

**LAPORAN PRAKTIKUM**  
**Modul 02**  
**“PENGENALAN BAHASA C++ (BAGIAN KEDUA)”**



**Disusun Oleh:**  
**Ganesha Rahman Gibran -2211104058**  
**Kelas S1SE-07-02**

**Dosen :**  
**Wahyu Andi Saputra, S.Pd., M.Eng.**

## Tujuan

1. Memahami penggunaan pointer dan alamat memori
2. Mengimplementasikan fungsi dan prosedur dalam program

## Landasan Teori

- **Array**

Array dalam C++ adalah struktur data yang menyimpan sekumpulan elemen dengan tipe data yang sama dalam memori yang berurutan. Array dideklarasikan dengan sintaks:

```
tipe_data nama_array[ukuran];
```

- tipe\_data: Menentukan tipe data elemen (misalnya, int, float).
- nama\_array: Nama variabel array.
- ukuran: Jumlah elemen maksimum dalam array

Contoh deklarasi array:

```
int angka[5];
```

Elemen dalam array dapat diakses menggunakan indeks, dimulai dari 0. Misalnya, angka mengacu pada elemen pertama.

- **Pointer**

Pointer adalah variabel yang menyimpan alamat memori dari variabel lain. Dalam C++, pointer dideklarasikan menggunakan simbol \*:

```
tipe_data *nama_pointer;
```

Pointer memungkinkan akses langsung ke lokasi memori dan dapat digunakan untuk manipulasi data. Contoh penggunaan pointer:

```
int var = 10;  
int* ptr = &var;
```

Untuk mengakses nilai yang ditunjuk oleh pointer, digunakan operator dereference \*:

```
cout << *ptr
```

- **Fungsi**

Fungsi dalam C++ adalah blok kode yang dapat dipanggil untuk melakukan tugas tertentu. Fungsi dideklarasikan dengan sintaks:

```
tipe_data nama_fungsi(parameter)  
{  
}
```

Fungsi dapat menerima parameter dan mengembalikan nilai. Contoh fungsi sederhana:

```
int tambah(int a, int b) {  
    return a + b;  
}
```

- **Prosedur**

Prosedur mirip dengan fungsi tetapi tidak mengembalikan nilai. Prosedur digunakan untuk menjalankan serangkaian perintah tanpa memberikan hasil kembali. Dalam C++, prosedur dideklarasikan seperti fungsi tetapi tanpa tipe pengembalian:

```
void tampilkanPesan() {  
    cout << "Hello, World!";  
}
```

Penggunaan prosedur membantu menyederhanakan kode dan meningkatkan keterbacaan dengan membagi program menjadi bagian-bagian kecil yang lebih mudah dikelola.

## Guided

1. Array

Input :

```
//Array satu dimensi  
int nilai[5]= {1,2,3,4,5};  
cout << nilai[0];  
cout << nilai[1];  
cout << nilai[2];  
cout << nilai[3];  
cout << nilai[4];  
  
int nilai[5]= {1,2,3,4,5};  
for(int i=0; i<5; i++){  
    cout << nilai[i] << endl;  
}
```

Output :

```
--interpreter=444  
1  
2  
3  
4  
5  
PS E:\Struktur Data\2211104058 Ganesha Rahman Gibran SE-06-02>
```

Input :

```
//Array dua dimensi
int nilai [3][4] = {
    {1,2,3,4},
    {5,6,7,8},
    {9,10,11,12}
};

for(int i=0; i<3; i++){
    for(int j=0; j<4; j++){
        cout << nilai[i][j] << " ";
    }
    cout << endl;
}
}
```

Output :

```
--interpreter=mi
1 2 3 4
5 6 7 8
9 10 11 12
PS E:\Struktur Data\2211104058_Ganesha_Rahman_Gibran_SE-06-02> ^
```

## 2. Pointer

Input :

```
int main(){
//Pointer
    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 yang ditunjuk px= " << *px << endl;
    cout << "Nilai Y= " << y << endl;
    getch();
    return 0;
}
```

Output :

```
'--interpreter=mi'
Alamat x= 0x5ffe90
Isi px= 0x5ffe90
Isi x= 87
Nilai yang ditunjuk px= 87
Nilai Y= 87
```

### 3. Fungsi

Input :

```
#include <iostream>
#include <conio.h>
#include <stdlib.h>

using namespace std;

//Fungsi
int penjumlahan(int a, int b){
    return a+b;
}

int main(){
    int hasil = penjumlahan(5,3);

    cout << "hasil " << hasil << endl;
}
```

Output :

```
--interpreter=mi'
hasil 8
PS E:\Struktur Data\2211104058_Ganesha_Rahman_Gibran_SE-06-02>
```

### 4. Prosedur

Input :

```
#include <iostream>
#include <conio.h>
#include <stdlib.h>

using namespace std;

//Fungsi
int penjumlahan(int a, int b){
    return a+b;
}

//Prosedur
void greet (string name){
    cout << "Hello, " << name << "!" << endl;
}

int main(){
    int hasil = penjumlahan(5,3);

    cout << "hasil " << hasil << endl;

    greet("alice");
}
```

Output :

```
--interpreter=mi'
hasil 8
Hello, alice!
PS E:\Struktur Data\2211104058_Ganesha_Rahman_Gibran_SE-06-02>
```

### Unguided

1. Buatlah program untuk menampilkan Output seperti berikut dengan data yang diinputkan oleh user!

Contoh output :

```
Data Array : 1 2 3 4 5 6 7 8 9 10
Nomor Genap : 2, 4, 6, 8, 10,
Nomor Ganjil : 1, 3, 5, 7, 9,
```

Input :

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

int main() {
    string input;
    vector<int> numbers;

    cout << "Masukkan angka (pisahkan dengan spasi): ";
    getline(cin, input);

    stringstream ss(input);
    int number;
    while (ss >> number) {
        numbers.push_back(number);
        if (ss.peek() == ' ') {
            ss.ignore();
        }
    }

    cout << "Angka yang anda masukkan: ";
    for (size_t i = 0; i < numbers.size(); i++) {
        cout << numbers[i];
        if (i < numbers.size() - 1) {
            cout << ", ";
        }
    }
    cout << endl;
```

```
cout << "Nomor Genap: ";
bool firstEven = true;
for (int num : numbers) {
    if (num % 2 == 0) {
        if (!firstEven) {
            cout << ",";
        }
        cout << num;
        firstEven = false;
    }
}
cout << endl;

cout << "Nomor Ganjil: ";
bool firstOdd = true;
for (int num : numbers) {
    if (num % 2 != 0) {
        if (!firstOdd) {
            cout << ",";
        }
        cout << num;
        firstOdd = false;
    }
}
cout << endl;

return 0;
```

Output :

```
Masukkan angka (pisahkan dengan spasi): 1 2 3 4 5 6 7 8 9 10
Angka yang anda masukkan: 1,2,3,4,5,6,7,8,9,10
Nomor Genap: 2,4,6,8,10
Nomor Ganjil: 1,3,5,7,9
PS E:\Struktur Data\2211104058_Ganesha_Rahman_Gibran_SE-06-02>
```

2. Buatlah program Input array tiga dimensi tetapi jumlah atau ukuran elemennya diinputkan oleh user!

Input :

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

using namespace std;

void inputArrayTigaDimensi() {
    int x, y, z;

    cout << "Masukkan ukuran pertama (x): ";
    cin >> x;
    cout << "Masukkan ukuran kedua (y): ";
    cin >> y;
    cout << "Masukkan ukuran ketiga (z): ";
    cin >> z;

    vector<vector<vector<int>>> arrayTigaDimensi(x, vector<vector<int>>(y, vector<int>(z)));

    for (int i = 0; i < x; ++i) {
        for (int j = 0; j < y; ++j) {
            for (int k = 0; k < z; ++k) {
                cout << "Masukkan elemen untuk [" << i << "]" << j << "]" << k << "]: ";
                cin >> arrayTigaDimensi[i][j][k];
            }
        }
    }

    cout << "\nArray Tiga Dimensi yang Dimasukkan:\n";
    for (int i = 0; i < x; ++i) {
        cout << "Lapisan " << i << ":\n";
        for (int j = 0; j < y; ++j) {
            for (int k = 0; k < z; ++k) {
                cout << arrayTigaDimensi[i][j][k] << " ";
            }
            cout << endl;
        }
        cout << endl;
    }

    int main() {
        inputArrayTigaDimensi();
        return 0;
    }
```

Output :



```
--interpreter=ml
Masukkan ukuran pertama (x): 1
Masukkan ukuran kedua (y): 2
Masukkan ukuran ketiga (z): 3
Masukkan elemen untuk [0][0][0]: 3
Masukkan elemen untuk [0][0][1]: 2
Masukkan elemen untuk [0][0][2]: 1
Masukkan elemen untuk [0][1][0]: 1
Masukkan elemen untuk [0][1][1]: 2
Masukkan elemen untuk [0][1][2]: 3
```

```
Array Tiga Dimensi yang Dimasukkan:
Lapisan 0:
3 2 1
1 2 3
```

```
PS E:\Struktur Data\2211104058_Ganesha_Rahman_Gibran_SE-06-02>
```

3. Buatlah program menu untuk mencari nilai Maksimum, Minimum dan Nilai rata – rata dari suatu array dengan input yang dimasukan oleh user!

Input :

```
#include <iostream>
#include <vector>
#include <limits>

using namespace std;

int cariMaksimum(const vector<int>& arr) {
    int maksimum = numeric_limits<int>::min();
    for (int num : arr) {
        if (num > maksimum) {
            maksimum = num;
        }
    }
    return maksimum;
}

int cariMinimum(const vector<int>& arr) {
    int minimum = numeric_limits<int>::max();
    for (int num : arr) {
        if (num < minimum) {
            minimum = num;
        }
    }
    return minimum;
}

double hitungRataRata(const vector<int>& arr) {
    if (arr.empty()) return 0.0;

    double total = 0;
    for (int num : arr) {
        total += num;
    }
    return total / arr.size();
}
```

```
}

int main() {
    int n;

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

    vector<int> arr(n);

    cout << "Masukkan " << n << " elemen array:\n";
    for (int i = 0; i < n; ++i) {
        cout << "Elemen [" << i + 1 << "]: ";
        cin >> arr[i];
    }

    int pilihan;

    do {
        cout << "\nMenu:\n";
        cout << "1. Cari Nilai Maksimum\n";
        cout << "2. Cari Nilai Minimum\n";
        cout << "3. Hitung Rata-Rata\n";
        cout << "4. Keluar\n";
        cout << "Pilih opsi (1-4): ";
        cin >> pilihan;

        do {
            cout << "\nMenu:\n";
            cout << "1. Cari Nilai Maksimum\n";
            cout << "2. Cari Nilai Minimum\n";
            cout << "3. Hitung Rata-Rata\n";
            cout << "4. Keluar\n";
            cout << "Pilih opsi (1-4): ";
            cin >> pilihan;

            switch (pilihan) {
                case 1:
                    cout << "Nilai Maksimum: " << cariMaksimum(arr) << endl;
                    break;
                case 2:
                    cout << "Nilai Minimum: " << cariMinimum(arr) << endl;
                    break;
                case 3:
                    cout << "Rata-Rata: " << hitungRataRata(arr) << endl;
                    break;
                case 4:
                    cout << "Keluar dari program.\n";
                    break;
                default:
                    cout << "Pilihan tidak valid! Silakan coba lagi.\n";
                    break;
            }
        } while (pilihan != 4);

        return 0;
    }
```

Output :

```
--interpreter=md
Masukkan jumlah elemen array: 3
Masukkan 3 elemen array:
Elemen [1]: 2
Elemen [2]: 4
Elemen [3]: 8

Menu:
1. Cari Nilai Maksimum
2. Cari Nilai Minimum
3. Hitung Rata-Rata
4. Keluar
Pilih opsi (1-4): 1
Nilai Maksimum: 8

Menu:
1. Cari Nilai Maksimum
2. Cari Nilai Minimum
3. Hitung Rata-Rata
4. Keluar
Pilih opsi (1-4): 2
Nilai Minimum: 2

Menu:
1. Cari Nilai Maksimum
2. Cari Nilai Minimum
3. Hitung Rata-Rata
4. Keluar
Pilih opsi (1-4): 3
Rata-Rata: 4.66667

Menu:
1. Cari Nilai Maksimum
2. Cari Nilai Minimum
3. Hitung Rata-Rata
4. Keluar
Pilih opsi (1-4): 4
Keluar dari program.
PS E:\Struktur Data\2211104058_Ganesha_Rahman_Gibran_SE-06-02>
```

## Kesimpulan

pemrograman C++ seperti Array, Pointer, Fungsi, dan Prosedur. Array digunakan untuk menyimpan elemen-elemen bertipe sama secara berurutan, sementara Pointer memungkinkan akses langsung ke alamat memori variabel lain. Fungsi memungkinkan pemrograman modular dengan mengelompokkan kode yang bisa dipanggil berulang kali, sedangkan Prosedur (fungsi dengan tipe pengembalian void) menjalankan tugas tanpa mengembalikan nilai. Pemahaman akan konsep-konsep ini penting dalam menulis program yang lebih efisien, terstruktur, dan mudah dikelola.