

**LAPORAN PRAKTIKUM
STRUKTUR DATA**

**MODUL 1
PENGENALAN BAHASA C++**



Disusun Oleh :

NAMA : M.IRFAN ADIB

NIM : 103112400257

Dosen

FAHRUDIN MUKTI WIBOWO

**PROGRAM STUDI STRUKTUR DATA
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2025**

A. Dasar Teori

Bahasa C++ adalah pengembangan dari bahasa C yang mendukung Pemrograman Berorientasi Objek (OOP) dan bersifat case-sensitive. Struktur dasar program C++ terdiri dari preprocessor directive (seperti `#include <iostream>`), deklarasi `using namespace std;` untuk menggunakan pustaka standar, dan fungsi `int main()` sebagai titik awal eksekusi, yang diakhiri dengan `return 0;`. Data dikelola menggunakan variabel bertipe spesifik (seperti `int`, `float`, `char`, `bool`) dan dimanipulasi oleh operator seperti aritmatika (`+`, `-`, `*`, `/`) dan relasional (`==`, `>`, `<`). Interaksi I/O dilakukan melalui fungsi `cout` dan `cin`. Alur eksekusi program dikendalikan oleh struktur kontrol, termasuk struktur kondisional (`if-else`, `switch-case`) untuk pengambilan keputusan, dan struktur perulangan (`for`, `while`, `do-while`) untuk eksekusi kode berulang, membentuk logika dasar dalam pembangunan aplikasi C++.

B. Guided (berisi screenshot source code & output program disertai penjelasannya)

Guided 1

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

int main() {
    int hari;
    cout << "Hari 1-7 : ";
    cin >> hari;
    if (hari == 7) {
        cout << "Hari Minggu\n";
    } else {
        cout << "Hari Kerja\n";
    }

    switch (hari)
    {
        case 7:
            cout << "Hari Minggu\n";
            break;

        default:
            cout << "Hari Kerja\n";
            break;
    }
}
```

Screenshots Output

```
PS D:\C++\modul1> d:\C++\modul1\guided1
Hari 1-7 : 7
Hari Minggu
Hari Minggu
PS D:\C++\modul1> 
```

Deskripsi:

Program C++ ini adalah program sederhana yang meminta pengguna memasukkan sebuah bilangan bulat antara 1 hingga 7 untuk merepresentasikan hari. Program ini bertujuan untuk mengidentifikasi apakah hari yang dimasukkan adalah "Hari Minggu" (jika inputnya 7) atau "Hari Kerja" (jika inputnya 1-6). Program ini secara unik menggunakan dua blok logika terpisah yaitu pernyataan if-else dan pernyataan switch secara berurutan untuk memproses input dan menghasilkan output yang sama, sehingga hasil akhirnya adalah pesan "Hari Minggu" atau "Hari Kerja" yang ditampilkan sebanyak dua kali.

Guided 2

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

struct mahasiswa
{
    string nama;
    string NIM;
};

int main() {
    mahasiswa mhs;

    cout << "Nama : ";
    getline(cin, mhs.nama);
    cout << "NIM : ";
    cin >> mhs.NIM;

    cout << "Nama : " << mhs.nama << endl << "NIM : " << mhs.NIM;
}
```

Screenshots Output

```
PS D:\Laporan praktikum stuktur data\Modul 1\GUIDED> cd "d:\Laporan praktikum stuktur data\Modul 1\GUIDED\" ; if ($?) { g++ guided2.cpp -o guided2 } ; if ($?) { .\guided2 }
Nama : M.Irfan Adib
NIM : 103112400257
Nama : M.Irfan Adib
NIM : 103112400257
PS D:\Laporan praktikum stuktur data\Modul 1\GUIDED> |
```

Deskripsi:

Kode C++ di atas mendefinisikan sebuah struktur bernama mahasiswa yang berfungsi sebagai *blueprint* untuk mengelompokkan dua data terkait, yaitu nama dan NIM (Nomor Induk Mahasiswa). Di dalam fungsi utama (main), program mendeklarasikan variabel mhs dari tipe mahasiswa, kemudian meminta pengguna memasukkan nama menggunakan fungsi getline() agar dapat membaca spasi, dan NIM menggunakan cin. Setelah data disimpan ke dalam anggota mhs.nama dan mhs.NIM, program diakhiri dengan mencetak kembali kedua informasi tersebut ke konsol, menampilkan data yang telah berhasil diinput dan disimpan dalam struktur.

Guided 3

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

struct Mahasiswa
{
    string nama;
    string NIM;
};

int main() {
    Mahasiswa mhs;
    mhs.nama = "M.Irfan Adib";
    mhs.NIM = "103112400257";

    cout << "Nama : " << mhs.nama << endl << "NIM : " << mhs.NIM;
}
```

Screenshots Output

```
PS D:\Laporan praktikum stuktur data\Modul 1\GUIDED> cd "d:\Laporan praktikum stuktur data\Modul 1\GUIDED\" ; if ($?) { g++ guided3.cpp -o guided3 } ; if ($?) { .\guided3 }
Nama : M.Irfan Adib
NIM : 103112400257
PS D:\Laporan praktikum stuktur data\Modul 1\GUIDED>
```

Deskripsi:

Kode C++ ini berfungsi untuk mendemonstrasikan penggunaan struktur (struct) yang diberi nama Mahasiswa untuk mengelompokkan variabel nama dan NIM di bawah satu entitas. Di dalam fungsi utama, program mendeklarasikan variabel mhs dari tipe Mahasiswa, lalu secara langsung (hardcoded) memberikan nilai *string* "M.Irfan Adib" ke mhs.nama dan "103112400257" ke mhs.NIM. Akhirnya, program menggunakan cout untuk mencetak kedua nilai data anggota struktur yang telah tersimpan tersebut, menunjukkan data yang berhasil diorganisir dan diakses melalui variabel struktur tunggal.

Guided 4

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

int main() {

    for (int i = 0; i < 10; i++)
    {
        cout << "M.Irfan Adib";
        cout << endl;
    }
    cout << endl;

    int i = 0;
    while (i < 10) {
        cout << "103112400257";
        i++;
        cout << endl;
    }
    cout << endl;

    int j = 0;
    do
```

```
{  
    cout << j;  
} while (j != 0);  
}
```

Screenshots Output

```
PS D:\Laporan praktikum stuktur data\Modul 1\GUIDED> cd "d:\Laporan praktikum stuktur  
data\Modul 1\GUIDED\" ; if ($?) { g++ guided4.cpp -o guided4 } ; if ($?) { .\guided4  
}  
M.Irfan Adib  
M.Irfan Adib  
M.Irfan Adib  
M.Irfan Adib  
M.Irfan Adib  
M.Irfan Adib  
M.Irfan Adib  
M.Irfan Adib  
M.Irfan Adib  
M.Irfan Adib  
M.Irfan Adib  
  
103112400257  
103112400257  
103112400257
```

Deskripsi:

Kode C++ ini mendemonstrasikan tiga jenis perulangan dasar. Pertama, perulangan for menjalankan pencetakan "M.Irfan Adib" sebanyak 10 kali menggunakan penghitung internal. Kedua, perulangan while mengulang pencetakan "103112400257" sebanyak 10 kali, dengan syarat yang dicek di awal setiap iterasi, dan membutuhkan peningkatan variabel secara manual. Terakhir, perulangan do-while menginisialisasi variabel j dengan nilai 0 dan mencetak 0 satu kali; karena kondisi perulangan (j != 0) langsung bernilai salah setelah eksekusi pertama, perulangan berhenti, menunjukkan bahwa do-while menjamin eksekusi blok kode minimal satu kali.

- C. Unguided/Tugas (berisi screenshot source code & output program disertai penjelasannya)

Unguided 1

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

int main() {
    float a, b, penjumlahan, pengurangan, perkalian, pembagian;

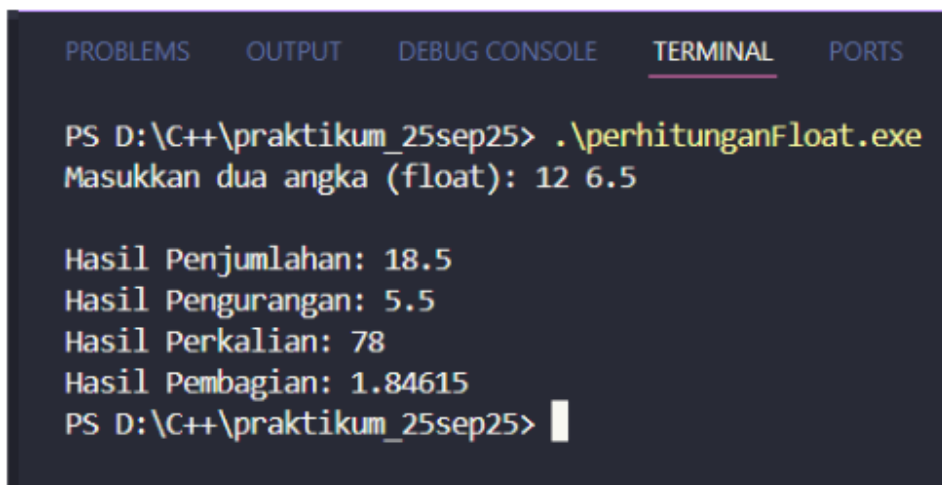
    cout << "Masukkan dua angka (float): ";
    cin >> a >> b;

    penjumlahan = a + b;
    pengurangan = a - b;
    perkalian = a * b;
    pembagian = a / b;

    cout << endl;
    cout << "Hasil Penjumlahan: " << penjumlahan << endl;
    cout << "Hasil Pengurangan: " << pengurangan << endl;
    cout << "Hasil Perkalian: " << perkalian << endl;
    cout << "Hasil Pembagian: " << pembagian << endl;

    return 0;
}
```

Screenshots Output



The screenshot shows a terminal window with the following content:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS D:\C++\praktikum_25sep25> .\perhitunganFloat.exe
Masukkan dua angka (float): 12 6.5

Hasil Penjumlahan: 18.5
Hasil Pengurangan: 5.5
Hasil Perkalian: 78
Hasil Pembagian: 1.84615
PS D:\C++\praktikum_25sep25> |
```


Deskripsi:

Program C++ sederhana ini bertujuan untuk melakukan empat operasi aritmatika dasar penjumlahan, pengurangan, perkalian, dan pembagian—terhadap dua bilangan bertipe floating-point. Setelah menyertakan pustaka `iostream` dan menggunakan namespace standar, program mendeklarasikan enam variabel float: dua untuk masukan (a dan b) dan empat untuk menyimpan hasil operasi. Program kemudian meminta pengguna untuk memasukkan dua angka (menggunakan `cout` dan `cin`), menghitung masing-masing operasi aritmatika, dan terakhir menampilkan semua hasil perhitungan (penjumlahan, pengurangan, perkalian, dan pembagian) ke layar konsol secara terpisah. Program diakhiri dengan `return 0`, menandakan eksekusi yang sukses.

Unguided 2

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

string bentukTulisan(int angka)
{
    switch (angka)
    {
        case 1:
            return "Satu";
        case 2:
            return "Dua";
        case 3:
            return "Tiga";
        case 4:
            return "Empat";
        case 5:
            return "Lima";
        case 6:
            return "Enam";
        case 7:
            return "Tujuh";
        case 8:
            return "Delapan";
        case 9:
            return "Sembilan";
    }
}
```

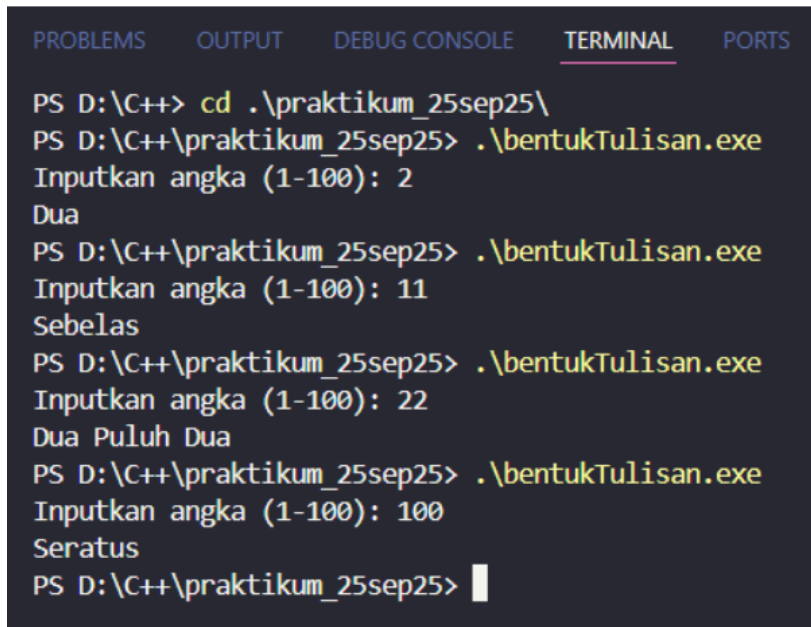
```
    case 10:
        return "Sepuluh";
    case 11:
        return "Sebelas";
    case 100:
        return "Seratus";
    default:
        break;
}

if (angka < 20)
{
    return bentukTulisan(angka - 10) + " Belas";
} else if (angka < 100)
{
    return bentukTulisan(angka / 10) + " Puluh " +
bentukTulisan(angka % 10);
}
return "";
}

int main()
{
    int angka;
    cout << "Inputkan angka (1-100): ";
    cin >> angka;

    cout << bentukTulisan(angka) << endl;
    return 0;
}
```

Screenshots Output



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS D:\C++> cd .\praktikum_25sep25\
PS D:\C++\praktikum_25sep25> .\bentukTulisan.exe
Inputkan angka (1-100): 2
Dua
PS D:\C++\praktikum_25sep25> .\bentukTulisan.exe
Inputkan angka (1-100): 11
Sebelas
PS D:\C++\praktikum_25sep25> .\bentukTulisan.exe
Inputkan angka (1-100): 22
Dua Puluh Dua
PS D:\C++\praktikum_25sep25> .\bentukTulisan.exe
Inputkan angka (1-100): 100
Seratus
PS D:\C++\praktikum_25sep25> 
```

Deskripsi:

Program C++ ini berfungsi untuk mengubah input angka bulat (integer) dari 1 hingga 100 menjadi bentuk tulisannya (terbilang) dalam bahasa Indonesia. Fungsi utamanya, `bentukTulisan(int angka)`, menggunakan kombinasi dari struktur kontrol switch untuk menangani kasus khusus (seperti 1 hingga 11, dan 100) dan rekursi dengan kondisi if-else untuk angka belasan dan puluhan. Jika angka di bawah 20 namun lebih dari 11, ia memanggil dirinya sendiri untuk menerjemahkan angka satuan (misalnya, untuk 15, memanggil `bentukTulisan(5)` lalu menambahkan "Belas"). Untuk angka puluhan (misalnya 42), ia membagi bilangan untuk mendapatkan puluhan ($\text{angka} / 10$) dan mencari sisa bagi untuk mendapatkan satuan ($\text{angka} \% 10$), lalu menggabungkannya dengan kata "Puluh". Fungsi `main()` meminta pengguna memasukkan angka dan kemudian menampilkan hasil konversi terbilang tersebut.

Unguided 3

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

int main()
{
    int angka;
    cout << "Inputkan angka: ";
    cin >> angka;
    cout << endl;

    for (int i = angka; i > 0; i--)
    {
        cout << i << " ";
        for (int j = i - 1; j >= 1; j--)
        {
            cout << j << " ";
        }

        cout << "* ";
        for (int k = 1; k <= i; k++)
        {
            cout << k << " ";
        }

        cout << endl;

        for (int l = 0; l < angka - i + 1; l++)
        {
            cout << " ";
        }
    }
    return 0;
}
```

Screenshots Output

```
PS D:\C++> & 'c:\Users\LAPTOP
osoft-MIEngine-In-w23einyz.f2c
t-MIEngine-Pid-x42qoyn2.fry' '
Inputkan angka: 3

3 2 1 * 1 2 3
 2 1 * 1 2
   1 * 1

PS D:\C++> |
```

Deskripsi:

Program C++ ini berfungsi untuk mencetak pola karakter dan angka bertingkat (berjenjang) ke konsol, dengan jumlah baris yang ditentukan oleh input bilangan bulat (integer) dari pengguna. Program ini menggunakan dua loop for bersarang utama di dalam loop for luar. Loop luar (i) bertanggung jawab untuk mengontrol baris dan dimulai dari angka input pengguna dan menurun hingga 1. Di setiap baris, loop pertama (j) mencetak deret angka menurun dari (i-1) hingga 1, diikuti oleh karakter *, dan loop kedua (k) mencetak deret angka menaik dari 1 hingga i. Selanjutnya, di bawah setiap baris pola utama, terdapat loop ketiga (l) yang bertugas membuat indentasi (spasi kosong) yang terus bertambah pada setiap baris berikutnya, menghasilkan efek pola berbentuk segitiga atau tangga terbalik yang tergeser ke kanan.

D. Kesimpulan

Praktikum ini berhasil memberikan pengalaman langsung dalam membangun fondasi pemrograman C++, dimulai dari implementasi dasar-dasar seperti variabel dan operator pada program kalkulator sederhana. Pemahaman ini kemudian berkembang pada penerapan logika percabangan if-else untuk penentu hari, diikuti dengan melatih kemampuan mengelola alur program melalui implementasi perulangan yang puncaknya terlihat pada pembuatan pola piramida angka menggunakan nested loops. Selain itu, praktikum juga mencakup konsep pengorganisasian data menggunakan struct (dicontohkan pada struct Mahasiswa) dan mengasah kemampuan memecahkan masalah secara efisien melalui fungsi rekursif yang diterapkan pada program konversi angka.

E. Referensi

Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2022). *Introduction to Algorithms* (4th ed.). The MIT Press.

Stroustrup, B. (2014). *Programming: Principles and Practice Using C++* (2nd ed.). Addison-Wesley Professional.