

LAPORAN PRAKTIKUM STRUKTUR DATA DAN ALGORITMA

MODUL 1

Tipe data



DISUSUN OLEH:

PRIESTY AMEILIANA MAULIDAH

2311102175

S1 IF-11-E

DOSEN:

Muhammad Afrizal Amrustian, S. Kom.

PROGRAM STUDI S1 TEKNIK INFORMATIKA

FAKULTAS INFORMATIKA

INSTITUT TEKNOLOGI TELKOM PURWOKERTO PURWOKERTO

2024

A.DASAR TEORI

Tipe data adalah pengklasifikasian data berdasarkan jenisnya. Tipe data dibutuhkan agar kompiler dapat mengetahui penggunaan data tersebut. Ada tiga jenis tipe data yang akan dipelajari: tipe data primitif, tipe data abstrak, dan tipe data koleksi.

Tipe data primitif sudah ditentukan oleh sistem dan disediakan oleh banyak bahasa pemrograman. Contohnya adalah tipe data `int` untuk bilangan bulat, `float` untuk bilangan desimal, `char` untuk huruf, dan `boolean` untuk nilai boolean.

Tipe data abstrak atau ADT dibentuk oleh programmer. Pada tipe data ini, bisa terdapat banyak jenis data dan nilainya bisa beragam. Fitur `class` pada bahasa C++ mirip dengan fitur data structures `struct` pada bahasa C.

Tipe data koleksi digunakan untuk mengelompokkan dan menyimpan beberapa nilai atau objek secara bersamaan. Contohnya adalah `array`, `vector`, dan `map`. `Array` adalah struktur data statis, `vector` adalah STL dengan alokasi memorinya otomatis, dan `map` menggunakan self-balancing tree seperti red-black tree.

B.Guided

Guided 1

```
// priesty ameiliana maulidah
// 2311102175

#include <iostream>
using namespace std;
// Main program
int main()
{
    char op;
    float num1, num2;
    // It allows user to enter operator i.e. +, -, *, /
    cin >> op;
    // It allow user to enter the operands
    cin >> num1 >> num2;
    // Switch statement begins
    switch (op)
    {
        // If user enter +
        case '+':
            cout << num1 + num2;
            break;
        // If user enter -
        case '-':
            cout << num1 - num2;
            break;
```

```
// If user enter *  
case '*':  
    cout << num1 * num2;  
    break;  
// If user enter /  
case '/':  
    cout << num1 / num2;  
    break;  
// If the operator is other than +, -, * or /,  
// error message will display  
default:  
    cout << "Error! operator is not correct";  
} // switch statement ends  
return 0;  
}
```

Screenshots output

Deskripsi :

➤ **Deklarasi Variabel:**

- char op: Menyimpan operator aritmatika yang dimasukkan oleh pengguna.
- float num1, num2: Menyimpan dua angka operand yang dimasukkan oleh pengguna.

➤ **Input dari Pengguna:**

- cin >> op;: Menerima input operator dari pengguna.
- cin >> num1 >> num2;: Menerima dua angka operand dari pengguna.

➤ **Switch Statement:**

- Berdasarkan nilai op, program akan melakukan operasi yang sesuai:
 - Jika op adalah +, maka program akan menjumlahkan num1 dan num2.
 - Jika op adalah -, maka program akan mengurangi num1 dengan num2.
 - Jika op adalah *, maka program akan mengalikan num1 dengan num2.

- Jika op adalah /, maka program akan membagi num1 dengan num2.
- Jika op bukan salah satu dari +, -, *, atau /, maka program akan menampilkan pesan kesalahan "Error! operator is not correct".

➤ **Output:**

- Program akan menampilkan hasil dari operasi yang sesuai atau pesan kesalahan jika operator tidak valid.

Guided 2

```
// priesty ameiliana maulidah
// 2311102175

#include <stdio.h>

//Struct
struct Mahasiswa
{
    const char *name;
    const char *address;
    int age;
};

int main()
{
    // menggunakan struct
    struct Mahasiswa mhs1, mhs2;
    // mengisi nilai ke struct
    mhs1.name = "Dian";
    mhs1.address = "Mataram";
    mhs1.age = 22;
    mhs2.name = "Bambang";
    mhs2.address = "Surabaya";
    mhs2.age = 23;
```

```
// mencetak isi struct

printf("## Mahasiswa 1 ##\n");

printf("Nama: %s\n", mhs1.name);

printf("Alamat: %s\n", mhs1.address);

printf("Umur: %d\n", mhs1.age);

printf("## Mahasiswa 2 ##\n");

printf("Nama: %s\n", mhs2.name);

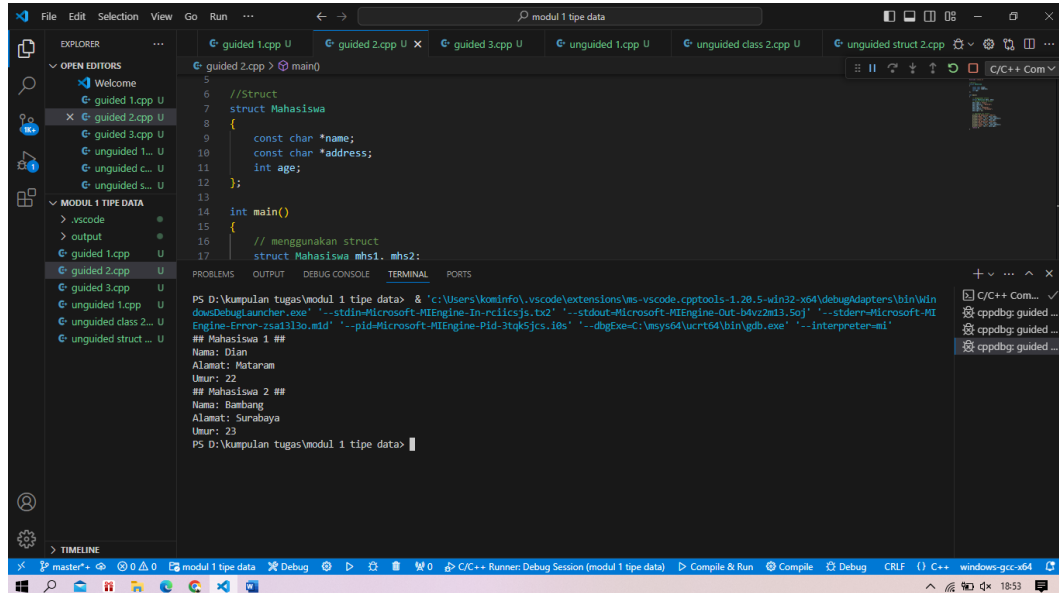
printf("Alamat: %s\n", mhs2.address);

printf("Umur: %d\n", mhs2.age);

return 0;

}
```


Screenshots output



```
5 // Struct
6 struct Mahasiswa
7 {
8     const char *name;
9     const char *address;
10    int age;
11 };
12
13
14 int main()
15 {
16     // menggunakan struct
17     struct Mahasiswa mhs1, mhs2;
```

```
PS D:\kumpulan tugas\modul 1 tipe data> & 'C:\Users\kominfo\.vscode\extensions\ms-vscode.cpptools-1.28.5-win32-x64\debugAdapters\bin\Win
dowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-rclicsjs.tx2' '--stdout=Microsoft-MIEngine-Out-b4vz2m13.5oJ' '--stderr=Microsoft-MI
Engine-Error-zsa13l3o.mid' '--pid=Microsoft-MIEngine-Pid-3tq5jcs.i0s' '--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
## Mahasiswa 1 ##
Nama: Dian
Alamat: Mataram
Umur: 22
## Mahasiswa 2 ##
Nama: Bambang
Alamat: Surabaya
Umur: 23
PS D:\kumpulan tugas\modul 1 tipe data> |
```

Deskripsi :

➤ Pendefinisian Struct:

- struct Mahasiswa: Didefinisikan dengan tiga anggota:
 - const char *name: Menyimpan nama mahasiswa sebagai pointer ke string.
 - const char *address: Menyimpan alamat mahasiswa sebagai pointer ke string.
 - int age: Menyimpan umur mahasiswa sebagai integer.

➤ Deklarasi dan Inisialisasi Variabel Struct:

- `struct Mahasiswa mhs1, mhs2;` Mendeklarasikan dua variabel `mhs1` dan `mhs2` dari tipe `struct Mahasiswa`.
- Mengisi nilai-nilai untuk anggota `mhs1` dan `mhs2`:

- `mhs1`:

- `name = "Dian"`
- `address = "Mataram"`
- `age = 22`

- `mhs2`:

- `name = "Bambang"`
- `address = "Surabaya"`
- `age = 23`

➤ **Menampilkan Isi Struct:**

- Menggunakan `printf` untuk mencetak nilai anggota dari `mhs1` dan `mhs2`.

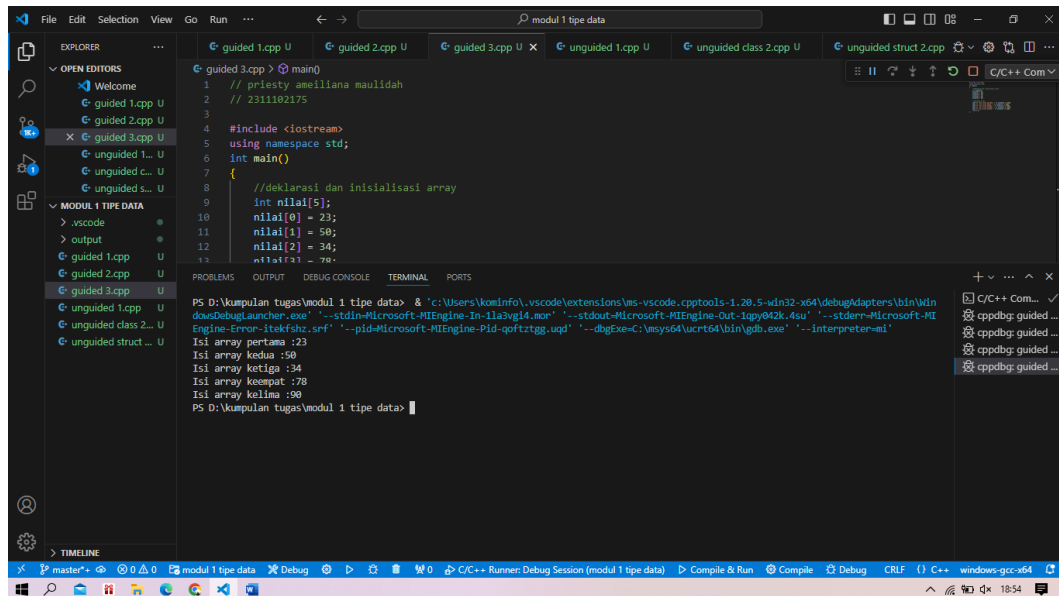
Guided 3

```
// priesty ameiliana maulidah
// 2311102175

#include <iostream>
using namespace std;
int main()
{
    //deklarasi dan inisialisasi array
    int nilai[5];
    nilai[0] = 23;
    nilai[1] = 50;
    nilai[2] = 34;
    nilai[3] = 78;
    nilai[4] = 90;

    //mencetak array
    cout << "Isi array pertama :" << nilai[0] << endl;
    cout << "Isi array kedua :" << nilai[1] << endl;
    cout << "Isi array ketiga :" << nilai[2] << endl;
    cout << "Isi array keempat :" << nilai[3] << endl;
    cout << "Isi array kelima :" << nilai[4] << endl;
    return 0;
}
```

Screenshots output



The screenshot shows the Visual Studio Code interface with a C++ file named 'guided 3.cpp' open. The code defines an array 'nilai' of size 5 and initializes it with values 23, 50, 34, 78, and 90. The output window shows the execution results, including the array values and the program's exit status.

```
1 // priesty amelliana maulidah
2 // 2311102175
3
4 #include <iostream>
5 using namespace std;
6 int main()
7 {
8     //deklarasi dan inisialisasi array
9     int nilai[5];
10    nilai[0] = 23;
11    nilai[1] = 50;
12    nilai[2] = 34;
13    nilai[3] = 78;
14    nilai[4] = 90;
15 }
```

Output:

```
PS D:\kumpulan tugas\modul 1 tipe data> & 'C:\Users\kominfo\.vscode\extensions\ms-vscode.cpptools-1.20.5-win32-x64\debugAdapters\bin\Win
dowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-11a3vg14.moe' '--stdout=Microsoft-MIEngine-Out-1qpy042k.4su' '--stderr=Microsoft-MI
Engine-Error-itekfshz.srf' '--pid=Microsoft-MIEngine-Pid-qoftzgg.ugd' '--dbgExe=C:\msys64\usr\bin\gdb.exe' '--interpreter=mi'
Isi array pertama :23
Isi array kedua :50
Isi array ketiga :34
Isi array keempat :78
Isi array kelima :90
PS D:\kumpulan tugas\modul 1 tipe data>
```

Deskripsi :

➤ Deklarasi dan Inisialisasi Array:

- `int nilai[5];`: Mendeklarasikan sebuah array nilai yang dapat menyimpan 5 elemen integer.
- `nilai[0] = 23;`: Menginisialisasi elemen pertama array dengan nilai 23.
- `nilai[1] = 50;`: Menginisialisasi elemen kedua array dengan nilai 50.
- `nilai[2] = 34;`: Menginisialisasi elemen ketiga array dengan nilai 34.
- `nilai[3] = 78;`: Menginisialisasi elemen keempat array dengan nilai 78.

- `nilai[4] = 90;` Menginisialisasi elemen kelima array dengan nilai 90.

➤ **Mencetak Isi Array:**

- Program menggunakan `cout` untuk mencetak nilai dari masing-masing elemen array.
- `cout << "Isi array pertama :" << nilai[0] << endl;`
Mencetak elemen pertama array.
- `cout << "Isi array kedua :" << nilai[1] << endl;`
Mencetak elemen kedua array.
- `cout << "Isi array ketiga :" << nilai[2] << endl;`
Mencetak elemen ketiga array.
- `cout << "Isi array keempat :" << nilai[3] << endl;`
Mencetak elemen keempat array.
- `cout << "Isi array kelima :" << nilai[4] << endl;`
Mencetak elemen kelima array.

c. unguided/tugas

unguided 1

```
// priesty ameiliana maulidah
// 2311102175

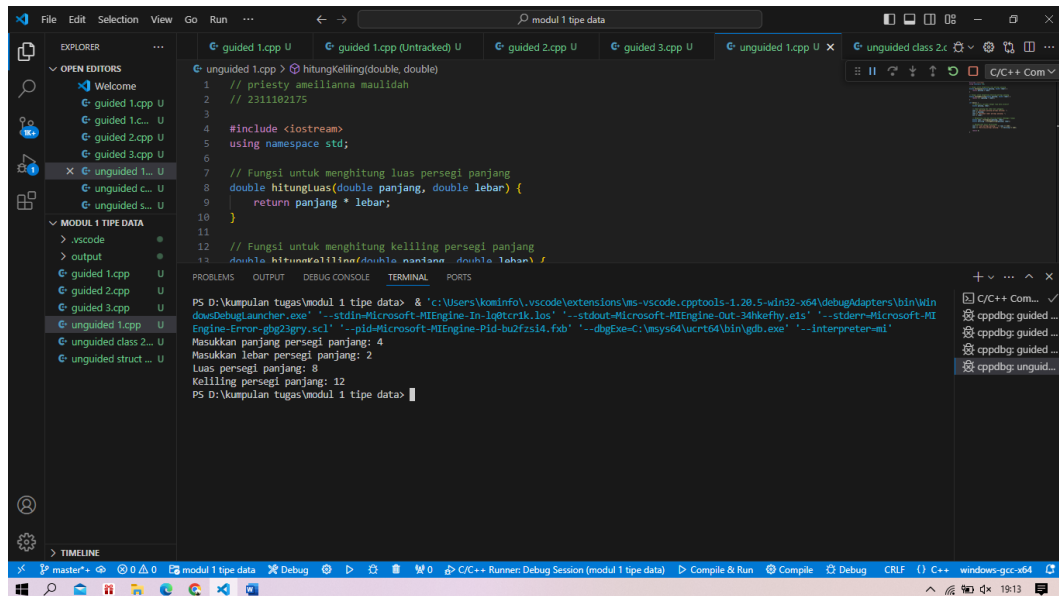
#include <iostream>
using namespace std;

// Fungsi untuk menghitung luas persegi panjang
double hitungLuas(double panjang, double lebar) {
    return panjang * lebar;
}

// Fungsi untuk menghitung keliling persegi panjang
double hitungKeliling(double panjang, double lebar) {
    return 2 * (panjang + lebar);
}
```

```
int main() {  
  
    // Deklarasi variabel dengan tipe data primitif  
    double panjang, lebar;  
  
    // Input panjang dan lebar dari pengguna  
    cout << "Masukkan panjang persegi panjang: ";  
    cin >> panjang;  
    cout << "Masukkan lebar persegi panjang: ";  
    cin >> lebar;  
  
    // Menghitung luas dan keliling dengan memanggil fungsi  
    double luas = hitungLuas(panjang, lebar);  
    double keliling = hitungKeliling(panjang, lebar);  
  
    // Menampilkan hasil perhitungan  
    cout << "Luas persegi panjang: " << luas << endl;  
    cout << "Keliling persegi panjang: " << keliling << endl;  
  
    return 0;  
}
```

Screenshot output



```
1 // priesty azeilianna maulidah
2 // 2311102175
3
4 #include <iostream>
5 using namespace std;
6
7 // Fungsi untuk menghitung luas persegi panjang
8 double hitungLuas(double panjang, double lebar) {
9     return panjang * lebar;
10 }
11
12 // Fungsi untuk menghitung keliling persegi panjang
13 double hitungKeliling(double panjang, double lebar) {
14     return 2 * (panjang + lebar);
15 }
```

PS D:\kumpulan tugas\modul 1 tipe data> & 'C:\Users\kominfo\.vscode\extensions\ms-vscode.cpptools-1.20.5-win32-x64\debugAdapters\bin\WindowsDebugLauncher.exe' '--stdin-Microsoft-MIEngine-In-lq0trik.los' '--stdout-Microsoft-MIEngine-Out-34hkefhy.eis' '--stderr-Microsoft-MIEngine-Error-gbg23gry.scl' '--pid-Microsoft-MIEngine-Pid-bu2fzsi4.fxb' '--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'

Masukkan panjang persegi panjang: 4
Masukkan lebar persegi panjang: 2
Luas persegi panjang: 8
Keliling persegi panjang: 12
PS D:\kumpulan tugas\modul 1 tipe data> |

Deskripsi :

➤ Deklarasi Fungsi:

- **hitungLuas(double panjang, double lebar):**
Fungsi ini menerima dua parameter bertipe double dan mengembalikan hasil perkalian antara panjang dan lebar, yang merupakan luas persegi panjang.
- **hitungKeliling(double panjang, double lebar):**
Fungsi ini juga menerima dua parameter bertipe double dan mengembalikan hasil dari rumus keliling persegi panjang yaitu $2 * (\text{panjang} + \text{lebar})$.

➤ Fungsi main:

- Deklarasi variabel panjang dan lebar dengan tipe data double.
- Meminta input dari pengguna untuk nilai panjang dan lebar.
- Memanggil fungsi hitungLuas dan hitungKeliling dengan argumen panjang dan lebar, lalu menyimpan hasilnya ke variabel luas dan keliling.
- Menampilkan hasil perhitungan luas dan keliling ke layar.

Unguided 2 (class)

```
// priesty ameiliana maulidah
// 2311102175

#include <iostream>
using namespace std;

class PersegiPanjang {
private:
    double panjang;
    double lebar;

public:
    // Konstruktor
    PersegiPanjang(double p, double l) {
        panjang = p;
        lebar = l;
    }

    // Fungsi untuk menghitung luas
    double hitungLuas() {
        return panjang * lebar;
    }
}
```

```
// Fungsi untuk menghitung keliling
double hitungKeliling() {
    return 2 * (panjang + lebar);
}

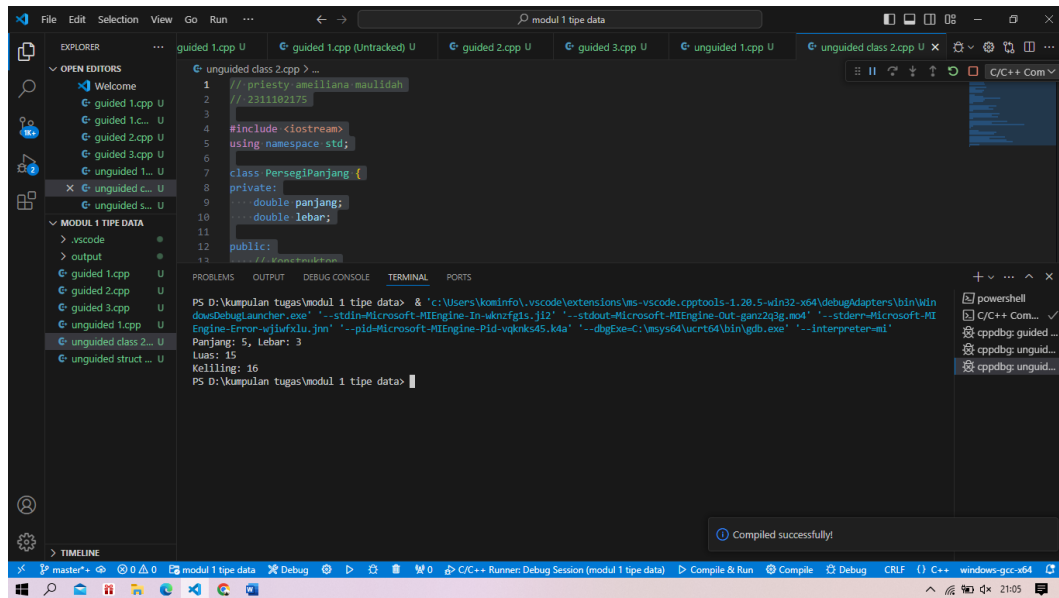
// Fungsi untuk menampilkan panjang dan lebar
void tampilkanDimensi() {
    cout << "Panjang: " << panjang << ", Lebar: " << lebar << endl;
}
};

int main() {
    // Membuat objek PersegiPanjang
    PersegiPanjang pp(5.0, 3.0);

    // Menampilkan dimensi, luas, dan keliling
    pp.tampilkanDimensi();
    cout << "Luas: " << pp.hitungLuas() << endl;
    cout << "Keliling: " << pp.hitungKeliling() << endl;

    return 0;
}
```

Screenshots output:



Deskripsi :

- Baris `#include <iostream>` menyertakan header file `iostream` yang diperlukan untuk operasi input dan output.
- Baris `using namespace std;` menggunakan namespace `std` sehingga nama-nama kelas dan fungsi dari namespace tersebut dapat digunakan tanpa perlu menuliskan `std::` di depannya.
- Kelas `PersegiPanjang` didefinisikan dengan atribut privat `panjang` dan `lebar`, dan beberapa fungsi publik seperti konstruktor, `hitungLuas`, `hitungKeliling`, dan `tampilkanDimensi`.

- Konstruktor `PersegiPanjang` menginisialisasi nilai panjang dan lebar dengan nilai yang diberikan saat objek dibuat.
- Fungsi `hitungLuas` menghitung dan mengembalikan hasil perkalian panjang dan lebar.
- Fungsi `hitungKeliling` menghitung dan mengembalikan hasil penjumlahan dua kali panjang dan dua kali lebar.
- Fungsi `tampilkanDimensi` mencetak nilai panjang dan lebar ke konsol.
- Fungsi utama membuat objek `PersegiPanjang` baru, memanggil fungsi `tampilkanDimensi`, `hitungLuas`, dan `hitungKeliling`, dan kemudian mengembalikan nilai 0.

Unguided 2 (struct)

```
// priesty ameiliana maulidah
// 2311102175

#include <iostream>
using namespace std;

struct PersegiPanjang {
    double panjang;
    double lebar;

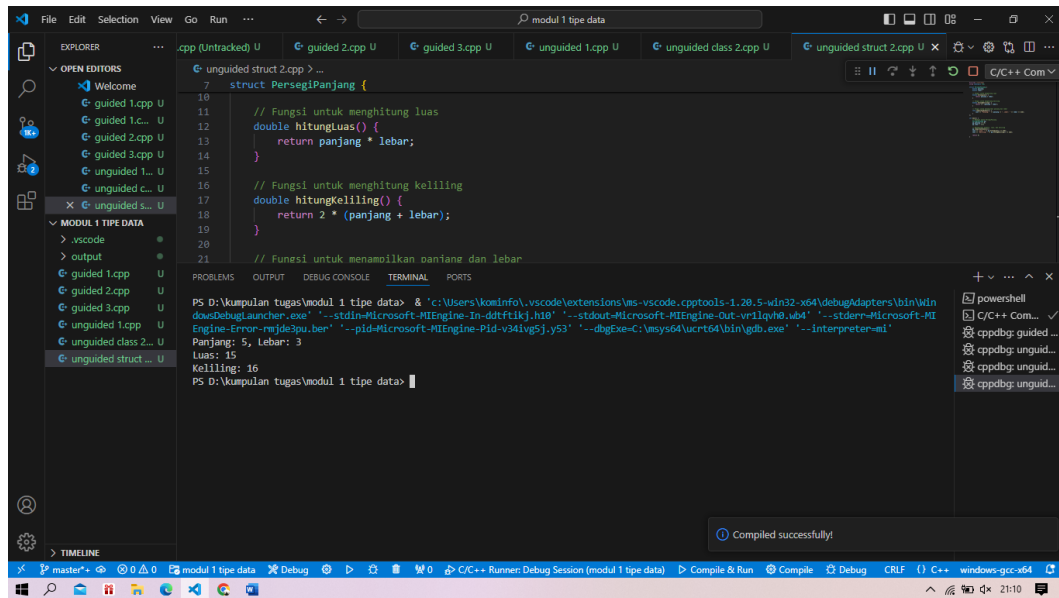
    // Fungsi untuk menghitung luas
    double hitungLuas() {
        return panjang * lebar;
    }

    // Fungsi untuk menghitung keliling
    double hitungKeliling() {
        return 2 * (panjang + lebar);
    }

    // Fungsi untuk menampilkan panjang dan lebar
    void tampilkanDimensi() {
        cout << "Panjang: " << panjang << ", Lebar: " << lebar << endl;
    }
};
```

```
int main() {  
    // Membuat objek PersegiPanjang  
    PersegiPanjang pp;  
    pp.panjang = 5.0;  
    pp.lebar = 3.0;  
  
    // Menampilkan dimensi, luas, dan keliling  
    pp.tampilkanDimensi();  
    cout << "Luas: " << pp.hitungLuas() << endl;  
    cout << "Keliling: " << pp.hitungKeliling() << endl;  
  
    return 0;  
}
```

Screenshot output



```
7 struct PersegiPanjang {
8     int panjang;
9     int lebar;
10
11     // Fungsi untuk menghitung luas
12     double hitungLuas() {
13         return panjang * lebar;
14     }
15
16     // Fungsi untuk menghitung keliling
17     double hitungKeliling() {
18         return 2 * (panjang + lebar);
19     }
20
21     // Fungsi untuk menampilkan panjang dan lebar
22     void tampilkanDimensi() {
23         cout << "Panjang: " << panjang << ", Lebar: " << lebar << endl;
24     }
25 }
```

```
PS D:\kumpulan tugas\modul 1 tipe data> & 'C:\Users\kominfo\.vscode\extensions\ms-vscode.cpptools-1.28.5-win32-x64\debugAdapters\bin\Win
dowsDebugLauncher.exe' '--stdin-Microsoft-MIEngine-in-ddtftlkj.h10' '--stdout-Microsoft-MIEngine-out-vr1lqv8.wb4' '--stderr-Microsoft-MI
Engine-Error-raw3d3pu.be' '--pid-Microsoft-MIEngine-Pid-v341vg5j.y53' '--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
Panjang: 5, Lebar: 3
Luas: 15
Keliling: 16
PS D:\kumpulan tugas\modul 1 tipe data>
```

Deskripsi :

- Baris `#include <iostream>` menyertakan header file `iostream` yang diperlukan untuk operasi input dan output.
- Baris `using namespace std;` menggunakan namespace `std` sehingga nama-nama kelas dan fungsi dari namespace tersebut dapat digunakan tanpa perlu menuliskan `std::` di depannya.
- Struktur data `PersegiPanjang` didefinisikan dengan dua anggota, yaitu `panjang` dan `lebar`, dan beberapa fungsi seperti `hitungLuas`, `hitungKeliling`, dan `tampilkanDimensi`.

- Fungsi `hitungLuas` menghitung dan mengembalikan hasil perkalian panjang dan lebar.
- Fungsi `hitungKeliling` menghitung dan mengembalikan hasil penjumlahan dua kali panjang dan dua kali lebar.
- Fungsi `tampilkanDimensi` mencetak nilai panjang dan lebar ke konsol.
- Fungsi utama membuat variabel struktur data `PersegiPanjang` baru, menginisialisasi nilai panjang dan lebarnya, memanggil fungsi `tampilkanDimensi`, `hitungLuas`, dan `hitungKeliling`, dan kemudian mengembalikan nilai 0.

Unguided 3

```
// priesty ameiliana maulidah
// 2311102175

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

int main() {
    // Deklarasi map dengan kunci bertipe string dan nilai bertipe int
    map<string, int> nilaiMahasiswa;

    // Menambahkan data ke dalam map
    nilaiMahasiswa["yessika"] = 85;
    nilaiMahasiswa["priesty"] = 90;
    nilaiMahasiswa["lala"] = 75;

    // Menampilkan isi map
    cout << "Daftar nilai mahasiswa:" << endl;
    for (const auto& pasangan : nilaiMahasiswa) {
        cout << pasangan.first << ": " << pasangan.second << endl;
    }
}
```

```
// Mencari nilai berdasarkan kunci

string nama;

cout << "Masukkan nama mahasiswa yang ingin dicari nilainya: ";

cin >> nama;

auto it = nilaiMahasiswa.find(nama);

if (it != nilaiMahasiswa.end()) {

    cout << "Nilai " << nama << ": " << it->second << endl;

} else {

    cout << "Mahasiswa dengan nama " << nama << " tidak ditemukan." << endl;

}

return 0;

}
```

E. Referensi

<https://www.domainesia.com/berita/tipe-data/>