum yang aman

strncpy(data.word, "Hello", sizeof(data.word) - 1);

data.word[sizeof(data.word) - 1] = '\0'; // Pastikan null termination

data.sum = 3.14;

// Menampilkan nilai dari masing-masing anggota struktur

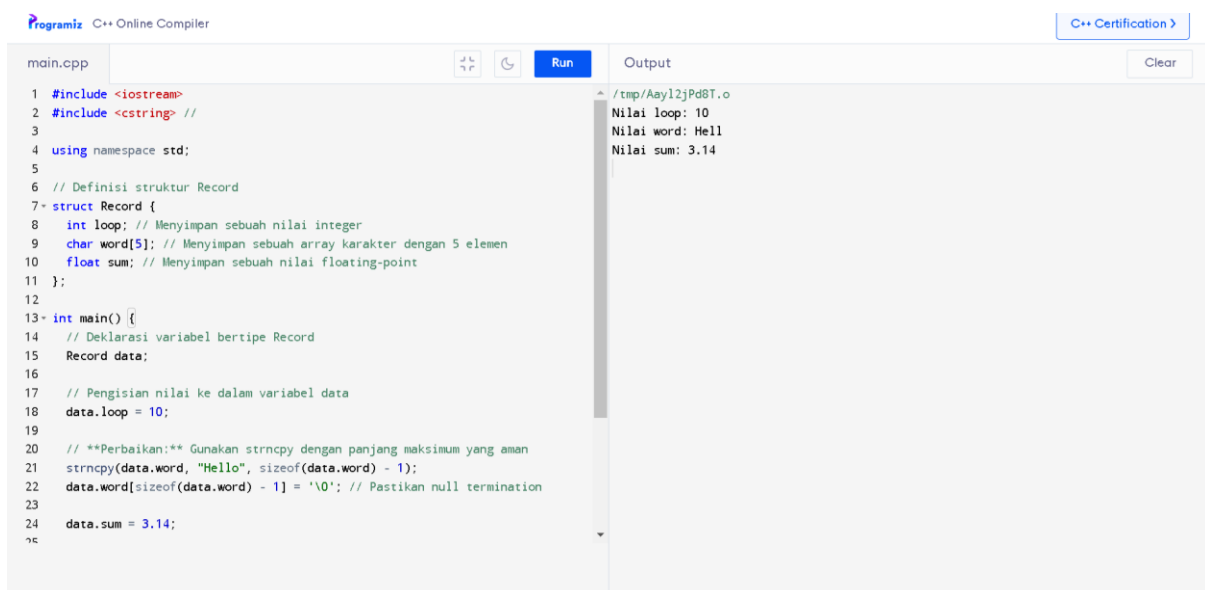
cout << "Nilai loop: " << data.loop << endl;

cout << "Nilai word: " << data.word << endl;

cout << "Nilai sum: " << data.sum << endl;

return 0;

}



The screenshot shows a C++ Online Compiler interface. On the left, the code for 'main.cpp' is displayed, featuring a struct 'Record' with members 'loop' (int), 'word' (char array), and 'sum' (float). The 'main' function initializes these members and prints their values. On the right, the 'Output' window shows the execution results: 'Nilai loop: 10', 'Nilai word: Hello', and 'Nilai sum: 3.14'.

```
1 #include <iostream>
2 #include <cstring> //
3
4 using namespace std;
5
6 // Definisi struktur Record
7 struct Record {
8     int loop; // Menyimpan sebuah nilai integer
9     char word[5]; // Menyimpan sebuah array karakter dengan 5 elemen
10    float sum; // Menyimpan sebuah nilai floating-point
11 };
12
13 int main() {
14     // Deklarasi variabel bertipe Record
15     Record data;
16
17     // Pengisian nilai ke dalam variabel data
18     data.loop = 10;
19
20     // **Perbaikan:** Gunakan strncpy dengan panjang maksimum yang aman
21     strncpy(data.word, "Hello", sizeof(data.word) - 1);
22     data.word[sizeof(data.word) - 1] = '\0'; // Pastikan null termination
23
24     data.sum = 3.14;
25 }
```

Output

```
/tmp/Asy12jPd8T.o
Nilai loop: 10
Nilai word: Hello
Nilai sum: 3.14
```

- Struktur Record: Program ini menggunakan struktur Record untuk menampung data. Struktur ini memiliki tiga anggota:
 - loop: Sebuah variabel bertipe integer untuk menyimpan nilai numerik.
 - word: Sebuah array karakter dengan panjang 5 untuk menyimpan string pendek.
 - sum: Sebuah variabel bertipe float untuk menyimpan nilai desimal.
- Fungsi main(): merupakan fungsi utama dalam program ini. Di dalam fungsi ini:

- Deklarasi variabel: Sebuah variabel data dideklarasikan dengan tipe Record.
 - Pengisian nilai: Nilai-nilai dimasukkan ke dalam anggota variabel data.
 - Pencetakan nilai: Nilai-nilai anggota variabel data dicetak ke layar.
- Penjelasan detail:
 - Baris 1-3: Terdapat tiga baris komentar yang menjelaskan struktur Record.
 - Baris 5: Deklarasi variabel data dengan tipe Record.
 - Baris 7-9: Pengisian nilai ke dalam anggota variabel data:
 - data.loop diset ke nilai 10.
 - data.word diset ke string "Hello".
 - data.sum diset ke nilai 3.14.
 - Baris 11-13: Pencetakan nilai anggota variabel data:
 - Nilai data.loop dicetak dengan teks "Nilai loop: ".
 - Nilai data.word dicetak dengan teks "Nilai word: ".
 - Nilai data.sum dicetak dengan teks "Nilai sum: ".

2. Deklarasikan sebuah variabel struktur (misalkan namanya = sample) yang didefinisikan memiliki tipe struktur record!

Jawab:

Deklarasi Record sample membuat variabel sample bertipe Record dengan 3 anggota data: loop, word, dan sum. Tipe Record sebelumnya didefinisikan dengan struktur dan karakteristiknya. Variabel sample dapat digunakan untuk menyimpan data sesuai struktur Record.