

**LAPORAN PRAKTIKUM
STRUKTUR DATA**

**MODUL 2
PENGENALAN BAHASA C++ (2)**



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

Praktikum ini bertujuan untuk memahami fondasi pemrograman C++ melalui implementasi konsep dasar seperti variabel dan operator pada program kalkulator, logika percabangan if-else untuk penentu hari, dan kemampuan mengelola alur program menggunakan perulangan yang berpuncak pada pembuatan pola piramida angka dengan nested loops. Selain itu, praktikum juga mencakup penggunaan struktur data seperti struct Mahasiswa untuk pengorganisasian data, serta fungsi rekursif untuk konversi angka. Program-program ini didukung oleh pemahaman dasar tentang struktur data berurutan seperti Array (yang nama dasarnya adalah pointer ke elemen pertama), dan perbedaan antara pengiriman parameter call by value (menerima salinan nilai) dengan call by pointer (menerima alamat memori untuk modifikasi langsung variabel asli). Semua materi ini bersumber dari panduan pemrograman dan referensi umum mengenai C++.

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

Guided 1

```
#include <iostream>
#define MAX 5

using namespace std;

int main()
{
    int i, j;
    float nilai[MAX];
    static int nilai_tahun[MAX][MAX] = {
        {0, 2, 2, 0, 0},
        {0, 1, 1, 1, 0},
        {0, 3, 3, 3, 0},
        {4, 4, 0, 0, 4},
        {5, 0, 0, 0, 5}
    };

    cout << "=== INPUT NILAI MAHASISWA ===\n";
    for (i = 0; i < MAX; i++) {
        cout << "Masukkan nilai Mahasiswa ke-" << i + 1 << ": ";
        cin >> nilai[i];
    }

    cout << "\n=== DATA NILAI MAHASISWA ===\n";
    for (i = 0; i < MAX; i++) {
        cout << "Nilai ke-" << i + 1 << " = " << nilai[i] << endl;
    }

    cout << "\n=== NILAI TAHUNAN ===\n";
    for (i = 0; i < MAX; i++) {
        for (j = 0; j < MAX; j++) {
            cout << nilai_tahun[i][j] << " ";
        }
        cout << endl;
    }
    return 0;
}
```

Screenshots Output

```
=== INPUT NILAI MAHASISWA ===
Masukkan nilai Mahasiswa ke-1: 60
Masukkan nilai Mahasiswa ke-2: 55
Masukkan nilai Mahasiswa ke-3: 48
Masukkan nilai Mahasiswa ke-4: 75
Masukkan nilai Mahasiswa ke-5: 67

=== DATA NILAI MAHASISWA ===
Nilai ke-1 = 60
Nilai ke-2 = 55
Nilai ke-3 = 48
Nilai ke-4 = 75
Nilai ke-5 = 67

=== NILAI TAHUNAN ===
0 2 2 0 0
0 1 1 1 0
0 3 3 3 0
4 4 0 0 4
5 0 0 0 5
```

Deskripsi:

Program C++ ini berfungsi untuk mendemonstrasikan penggunaan array satu dimensi dan array dua dimensi untuk menyimpan data nilai. Program dimulai dengan mendefinisikan konstanta MAX (bernilai 5) dan mendeklarasikan array nilai bertipe float untuk menampung lima nilai mahasiswa yang akan diinput oleh pengguna. Selain itu, dideklarasikan pula array dua dimensi nilai_tahun bertipe integer berukuran 5×5 yang diinisialisasi secara statis dengan sekumpulan nilai tertentu, merepresentasikan data nilai dalam format tabular. Setelah meminta dan menyimpan lima input nilai mahasiswa menggunakan loop for, program kemudian menampilkan kembali nilai-nilai yang telah diinput tersebut, dan diakhiri dengan mencetak seluruh elemen dari array dua dimensi nilai_tahun dalam format matriks 5×5, di mana nested loops digunakan untuk mengiterasi baris (i) dan kolom (j).

Guided 2

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

int maks3(int a, int b, int c);

int main()
{
    int x, y, z;
    cout << "Masukkan nilai bilangan ke-1: ";
    cin >> x;
    cout << "Masukkan nilai bilangan ke-2: ";
    cin >> y;
    cout << "Masukkan nilai bilangan ke-3: ";
    cin >> z;

    cout << "Nilai maksimumnya adalah = "
    << maks3(x, y, z) << endl;

    return 0;
}

int maks3(int a, int b, int c)
{
    int maks = a;
    if (b > maks) {
        maks = b;
    }
    if (c > maks) {
        maks = c;
    }
    return maks;
}
```

Screenshot Output

```
Masukkan nilai bilangan ke-1: 34
Masukkan nilai bilangan ke-2: 56
Masukkan nilai bilangan ke-3: 76
Nilai maksimumnya adalah = 76
```

Deskripsi:

Program C++ ini berfungsi untuk menentukan dan menampilkan nilai maksimum dari tiga bilangan bulat (integer) yang diinput oleh pengguna. Logika inti program diimplementasikan dalam sebuah fungsi terpisah bernama maks3(int a, int b, int c). Fungsi main() bertanggung jawab untuk menerima tiga input bilangan dari pengguna (x, y, dan z) dan memanggil fungsi maks3() untuk mendapatkan hasilnya, yang kemudian dicetak ke konsol. Di dalam fungsi maks3(), proses penentuan nilai maksimum dilakukan secara bertahap: ia mengasumsikan bilangan pertama (a) sebagai nilai maksimum awal, kemudian membandingkannya secara berurutan dengan bilangan kedua (b) dan ketiga (c) menggunakan struktur kondisional if. Jika salah satu bilangan berikutnya lebih besar dari nilai maksimum saat ini, maka nilai maksimum akan diperbarui, dan hasil akhirnya dikembalikan ke fungsi main().

Guided 3

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

int main()
{
    int x,y;
    int *px;
    x = 87;
    px = &x;
    y= *px;

    cout << "Alamat x    = " << &x << endl;
    cout << "Isi px      = " << px << endl;
    cout << "Isi x        = " << x << endl;
    cout << "Nilai *px     = " << *px << endl;

    return 0;
}
```

Screenshots Output

```
Alamat x    = 0xd075bffd0
Isi px      = 0xd075bffd0
Isi x       = 87
Nilai *px   = 87
```

Deskripsi:

Program C++ ini mendemonstrasikan konsep dasar pointer. Program mendeklarasikan dua variabel integer biasa (x dan y) dan satu variabel pointer integer (*px). Variabel x diinisialisasi dengan nilai 87. Kemudian, pointer px dialamatkan ke variabel x menggunakan operator alamat (&). Selanjutnya, nilai dari lokasi memori yang ditunjuk oleh px (yaitu nilai x) diberikan ke variabel y menggunakan operator dereference (*). Program ini kemudian menampilkan ke konsol: alamat memori x (&x), isi pointer px (yang merupakan alamat memori x), nilai variabel x itu sendiri, dan nilai yang ditunjuk oleh pointer px (*px), yang semuanya akan menunjukkan nilai 87.

Guided 4

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

void tulis(int x);

int main()
{
    int jumlah;
    cout << "Masukkan jumlah baris kata = ";
    cin >> jumlah;
    tulis(jumlah);
    return 0;
}

void tulis(int x)
{
    for (int i = 0; i < x; i++) {
        cout << "Baris ke-" << i + 1 << endl;
    }
}
```

Screenshots Output

```
Masukkan jumlah baris kata = 5  
Baris ke-1  
Baris ke-2  
Baris ke-3  
Baris ke-4  
Baris ke-5
```

Deskripsi:

Program C++ ini bertujuan untuk mencetak serangkaian kalimat yang berulang ke konsol, dengan jumlah pengulangan ditentukan oleh input pengguna, menggunakan implementasi fungsi (void tulis(int x)). Fungsi main() adalah entry point yang meminta pengguna memasukkan jumlah baris yang diinginkan (jumlah) dan kemudian memanggil fungsi tulis() sambil mengirimkan nilai jumlah tersebut sebagai parameter (x). Fungsi tulis() kemudian menggunakan loop for untuk berulang dari 0 hingga kurang dari nilai parameter x. Dalam setiap iterasi loop, fungsi tersebut mencetak pesan "Baris ke-" diikuti oleh nomor baris saat ini (i + 1), sehingga program secara keseluruhan berhasil mencetak kalimat sebanyak jumlah yang diminta pengguna.

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

Unguided 1

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

int main()
{
    const int JML_MHS = 5;
    const int JML_MK = 3;
    string nama_mhs[JML_MHS];
    float nilai[JML_MHS][JML_MK];
    float rata_rata[JML_MHS];
    int i, j;

    cout << "=== INPUT NILAI MAHASISWA ===\n";
    for (i = 0; i < JML_MHS; i++) {
        cout << "Masukkan nama Mahasiswa ke-" << i + 1 << ": ";
        cin >> nama_mhs[i];
        for (j = 0; j < JML_MK; j++) {
            cout << "    Masukkan nilai Mata Kuliah ke-" << j + 1 <<
": ";

            cin >> nilai[i][j];
        }
        cout << endl;
    }

    for (i = 0; i < JML_MHS; i++) {
        float total = 0;
        for (j = 0; j < JML_MK; j++) {
            total = total + nilai[i][j];
        }
        rata_rata[i] = total / JML_MK;
    }

    int indeks_terbaik = 0;
    float max_rata = rata_rata[0];
    for (i = 1; i < JML_MHS; i++) {
        if (rata_rata[i] > max_rata) {
```

```

        max_rata = rata_rata[i];
        indeks_terbaik = i;
    }
}

cout << "\n=== TABEL NILAI MAHASISWA ===\n";
cout << "Nama\t\tMK1\tMK2\tMK3\tRata-rata\tKeterangan\n";
cout <<
"-----
\n";
    for (i = 0; i < JML_MHS; i++) {
        cout << nama_mhs[i] << "\t\t";
        for (j = 0; j < JML_MK; j++) {
            cout << nilai[i][j] << "\t";
        }
        cout << rata_rata[i];
        if (i == indeks_terbaik) {
            cout << "\t\t(Tertinggi)";
        }
        cout << endl;
    }

    return 0;
}

```

Screenshots Output

```
=== INPUT NILAI MAHASISWA ===
Masukkan nama Mahasiswa ke-1: ahmad
Masukkan nilai Mata Kuliah ke-1: 34
Masukkan nilai Mata Kuliah ke-2: 54
Masukkan nilai Mata Kuliah ke-3: 76

Masukkan nama Mahasiswa ke-2: budi
Masukkan nilai Mata Kuliah ke-1: 78
Masukkan nilai Mata Kuliah ke-2: 21
Masukkan nilai Mata Kuliah ke-3: 56

Masukkan nama Mahasiswa ke-3: samsul
Masukkan nilai Mata Kuliah ke-1: 56
Masukkan nilai Mata Kuliah ke-2: 89
Masukkan nilai Mata Kuliah ke-3: 43

Masukkan nama Mahasiswa ke-4: panjul
Masukkan nilai Mata Kuliah ke-1: 34
Masukkan nilai Mata Kuliah ke-2: 15
Masukkan nilai Mata Kuliah ke-3: 73

Masukkan nama Mahasiswa ke-5: slamet
Masukkan nilai Mata Kuliah ke-1: 87
Masukkan nilai Mata Kuliah ke-2: 45
Masukkan nilai Mata Kuliah ke-3: 94

=== TABEL NILAI MAHASISWA ===
```

Nama	MK1	MK2	MK3	Rata-rata	Keterangan
ahmad	34	54	76	54.6667	
budi	78	21	56	51.6667	
samsul	56	89	43	62.6667	
panjul	34	15	73	40.6667	
slamet	87	45	94	75.3333	(Tertinggi)

Deskripsi:

Program C++ ini dirancang untuk mengelola dan menganalisis nilai dari 5 mahasiswa (JML_MHS) untuk 3 mata kuliah (JML_MK). Program menggunakan array satu dimensi (nama_mhs) untuk nama dan array dua dimensi (nilai) untuk nilai mata kuliah, serta array satu dimensi lain (rata_rata) untuk menyimpan rata-rata nilai per mahasiswa. Prosesnya dimulai dengan nested loops untuk menginput nama dan tiga nilai untuk setiap mahasiswa. Kemudian, program menggunakan loop terpisah untuk menghitung rata-rata nilai setiap mahasiswa. Setelah itu, loop ketiga digunakan untuk mencari dan menandai mahasiswa dengan nilai rata-rata tertinggi. Terakhir, program menampilkan tabel ringkasan yang memuat nama, nilai per mata kuliah, nilai rata-rata, dan keterangan khusus untuk mahasiswa yang mendapat rata-rata tertinggi.

Unguided 2

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

int main()
{
    int N, i;
    int arr[100];
    int *p;

    cout << "Masukkan jumlah bilangan (N): ";
    cin >> N;
    p = arr;

    cout << "\nMasukkan " << N << " bilangan:\n";
    for (i = 0; i < N; i++) {
        cout << "Bilangan ke-" << i + 1 << ": ";
        cin >> *(p + i);
    }

    int jumlah = 0;
    int maks = *p;
    int min = *p;

    for (i = 0; i < N; i++) {
        jumlah = jumlah + *(p + i);
        if (*(p + i) > maks) {
            maks = *(p + i);
        }

        if (*(p + i) < min) {
            min = *(p + i);
        }
    }

    cout << "\n=== HASIL PERHITUNGAN ===\n";
    cout << "Jumlah seluruh bilangan = " << jumlah << endl;
    cout << "Nilai maksimum          = " << maks << endl;
    cout << "Nilai minimum          = " << min << endl;

    return 0;
}
```

Screenshot Output

```
Masukkan jumlah bilangan (N): 2

Masukkan 2 bilangan:
Bilangan ke-1: 75
Bilangan ke-2: 12

=== HASIL PERHITUNGAN ===
Jumlah seluruh bilangan = 87
Nilai maksimum          = 75
Nilai minimum           = 12
```

Deskripsi:

Program C++ ini berfungsi untuk melakukan analisis statistik dasar—menghitung jumlah, nilai maksimum, dan nilai minimum—pada sekumpulan bilangan bulat yang diinput oleh pengguna, dengan memanfaatkan konsep pointer dan aritmatika pointer. Program dimulai dengan meminta jumlah bilangan (N) yang akan dimasukkan, lalu menginisialisasi pointer *p* untuk menunjuk ke elemen pertama dari array *arr*. Seluruh proses input bilangan ke dalam array dan akses data selanjutnya dilakukan melalui pointer *p* dengan notasi $*(p + i)$, yang mendemonstrasikan hubungan erat antara array dan pointer dalam C++. Setelah semua data tersimpan, program melakukan satu iterasi loop untuk secara simultan mengakumulasi total nilai (jumlah) dan membandingkan setiap elemen untuk menemukan nilai tertinggi (maks) dan terendah (min). Hasil akhir dari ketiga perhitungan tersebut kemudian ditampilkan ke konsol.

Unguided 3

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

float hitungRataRata(int nilai[], int n);
void cariTertinggiTerendah(int nilai[], int n, int *tertinggi, int *terendah);

int main()
{
    int N;
    int arr_nilai[100];
```

```

    float rata_rata;
    int nilai_tertinggi, nilai_terendah;

    cout << "Masukkan jumlah siswa (N): ";
    cin >> N;
    cout << "\nMasukkan nilai ujian untuk " << N << " siswa:\n";
    for (int i = 0; i < N; i++) {
        cout << "Nilai siswa ke-" << i + 1 << ": ";
        cin >> arr_nilai[i];
    }

    rata_rata = hitungRataRata(arr_nilai, N);
    cariTertinggiTerendah(arr_nilai, N, &nilai_tertinggi,
&nilai_terendah);

    // 4. Menampilkan hasil
    cout << "\n=== HASIL ANALISIS NILAI ===\n";
    cout << "Rata-rata kelas    = " << rata_rata << endl;
    cout << "Nilai tertinggi    = " << nilai_tertinggi << endl;
    cout << "Nilai terendah      = " << nilai_terendah << endl;

    return 0;
}

float hitungRataRata(int nilai[], int n)
{
    float total = 0;
    for (int i = 0; i < n; i++) {
        total += nilai[i];
    }
    return total / n;
}

void cariTertinggiTerendah(int nilai[], int n, int *tertinggi, int
*terendah)
{
    *tertinggi = nilai[0];
    *terendah = nilai[0];

    for (int i = 1; i < n; i++) {
        if (nilai[i] > *tertinggi) {
            *tertinggi = nilai[i];
        }
    }
}

```

```
        if (nilai[i] < *terendah) {  
            *terendah = nilai[i];  
        }  
    }  
}
```

Screenshots Output

```
Masukkan jumlah siswa (N): 2  
  
Masukkan nilai ujian untuk 2 siswa:  
Nilai siswa ke-1: 56  
Nilai siswa ke-2: 75  
  
=== HASIL ANALISIS NILAI ===  
Rata-rata kelas      = 65.5  
Nilai tertinggi      = 75  
Nilai terendah       = 56
```

Deskripsi:

Program C++ ini dirancang untuk melakukan analisis statistik dasar (menghitung rata-rata, nilai tertinggi, dan nilai terendah) dari sekumpulan nilai ujian siswa, dengan menggunakan dua fungsi terpisah untuk pemrosesan data. Fungsi main() bertanggung jawab untuk meminta jumlah siswa (N) dan menerima input nilai ujian ke dalam array arr_nilai. Perhitungan rata-rata ditangani oleh fungsi hitungRataRata, yang menerima array sebagai parameter dan mengembalikan nilai float rata-rata. Sementara itu, fungsi cariTertinggiTerendah menerima array nilai bersama dengan dua pointer (*tertinggi dan *terendah) dan mengimplementasikan mekanisme call by pointer, memungkinkan fungsi tersebut untuk memodifikasi langsung variabel nilai_tertinggi dan nilai_terendah di dalam main(). Hasil analisis dari ketiga statistik tersebut kemudian ditampilkan di akhir program.

Unguided 4

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

void cetakPolaSegitiga(int n);

int main()
{
    int tinggi;
    cout << "Masukkan tinggi segitiga (n): ";
    cin >> tinggi;

    cout << "\n=== POLA SEGITIGA ANGKA ===\n";
    cetakPolaSegitiga(tinggi);
    return 0;
}

void cetakPolaSegitiga(int n)
{
    int i, j;
    for (i = 1; i <= n; i++) {
        for (j = 1; j <= i; j++) {
            cout << j << " ";
        }
        cout << endl;
    }
}
```

Screenshots Output

```
Masukkan tinggi segitiga (n): 3

=== POLA SEGITIGA ANGKA ===
1
1 2
1 2 3
```


Deskripsi:

Program C++ ini berfungsi untuk mencetak pola segitiga siku-siku yang berisi angka ke konsol, dengan tinggi pola ditentukan oleh input bilangan bulat (tinggi) dari pengguna. Seluruh logika pencetakan pola dienkapsulasi dalam fungsi terpisah bernama `void cetakPolaSegitiga(int n)`. Fungsi ini menggunakan struktur yang didominasi oleh loop luar (i) yang mengontrol jumlah baris (dari 1 hingga n), dan loop dalam (j) yang mengontrol jumlah dan isi elemen yang dicetak per baris. Karena loop dalam (j) berjalan hanya sampai nilai loop luar (i), pola yang dihasilkan adalah segitiga siku-siku. Dalam setiap perulangan loop dalam, nilai counter j dicetak, sehingga setiap baris dimulai dari angka 1 dan terus menaik hingga nomor baris saat itu. perulangan bersarang (nested loop).

D. Kesimpulan

Praktikum ini berhasil memberikan pengalaman langsung dalam memahami fondasi C++, dimulai dari dasar-dasar seperti implementasi variabel dan operator pada kalkulator sederhana, hingga penerapan logika percabangan if-else dan pengelolaan alur program melalui perulangan (looping), yang berpuncak pada pembuatan pola piramida angka menggunakan nested loops. Selain itu, konsep fundamental C++ seperti penggunaan array satu dan dua dimensi untuk pengelompokan data juga dicakup, bersama dengan konsep pengorganisasian data yang rapi lewat struct Mahasiswa. Praktikum ini juga memperkenalkan pemanfaatan pointer dan aritmatika pointer untuk manipulasi memori secara efisien serta menunjukkan bagaimana memecah program menjadi modul terstruktur menggunakan fungsi dan prosedur (fungsi void). Hal ini mencakup pemahaman penting mengenai mekanisme pengiriman parameter, baik secara nilai (by value) maupun alamat (by pointer), yang memungkinkan fungsi untuk melakukan tugas spesifik seperti perhitungan, pencarian nilai, atau pencetakan pola, serta melatih kemampuan memecahkan masalah secara efisien dengan fungsi rekursif.

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.