

**LAPORAN PRAKTIKUM  
STRUKTUR DATA**

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



**Disusun Oleh :**

NAMA : Loh Suwargi Nitis Hamengku Bintang  
NIM : 103112400116

**Dosen**

FAHRUDIN MUKTI WIBOWO

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

## A. Dasar Teori

Struktur data menjelaskan tentang sekumpulan data yang dapat diidentifikasi berdasarkan cara pengorganisasian serta operasi yang dapat dilakukan terhadapnya. Penggunaan struktur data yang sesuai dalam proses pemrograman akan membantu membentuk algoritma yang lebih jelas dan efisien, sehingga keseluruhan program menjadi lebih sederhana. Dalam struktur data terdapat beberapa bentuk dasar, salah satunya adalah *array*. Array dapat dianggap sebagai hasil dari perkembangan teknologi yang lahir dari pemikiran manusia untuk mempermudah penyelesaian berbagai perhitungan.

Array merupakan struktur data yang terdiri atas sejumlah elemen dengan tipe data yang sama. Setiap elemen array tersimpan secara berurutan di dalam memori komputer. Semua elemen yang ada di dalam array memiliki tipe data seragam. Struktur ini sangat sesuai digunakan untuk mengelola sekumpulan data yang bersifat homogen, di mana jumlah atau ukuran elemennya sudah diketahui sejak awal. Istilah *homogen* berarti bahwa setiap elemen dalam array harus memiliki jenis data yang sama.

Array atau variabel larik adalah tipe data terstruktur yang memuat sejumlah elemen dengan tipe data yang sama dan tersimpan secara berurutan dalam memori. Jumlah elemennya bersifat tetap dan diidentifikasi melalui indeks yang berfungsi membedakan setiap elemen. Dalam bahasa pemrograman C++ dan Java, indeks array dimulai dari 0, berbeda dengan Pascal yang memulainya dari 1. Array dideklarasikan menggunakan tanda [ ], dan berfungsi sebagai penampung data homogen yang diakses menggunakan satu nama variabel. Setiap elemen array diakses melalui indeks bernilai 0 hingga n-1, di mana n merupakan jumlah total elemen yang tersimpan.

Pointer merupakan variabel yang berisi alamat memori dari variabel lain. Satuan dasar dalam memori adalah bit, yaitu unit terkecil penyimpanan data. Bit-bit tersebut dikelompokkan menjadi delapan bit yang disebut byte. Memori utama (RAM) komputer terdiri atas kumpulan byte yang diberi nomor secara berurutan, dan nomor tersebut berfungsi sebagai alamat memori. Saat sebuah variabel dideklarasikan, tipe data dan nama variabel ditentukan, kemudian sistem operasi bersama kompilator akan menyediakan blok memori untuk menyimpan nilainya. Jumlah byte yang dialokasikan bergantung pada tipe data variabel tersebut

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

### Guided 1

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

int main() {
    int x, y;
    int *px;
```

```

x = 87;
px = &x;
y = *px;

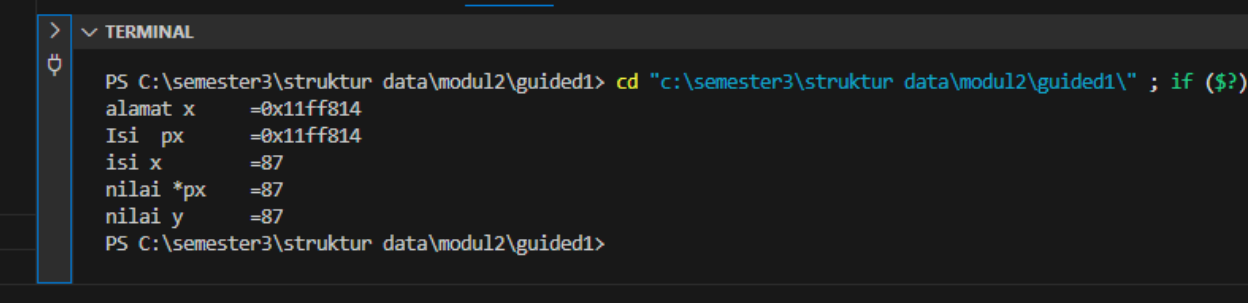
cout << "alamat x   =" << &x << endl;
cout << "Isi  px    =" << px << endl;
cout << "isi x      =" << x << endl;
cout << "nilai *px   =" << *px << endl;
cout << "nilai y     =" << y << endl;

return 0;

}

```

### Screenshots Output



```

> v TERMINAL
PS C:\semester3\struktur data\modul2\guided1> cd "c:\semester3\struktur data\modul2\guided1\" ; if ($?) {
alamat x   =0x11ff814
Isi  px    =0x11ff814
isi x      =87
nilai *px   =87
nilai y     =87
PS C:\semester3\struktur data\modul2\guided1>

```

Deskripsi: Program ini dirancang untuk menunjukkan cara kerja pointer dalam C++. Variabel x diberi nilai 87, kemudian pointer px menyimpan alamat memori dari x menggunakan operator &. Nilai \*px merepresentasikan isi dari alamat tersebut, sehingga ketika y = \*px, variabel y juga bernilai 87. Program ini menampilkan alamat x, isi pointer px, nilai x, nilai \*px, dan nilai y. Hasilnya menunjukkan bahwa x, \*px, dan y memiliki nilai yang sama. Secara singkat, program ini menjelaskan bahwa pointer digunakan untuk menyimpan dan mengakses nilai variabel melalui alamat memorinya.

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

#### Guided 2

```

#include <iostream>

#define MAX 5

using namespace std;

```

```

int main() {
    int i, j;
    float nilai[MAX]; // array 1 dimensi
    static int nilai_tahun[MAX][MAX] = { //array
        {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}
    };

    //input data array 1 dimensi
    cout << "=== Input nilai Siswa ===\n";
    for ( i = 0; i < MAX; i++){
        cout << "masukkan nilai ke-" << i + 1 << ": ";
        cin >> nilai[i];
    }

    // manampilkan isi array 1 dimensi
    cout << "\n=== Data Nilai Siswa ===\n";
    for (i = 0; i < MAX; i++){
        cout << "Nilai ke-" << i + 1 << " = "
        << nilai[i]
        << endl;
    }

    // manampilkan isi array 2 dimensi
    cout << "\n=== Data 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

```

> ▾ TERMINAL
❏ masukkan nilai ke-3: 8
   masukkan nilai ke-4: 76
   masukkan nilai ke-5: 98

   === Data Nilai Siswa ===
   Nilai ke-1 = 99
   Nilai ke-2 = 89
   Nilai ke-3 = 8
   Nilai ke-4 = 76
   Nilai ke-5 = 98

   === Data 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: Konstanta MAX bernilai 5 digunakan sebagai ukuran array. Array nilai digunakan untuk menyimpan lima nilai siswa yang diinput oleh pengguna, sedangkan nilai\_tahun merupakan array dua dimensi yang sudah diisi sebelumnya. Setelah data dimasukkan, program menampilkan isi array satu dimensi berisi nilai siswa, kemudian menampilkan isi array dua dimensi yang merepresentasikan data nilai tahunan. Secara singkat, program ini memperlihatkan cara menginput dan menampilkan data menggunakan array serta bagaimana perulangan bersarang digunakan untuk membaca isi array dua dimensi.

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

### Guided 3

```

#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 = ";

```

```

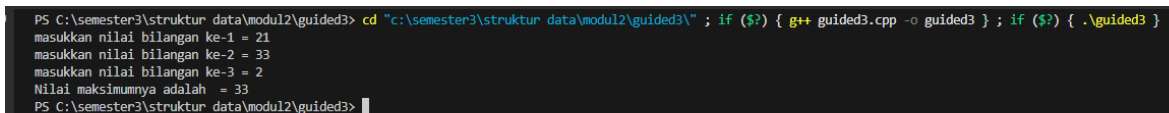
        cin >> z;

        cout << "Nilai maksimumnya adalah = "
        << maks3(x, y, z);
        return 0;
    }

    // badan fungsi
    int maks3(int a, int b, int c){
        int temp_max = a;
        if (b > temp_max)
            temp_max = b;
        if (b > temp_max)
            temp_max = c;
        return temp_max;
    }
}

```

### Screenshots Output



```

PS C:\semester3\struktur data\modul2\guided3> cd "c:\semester3\struktur data\modul2\guided3\" ; if ($?) { g++ guided3.cpp -o guided3 } ; if ($?) { .\guided3 }
masukkan nilai bilangan ke-1 = 21
masukkan nilai bilangan ke-2 = 33
masukkan nilai bilangan ke-3 = 2
Nilai maksimumnya adalah = 33
PS C:\semester3\struktur data\modul2\guided3>

```

Deskripsi: Pengguna diminta untuk menginput tiga nilai, yaitu x, y, dan z. Ketiga nilai ini kemudian dikirim ke fungsi maks3(x, y, z) untuk dibandingkan. Fungsi maks3 memiliki tiga parameter (a, b, dan c) yang masing-masing mewakili nilai dari x, y, dan z. Variabel temp\_max digunakan untuk menyimpan nilai terbesar sementara. Awalnya, temp\_max diisi dengan nilai a. Lalu dilakukan perbandingan dengan b dan c menggunakan pernyataan if. Jika salah satu nilai lebih besar dari temp\_max, maka nilainya akan diganti. Setelah proses perbandingan selesai, fungsi mengembalikan nilai terbesar tersebut ke fungsi utama (main) dan hasilnya ditampilkan ke layar.

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

### Guided 4

```

#include <iostream>

```

```

using namespace std;

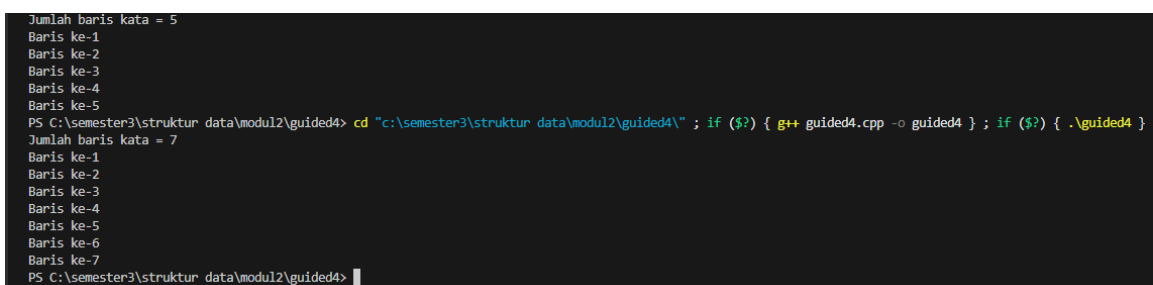
//prototype prosedur
void tulis (int x);

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

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

```

## Screenshots Output



```

Jumlah baris kata = 5
Baris ke-1
Baris ke-2
Baris ke-3
Baris ke-4
Baris ke-5
PS C:\semester3\struktur data\modul2\guided4> cd "c:\semester3\struktur data\modul2\guided4\" ; if ($?) { g++ guided4.cpp -o guided4 } ; if ($?) { .\guided4 }
Jumlah baris kata = 7
Baris ke-1
Baris ke-2
Baris ke-3
Baris ke-4
Baris ke-5
Baris ke-6
Baris ke-7
PS C:\semester3\struktur data\modul2\guided4>

```

Deskripsi: Ketika program dijalankan, pengguna diminta memasukkan sebuah bilangan bulat melalui perintah `cin >> jum`, yang menyatakan jumlah baris yang ingin ditampilkan. Nilai yang dimasukkan pengguna kemudian dikirim sebagai argumen ke prosedur `tulis(jum)`. Di dalam prosedur `tulis`, terdapat perulangan `for` yang berjalan sebanyak `x` kali (sesuai nilai yang dimasukkan oleh pengguna). Pada setiap iterasi, program akan

mencetak teks "Baris ke-" diikuti dengan nomor urut baris (i + 1).

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

Unguided 1

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

int main() {
    string nama[5];
    float nilai[5][3];
    float rata[5];
    int terbaik = 0;

    cout << "=== Input Data Nilai Mahasiswa ===" << endl;
    for (int i = 0; i < 5; i++) {
        cout << "\nNama mahasiswa ke-" << i + 1 << ": ";
        cin >> nama[i];
        float total = 0;
        for (int j = 0; j < 3; j++) {
            cout << "Nilai mata kuliah ke-" << j + 1 << ": ";
            cin >> nilai[i][j];
            total += nilai[i][j];
        }
        rata[i] = total / 3;
    }

    for (int i = 1; i < 5; i++) {
        if (rata[i] > rata[terbaik]) {
            terbaik = i;
        }
    }
}
```



```

    }
}

cout << "\n=== Daftar Nilai Mahasiswa ===" << endl;
cout << left << setw(12) << "Nama"
    << setw(10) << "MK1"
    << setw(10) << "MK2"
    << setw(10) << "MK3"
    << setw(10) << "Rata-rata"
    << "Keterangan" << endl;

for (int i = 0; i < 5; i++) {
    cout << left << setw(12) << nama[i];
    for (int j = 0; j < 3; j++) {
        cout << setw(10) << nilai[i][j];
    }
    cout << setw(10) << rata[i];
    if (i == terbaik) cout << " <- Terbaik";
    cout << endl;
}

return 0;
}

```

Screenshots Output

```
Nilai Mata Kuliah Ke-3: 85

=== Daftar Nilai Mahasiswa ===
Nama      MK1      MK2      MK3      Rata-rata Keterangan
bintang   98      80      88      88.6667
faiz      89      90      91      90
dzaki     90      90      85      88.3333
azzam     85      94      94      91
juna      95      95      89      93      <- Terbaik
PS C:\semester3\struktur data\modul2\soal1> |
```

Deskripsi: Program ini dibuat untuk mengelola data nilai mahasiswa dengan menggunakan array dua dimensi berukuran 5×3. Setiap baris dalam array mewakili seorang mahasiswa, sementara setiap kolom berisi nilai dari tiga mata kuliah yang berbeda. Program meminta pengguna untuk memasukkan nama masing-masing dari lima mahasiswa, kemudian memasukkan tiga nilai mata kuliah untuk setiap mahasiswa. Setelah semua data diinput, program menghitung rata-rata nilai setiap mahasiswa dan menentukan siapa yang memiliki rata-rata tertinggi. Hasil akhirnya ditampilkan dalam bentuk tabel rapi yang berisi nama, tiga nilai mata kuliah, rata-rata nilai, serta penanda khusus bagi mahasiswa yang memiliki nilai rata-rata tertinggi. Program ini melatih pemahaman dasar tentang penggunaan array 2 dimensi, perulangan bersarang, serta proses perhitungan sederhana menggunakan struktur data dasar di C++.

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

Unguided 2

```
#include <iostream>

using namespace std;

int main() {
    int N;

    cout << "Masukkan jumlah elemen array: ";
    cin >> N;

    int *arr = new int[N];
    int *ptr = arr;

    cout << "Masukkan " << N << " bilangan:" << endl;
    for (int i = 0; i < N; i++) {
        cout << "Elemen ke-" << i + 1 << ": ";
        cin >> *(ptr + i);
```

```

    }

    int jumlah = 0, maks = *ptr, min = *ptr;

    for (int i = 0; i < N; i++) {
        int val = *(ptr + i);
        jumlah += val;
        if (val > maks) maks = val;
        if (val < min) min = val;
    }

    cout << "\nHasil Perhitungan:" << endl;
    cout << "Jumlah = " << jumlah << endl;
    cout << "Nilai Maksimum = " << maks << endl;
    cout << "Nilai Minimum = " << min << endl;

    delete[] arr; // hapus alokasi memori

    return 0;
}

```

## Screenshots Output



```

PS C:\semester3\struktur data\modul2\soal2> cd "c:\semester3\struktur data\modul2\soal2\" ; if ($?) { g++ soal2.cpp -o soal2 } ; if ($?) { .\soal2 }
Masukkan jumlah elemen array: 5
Masukkan 5 bilangan:
Elemen ke-1: 5
Elemen ke-2: 7
Elemen ke-3: 8
Elemen ke-4: 6
Elemen ke-5: 6

Hasil Perhitungan:
Jumlah = 32
Nilai Maksimum = 8
Nilai Minimum = 5
PS C:\semester3\struktur data\modul2\soal2>

```

Deskripsi: Program diawali dengan meminta pengguna untuk menentukan jumlah elemen yang akan dimasukkan ke dalam array. Setelah jumlah tersebut ditentukan, program membuat alokasi memori dinamis menggunakan pointer agar ukuran array dapat menyesuaikan input pengguna. Selanjutnya, pengguna diminta mengisi setiap elemen array melalui pointer, bukan melalui nama array secara langsung. Setelah semua data terinput, pointer digunakan untuk menghitung jumlah total bilangan,

sekalius menentukan nilai maksimum dan nilai minimum dari elemen yang dimasukkan

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

Unguided 3

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

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

void cariNilai(float arr[], int n, float &maks, float &min) {
    maks = arr[0];
    min = arr[0];
    for (int i = 1; i < n; i++) {
        if (arr[i] > maks) maks = arr[i];
        if (arr[i] < min) min = arr[i];
    }
}

int main() {
    int N;
    cout << "Masukkan jumlah siswa: ";
    cin >> N;

    float nilai[100];
    for (int i = 0; i < N; i++) {
```

```

        cout << "Nilai siswa ke-" << i + 1 << ": ";
        cin >> nilai[i];
    }

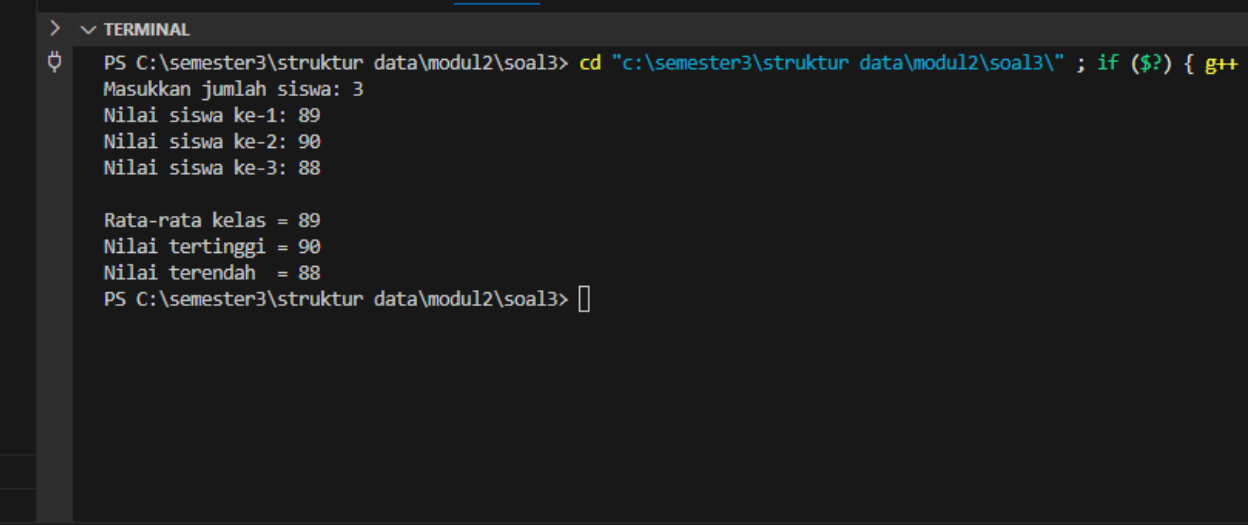
    float rata = hitungRata(nilai, N);
    float maks, min;
    cariNilai(nilai, N, maks, min);

    cout << "\nRata-rata kelas = " << rata << endl;
    cout << "Nilai tertinggi = " << maks << endl;
    cout << "Nilai terendah = " << min << endl;

    return 0;
}

```

### Screenshots Output



```

> ▾ TERMINAL
PS C:\semester3\struktur data\modul2\soal3> cd "c:\semester3\struktur data\modul2\soal3\" ; if ($?) { g++
Masukkan jumlah siswa: 3
Nilai siswa ke-1: 89
Nilai siswa ke-2: 90
Nilai siswa ke-3: 88

Rata-rata kelas = 89
Nilai tertinggi = 90
Nilai terendah = 88
PS C:\semester3\struktur data\modul2\soal3> 

```

Deskripsi: Program ini dirancang untuk menginput sejumlah nilai ujian dari beberapa siswa, yang kemudian disimpan dalam sebuah array satu dimensi. Setelah semua nilai dimasukkan, program memanggil dua fungsi terpisah: fungsi pertama bertugas untuk menghitung rata-rata nilai seluruh siswa, sedangkan fungsi kedua digunakan untuk mencari nilai tertinggi dan nilai terendah di dalam array.

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

Unguided 4

```
#include <iostream>

using namespace std;

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

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

    cout << "\nPola Segitiga Angka:\n";
    tampilSegitiga(n);

    return 0;
}
```

Screenshots Output



```
PS C:\semester3\struktur data\modul2\soal4> cd "c:\semester3\struktur data\modul2\soal4\" ; if ($?) { g++ soal4.cpp -o soal4 } ; if ($?) { .\soal4 }
Masukkan nilai n: 4

Pola Segitiga Angka:
1
1 2
1 2 3
1 2 3 4
PS C:\semester3\struktur data\modul2\soal4> cd "c:\semester3\struktur data\modul2\soal4\" ; if ($?) { g++ soal4.cpp -o soal4 } ; if ($?) { .\soal4 }
Masukkan nilai n: 5

Pola Segitiga Angka:
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
PS C:\semester3\struktur data\modul2\soal4>
```

Deskripsi: Pengguna diminta untuk memasukkan sebuah bilangan bulat  $n$ , kemudian program akan mencetak pola berbentuk segitiga, di mana baris pertama berisi angka 1, baris kedua berisi "1 2", dan seterusnya hingga baris ke- $n$  berisi deretan angka dari 1 sampai  $n$ . Program ini menggunakan perulangan bersarang di dalam sebuah prosedur bernama `tampilSegitiga`. Dengan memisahkan logika pencetakan pola ke dalam prosedur, program menjadi lebih rapi, mudah dipahami, dan dapat digunakan kembali jika diperlukan.

## K. Kesimpulan

Secara keseluruhan, dari seluruh program yang telah dibahas, dapat disimpulkan bahwa masing-masing memiliki tujuan untuk memperkenalkan konsep dasar pemrograman terstruktur dalam C++. Program yang menggunakan array satu dimensi dan dua dimensi untuk mengajarkan cara menyimpan serta menampilkan data secara terorganisir. Program tentang pointer memperlihatkan bagaimana suatu variabel dapat diakses melalui alamat memorinya, sedangkan program yang menggunakan fungsi dengan parameter menunjukkan cara mengirim data antarbagian program untuk diproses, seperti menghitung nilai maksimum atau rata-rata.

Selain itu, program dengan prosedur (fungsi void) memberikan pemahaman tentang bagaimana menjalankan suatu perintah berulang tanpa perlu mengembalikan nilai. Melalui seluruh contoh ini, dapat dipahami bahwa konsep seperti array, pointer, fungsi, dan prosedur merupakan dasar penting dalam membangun program yang efisien, rapi, dan mudah dipelihara. Penguasaan terhadap konsep dasar ini menjadi langkah awal yang sangat penting sebelum mempelajari topik pemrograman yang lebih kompleks di C++.

## L. Referensi

**Pratama, M. A. (2020). STRUKTUR DATA ARRAY DUA DIMENSI PADA PEMROGRAMAN C++.**

**Anita Sindar, R. M. S. (2019). *Struktur Data Dan Algoritma Dengan C++* (Vol. 1). CV. AA. RIZKY.**

**Sianipar, R. H. (2017). *Teori dan Aplikasi C++ dengan Contoh Lebih dari 280 Source***

***Code* (Vol. 1). Penerbit ANDI.**