

**Laporan praktikum**  
**Modul 2**  
**Pengenalan CPP bagian 2**



**Disusun Oleh:**  
**Dwi Candra Pratama – 2211104035**  
**SE-06-02**

**Dosen: Wahyu Andi**  
**FAKULTAS INFORMATIKA PROGRAM STUDI S1 REKAYASA**  
**PERANGKAT LUNAK UNIVERSITAS TELKOM**  
**PURWOKERTO**

## Guided

### 1. TUJUAN

- Mengimplementasikan fungsi dan prosedur dengan parameter call by value, call by pointer, dan call by reference.
- Memahami konsep dasar dan penerapan array (satu dimensi, dua dimensi, dan multi dimensi), pointer, dan Fungsi dalam C++.
- Mengelola memori secara efisien menggunakan pointer dan array.
- Mengaplikasikan konsep array dan pointer dalam membangun program yang terstruktur dan modular.
- Menguasai penggunaan pointer untuk memanipulasi alamat memori variabel.

#### a) Array

Array merupakan kumpulan data dengan nama yang sama dan setiap elemen bertipe data sama. Untuk mengakses setiap komponen / elemen array berdasarkan indeks dari setiap elemen.

- Array Satu Dimensi

Adalah array yang hanya terdiri dari satu larik data saja. Contohnya seperti ini:

```
1  #include <iostream>
2  using namespace std;
3
4  Tabnine | Edit | Test | Explain | Document | Ask
5  int main() {
6      // ARRAY 1D
7      int nilai[10] = {1, 2, 3, 4, 5};
8
9      // Pemanggilan Array menggunakan index
10     cout << nilai[0] << endl;
11     cout << nilai[1] << endl;
12     cout << nilai[2] << endl;
13     cout << nilai[3] << endl;
14     cout << nilai[4] << endl;
15
16     // Pemanggilan array menggunakan Looping
17     for (int i = 0; i < 5; i++) {
18         cout << nilai[i] << endl;
19     }
20     return 0;
21 }
```

```
1
2
3
4
5
1
2
3
4
5
PS C:\Praktikum Struktur Data\Pertemuan 2>
```

Terdapat 2 versi pemanggilan index dan looping alhasil outputnya double

- Array Dua Dimensi dan dimensi banyak

Array 2D adalah jenis array yang memiliki dua indeks, yaitu satu untuk baris dan satu untuk kolom, sehingga dapat direpresentasikan sebagai tabel atau matriks. Array ini memungkinkan penyimpanan data dalam bentuk grid, seperti contoh matriks 3x3 yang menampung angka dalam tiga baris dan tiga kolom. Sementara itu, array multidimensi adalah ekstensi dari array 2D, di mana lebih dari dua indeks digunakan untuk mengakses elemen, seperti array 3D yang menambahkan dimensi ekstra dan sering digunakan dalam konteks yang lebih kompleks, seperti representasi data ruang tiga dimensi. Array multidimensi memungkinkan penyimpanan dan pengelolaan data dalam lebih dari dua dimensi, sesuai dengan kebutuhan sistem yang lebih rumit, seperti grafik, geometri ruang, atau model data berbasis dimensi tinggi.

```
#include <iostream>
using namespace std;
// Array 2 dimensi

Tabnine | Edit | Test | Explain | Document | Ask
int main() {
    // Array 2D
    int data_nilai[3][4] = {{1, 2, 3, 4}, {4, 6, 7, 8}, {9, 10, 11, 12}};

    // Nested loops to print the array
    for (int i = 0; i < 3; i++) {
        for (int j = 0; j < 4; j++) {
            cout << data_nilai[i][j] << " ";
        }
        cout << endl;
    }

    return 0;
}
```

```
1 2 3 4
4 6 7 8
9 10 11 12
PS C:\Praktikum Struktur Data\Pertemuan 2> |
```

#### b) Pointer

Pointer adalah sebuah variabel dalam pemrograman yang menyimpan alamat memori dari variabel lain. Daripada menyimpan nilai langsung, pointer menyimpan referensi ke lokasi memori di mana nilai tersebut disimpan. Pointer sangat berguna dalam pengelolaan memori, manipulasi array, dan pengoperasian fungsi, karena mereka memungkinkan akses langsung ke alamat memori, memungkinkan kita untuk mengubah nilai yang berada di luar cakupan fungsi atau struktur data secara efisien.

```

#include <iostream>
using namespace std;

Tabnine | Edit | Test | Explain | Document | Ask
int main() {
    int x, y;
    int *px;

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

    cout << "Alamat x: " << &x << endl;
    cout << "Isi px (alamat x): " << px << endl;
    cout << "Isi x: " << x << endl;
    cout << "Nilai yang ditunjuk px (nilai x): " << *px << endl;
    cout << "Nilai y: " << y << endl;

    return 0;
}

```

```

Alamat x: 0xab47ff9c0
Isi px (alamat x): 0xab47ff9c0
Isi x: 87
Nilai yang ditunjuk px (nilai x): 87
Nilai y: 87
PS C:\Praktikum Struktur Data\Pertemuan 2>

```

## UNGUIDED

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

```

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,

```

```

#include <iostream>
using namespace std;

Tabnine | Edit | Test | Explain | Document | Ask

int main() {
    int n;

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

    int arr[n];

    // Input data array dari user
    cout << "Masukkan " << n << " angka: ";
    for (int i = 0; i < n; i++) {
        cin >> arr[i];
    }

    // Menampilkan seluruh data array
    cout << "Data Array : ";
    for (int i = 0; i < n; i++) {
        cout << arr[i];
        if (i != n - 1) {
            cout << ", "; // Tambahkan koma setelah setiap elemen kecuali elemen terakhir
        }
    }
    cout << endl;

    // Menampilkan angka genap
    cout << "Nomor Genap : ";
    bool foundEven = false;
    for (int i = 0; i < n; i++) {
        if (arr[i] % 2 == 0) {
            if (foundEven) {
                cout << ", "; // Tambahkan koma setelah setiap elemen genap kecuali elemen pertama
            }
            cout << arr[i];
            foundEven = true;
        }
    }
    if (!foundEven) {
        cout << "-"; // Jika tidak ada bilangan genap
    }
    cout << endl;

    // Menampilkan angka ganjil
    cout << "Nomor Ganjil : ";
    bool foundOdd = false;
    for (int i = 0; i < n; i++) {
        if (arr[i] % 2 != 0) {
            if (foundOdd) {
                cout << ", "; // Tambahkan koma setelah setiap elemen ganjil kecuali elemen pertama
            }
            cout << arr[i];
            foundOdd = true;
        }
    }
    if (!foundOdd) {
        cout << "-"; // Jika tidak ada bilangan ganjil
    }
    cout << endl;

    return 0;
}

```

```

Masukkan jumlah elemen array: 10
Masukkan 10 angka: 1 2 3 4 5 6 7 8 9 10
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
PS C:\Praktikum Struktur Data\Pertemuan 2\UNGUIDED>

```

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

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

Tabnine | Edit | Test | Explain | Document | Ask
int main() {
    int x, y, z;

    // Input ukuran array 3D dari user
    cout << "Masukkan ukuran dimensi pertama (x): ";
    cin >> x;
    cout << "Masukkan ukuran dimensi kedua (y): ";
    cin >> y;
    cout << "Masukkan ukuran dimensi ketiga (z): ";
    cin >> z;

    int arr[x][y][z];

    // Input elemen array 3D
    cout << "Masukkan elemen array 3D: " << endl;
    for (int i = 0; i < x; i++) {
        for (int j = 0; j < y; j++) {
            for (int k = 0; k < z; k++) {
                cout << "Elemen [" << i << "][" << j << "][" << k << "]: ";
                cin >> arr[i][j][k];
            }
        }
    }

    // Menampilkan array 3D
    cout << "Array 3D: " << endl;
    for (int i = 0; i < x; i++) {
        for (int j = 0; j < y; j++) {
            for (int k = 0; k < z; k++) {
                cout << arr[i][j][k] << " ";
            }
            cout << endl;
        }
        cout << endl;
    }

    return 0;
}
```

```
Masukkan ukuran dimensi pertama (x): 2
Masukkan ukuran dimensi kedua (y): 2
Masukkan ukuran dimensi ketiga (z): 2
Masukkan elemen array 3D:
Elemen [0][0][0]: 1
Elemen [0][0][1]: 2
Elemen [0][1][0]: 3
Elemen [0][1][1]: 4
Elemen [1][0][0]: 5
Elemen [1][0][1]: 6
Elemen [1][1][0]: 7
Elemen [1][1][1]: 8
Array 3D:
1 2
3 4

5 6
7 8

PS C:\Praktikum Struktur Data\Pertemuan 2\UNGUIDED>
```

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

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

Tabnine | Edit | Test | Explain | Document | Ask

```
✓ int main() {
    int n;

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

    int arr[n];

    // Input data array dari user
    cout << "Masukkan " << n << " angka: ";
    ✓ for (int i = 0; i < n; i++) {
        cin >> arr[i];
    }

    // Inisialisasi nilai maksimum dan minimum
    int maks = arr[0];
    int min = arr[0];
    int total = 0;

    // Mencari nilai maksimum, minimum, dan rata-rata
    ✓ for (int i = 0; i < n; i++) {
    ✓     if (arr[i] > maks) {
        maks = arr[i];
    }
    ✓     if (arr[i] < min) {
        min = arr[i];
    }
    total += arr[i];
    }
    float rata = (float)total / n;

    // Output hasil
    cout << "Nilai Maksimum: " << maks << endl;
    cout << "Nilai Minimum: " << min << endl;
    cout << "Nilai Rata-rata: " << rata << endl;

    return 0;
}
```

```
Masukkan jumlah elemen array: 4  
Masukkan 4 angka: 1 2 3 4  
Nilai Maksimum: 4  
Nilai Minimum: 1  
Nilai Rata-rata: 2.5  
PS C:\Praktikum Struktur Data\Pertemuan 2\UNGUIDED>
```