

# **LAPORAN PRAKTIKUM**

## **MODUL II “Array Multidimesi”**



**Disusun oleh:**

Shiva Indah Kurnia  
NIM 2311102035

**Dosen Pengampu:**

Wahyu Andi Saputra, S.Pd., M.Eng

**PROGRAM STUDI S1 TEKNIK INFORMATIKA  
FAKULTAS INFORMATIKA  
INSTITUT TEKNOLOGI TELKOM PURWOKERTO  
PURWOKERTO  
2024**

## **TUJUAN PRAKTIKUM**

Setelah melakukan praktikum ini diharapkan mahasiswa dapat:

1. Memahami konsep dasar array 1 dimensi, array 2 dimensi dan yang terfokus array multidimensi.
2. Mampu membuat program dengan mengaplikasikan array multidimensi.
3. Mampu menerapkan konsep array multidimesi dalam studi kasus nyata.

## DASAR TEORI

Array adalah kumpulan data yang bertipe sama yang menggunakan sama. Dengan menggunakan array, sejumlah variabel dapat memakai nama yang sama. Antara satu variabel dengan variabel lain di dalam array dibedakan berdasarkan nomor elemen (subscript/indeks) Variabel dapat dideklarasikan ber-type array dari suatu type tertentu. Setiap elemen array diakses dengan alamat berupa indeks yang bertipe integer.

**Contoh -> int TabInt[10];**

**Keterangan:**

- Array bernama **TabInt**
- Setiap elemen bertipe **integer**
- Dengan ukuran **10 elemen**
- Dengan alamat setiap elemen array (indeks) adalah dari indeks **ke-0 s.d. 9**

Anggap array sebagai sebuah deretan rumah pada sebuah jalan, dan rumah-rumah tersebut memiliki bentuk yang sama. Untuk membedakan rumah yang satu dengan yang lain maka dibutuhkan sebuah nomor. Nomor itulah yang kemudian kita kenal sebagai index.

Array dapat digolongkan menjadi:

- a. Array berdimensi satu
- b. Array berdimensi dua
- c. Array berdimensi banyak

Pendeklarasian array diawali dengan penyebutan **tipe data diikuti nama variabel array, diikuti jumlah elemen**. Jika array hendak diberi nilai awal, nilai-nilai awal dituliskan dalam inisialisasi. Jika inisialisasi kurang dari jumlah elemen array, sisanya akan diinisialisasikan dengan nol.

Elemen array diakses dengan menyebutkan indeks posisi elemen tersebut dalam array. Elemen pertama array memiliki indeks 0.

Dikarenakan berfokus kepada array multidimensi, array multidimensi sendiri terdiri dari indeks representatif indeks yang mewakili baris dan kolom. Indeks di sebelah kiri adalah ukurannya jumlah baris dalam tabel, sedangkan indeks yang tepat adalah ukurannya jumlah kolom dalam tabel. Untuk array tiga dimensi, index berikutnya adalah indeks yang mewakili kolom dan baris yang diukur kembali atau sesuatu.

## GUIDED

### 1. Guided 1

#### Source Code

```
#include <iostream>
using namespace std;
int main()
{
    int arr[2][3][3];

    for (int x = 0; x < 2; x++)
    {
        for (int y = 0; y < 3; y++)
        {
            for (int z = 0; z < 3; z++)
            {
                cout << "Input Array[" << x << "][" << y << "]["
<< z <<
                "]" = ";
                cin >> arr[x][y][z];
            }
            cout << endl;
        }
        // Output Array
        for (int x = 0; x < 2; x++)
        {
            for (int y = 0; y < 3; y++)
            {
                for (int z = 0; z < 3; z++)
                {
                    cout << "Data Array[" << x << "][" << y << "]["
<< z <<
                    "]" = " << arr[x][y][z] << endl;
                }
            }
        }
        cout << endl;
        // Tampilan array
        for (int x = 0; x < 2; x++)
        {
            for (int y = 0; y < 3; y++)
            {
```

```

        for (int z = 0; z < 3; z++)
        {
            cout << arr[x][y][z] << ends;
        }
        cout << endl;
    }
    cout << endl;
}
}

```

## Screenshot Program

```

g++ array3.cpp
PS C:\Laparak Sem 2 Strukdat\Guided 1> ./a
Input Array[0][0][0] = 2
Input Array[0][0][1] = 3
Input Array[0][0][2] = 4
Input Array[0][1][0] = 5
Input Array[0][1][1] = 6
Input Array[0][1][2] = 7
Input Array[0][2][0] = 3
Input Array[0][2][1] = 4
Input Array[0][2][2] = 5

Input Array[1][0][0] = 6
Input Array[1][0][1] = 7
Input Array[1][0][2] = 8
Input Array[1][1][0] = 3
Input Array[1][1][1] = 4
Input Array[1][1][2] = 9
Input Array[1][2][0] = 2
Input Array[1][2][1] = 3
Input Array[1][2][2] = 4

Data Array[0][0][0] = 2
Data Array[0][0][1] = 3
Data Array[0][0][2] = 4
Data Array[0][1][0] = 5
Data Array[0][1][1] = 6
Data Array[0][1][2] = 7
Data Array[0][2][0] = 3
Data Array[0][2][1] = 4
Data Array[0][2][2] = 5
Data Array[1][0][0] = 6
Data Array[1][0][1] = 7
Data Array[1][0][2] = 8
Data Array[1][1][0] = 3
Data Array[1][1][1] = 4
Data Array[1][1][2] = 9
Data Array[1][2][0] = 2
Data Array[1][2][1] = 3

```

```
Data Array[1][1][1] = 4  
Data Array[1][1][2] = 9  
Data Array[1][2][0] = 2  
Data Array[1][2][1] = 3  
Data Array[1][2][2] = 4
```

```
234  
567  
345
```

```
678  
349  
234
```

### Deskripsi Program

Pertama, pada program telah dideklarasikan array tiga dimensi dengan tipe data integer ukuran masing-masing dimensi 2, 3, 3. Kemudian, program melakukan input elemen array dengan tiga pengulangan for. Perulangan pertama akan mengakses elemen pada dimensi pertama begitu seterusnya sampai dengan dimensi ketiga. Selama perulangan, program akan menampilkan input array `[x][y][z]` dan meminta pengguna untuk memasukkan nilai elemen array yang kemudian disimpan ke array di indeks yang sesuai. Setelah itu, cetak array menggunakan tiga loop for yang mirip dengan loop fase input. Selama perulangan, program menampilkan pesan `"Data Array[x][y][z] ="` dan nilai elemen array pada indeks yang sesuai. Terakhir, program akan menampilkan tabel dengan tiga loop yang mirip dengan loop input dan output. Selama perulangan, program hanya menampilkan nilai elemen array pada indeks yang sesuai tanpa pesan tambahan. Program diakhiri dengan `return 0`.

## 2. Guided 2

### Source Code

```
#include <iostream>
using namespace std;
int main()
{
    int maks, a, i = 1, lokasi;
    cout << "Masukkan panjang array: ";
    cin >> a;
    int array[a];
    cout << "Masukkan " << a << " angka\n";
    for (i = 0; i < a; i++)
    {
        cout << "Array ke-" << (i) << ": ";
        cin >> array[i];
    }
    maks = array[0];
    for (i = 0; i < a; i++)
    {
        if (array[i] > maks)
        {
            maks = array[i];
            lokasi = i;
        }
    }
    cout << "Nilai maksimum adalah " << maks << " berada di
    Array ke " << lokasi << endl;
}
```

## Screenshot Program

```
array4.cpp
PS C:\Laprak Sem 2 Strukdat\Guided 1> ./a
Masukkan panjang array: 5
Masukkan 5 angka
Array ke-0: 2
Array ke-1: 5
Array ke-2: 7
Array ke-3: 8
Array ke-4: 3
Nilai maksimum adalah 8 berada di Array ke 3
PS C:\Laprak Sem 2 Strukdat\Guided 1> 
```

## Deskripsi Program

Pertama, program akan meminta pengguna untuk memasukan panjang array dan menyimpan input pada variabel a. Setelah itu, program akan membuat array dengan panjang a dengan menggunakan deklarasi array dinamis. Selanjutnya, program akan meminta pengguna untuk memasukan nilai di setiap elemen array dan menggunakan pengulangan for untuk meminta input nilai sebanyak a. Setelah semua nilai elemen array dimasukkan, program akan mencari nilai maksimum dari bilangan yang telah dimasukkan menggunakan pengulangan for. Program akan membandingkan nilai array pada indeks ke-l dengan nilai maksimum pada variabel maksimum. Setelah nilai maksimum diperoleh, program akan berakhir dan mengembalikan nilai 0.



## UNGUIDED

### 1. Unguided 1

#### Source Code

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

int main()
{
    int bilangan[10];
    cout << "Data Array : ";
    for (int i = 0; i < 10; i++)
    {
        cin >> bilangan[i];
    }
    cout << "Nomor Genap : ";
    for (int i = 0; i < 10; i++)
    {
        if (bilangan[i] % 2 == 0)
        {
            cout << bilangan[i] << " ";
        }
    }
    cout << endl;
    cout << "Nomor Ganjil : ";
    for (int i = 0; i < 10; i++)
    {
        if (bilangan[i] % 2 != 0)
        {
            cout << bilangan[i] << " ";
        }
    }

    return 0;
}
```

## Screenshot Program

```
g++ arrayunguided1.cpp
PS C:\Laprak Sem 2 Strukdat\Guided 1> ./a
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:\Laprak Sem 2 Strukdat\Guided 1> |
```

## Deskripsi Program

Pertama, program akan meminta pengguna untuk memasukan 10 bilangan kedalam array bertipe integer. Setelah itu, program akan menggunakan pengulangan for untuk memasukan bilangan kedalam array. Pada program tersebut juga menggunakan percabangan if untuk menentukan ganjil genapnya suatu bilangan.dengan menggunakan loop for dan percabangan yang sama. Setelah menampilkan bilangan ganjil dan genap, Program akan berakhir dan mengembalikan nilai 0.

## 2. Unguided 2

### Source Code

```
#include <iostream>
using namespace std;
int main()
{
    int x, y, z;
    cout << "Masukkan jumlah elemen matriks: ";
    cin >> x;
    cout << "Masukkan ukuran matriks (y z): ";
    cin >> y >> z;
    int arr[x][y][z];

    // Input elemen
    for (int i = 0; i < x; i++)
    {
        for (int j = 0; j < y; j++)
        {
            for (int k = 0; k < z; k++)
            {
```

```

        cout << "Input Array[" << i << "][" << j << "]["
<< k << "] = ";
        cin >> arr[i][j][k];
    }
}
cout << endl;
}
// Output Array
for (int i = 0; i < x; i++)
{
    for (int j = 0; j < y; j++)
    {
        for (int k = 0; k < z; k++)
        {
            cout << "Data Array[" << i << "][" << j << "]["
<< k << "] = " << arr[i][j][k] << endl;
        }
    }
}

cout << endl;
// Tampilan array
for (int i = 0; i < x; i++)
{
    cout << "Matriks ke-" << i + 1 << ":" << endl;
    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;
}

```

## Screenshot Program

```
PS C:\Laprak Sem 2 Strukdat\Guided 1> g++ arrayunguided2.cpp
Masukkan jumlah elemen matriks: 2
Masukkan ukuran matriks (y z): 3 3
Input Array[0][0][0] = 2
Input Array[0][0][1] = 3
Input Array[0][0][2] = 4
Input Array[0][1][0] = 5
Input Array[0][1][1] = 6
Input Array[0][1][2] = 7
Input Array[0][2][0] = 8
Input Array[0][2][1] = 9
Input Array[0][2][2] = 1

Input Array[1][0][0] = 5
Input Array[1][0][1] = 6
Input Array[1][0][2] = 7
Input Array[1][1][0] = 8
Input Array[1][1][1] = 8
Input Array[1][1][2] = 9
Input Array[1][2][0] = 0
Input Array[1][2][1] = 2
Input Array[1][2][2] = 2

Data Array[0][0][0] = 2
Data Array[0][0][1] = 3
Data Array[0][0][2] = 4
Data Array[0][1][0] = 5
Data Array[0][1][1] = 6
Data Array[0][1][2] = 7
Data Array[0][2][0] = 8
Data Array[0][2][1] = 9
Data Array[0][2][2] = 1
Data Array[1][0][0] = 5
Data Array[1][0][1] = 6
Data Array[1][0][2] = 7
Data Array[1][1][0] = 8
Data Array[1][1][1] = 8
Data Array[1][1][2] = 9
Data Array[1][2][0] = 0
```

```
Data Array[0][0][0] = 2
Data Array[0][0][1] = 3
Data Array[0][0][2] = 4
Data Array[0][1][0] = 5
Data Array[0][1][1] = 6
Data Array[0][1][2] = 7
Data Array[0][2][0] = 8
Data Array[0][2][1] = 9
Data Array[0][2][2] = 1
Data Array[1][0][0] = 5
Data Array[1][0][1] = 6
Data Array[1][0][2] = 7
Data Array[1][1][0] = 8
Data Array[1][1][1] = 8
Data Array[1][1][2] = 9
Data Array[1][2][0] = 0
Data Array[1][2][1] = 2
Data Array[1][2][2] = 2
```

Matriks ke-1:

```
2 3 4
5 6 7
8 9 1
```

Matriks ke-2:

```
5 6 7
8 8 9
0 2 2
```

PS C:\Laprak Sem 2 Strukdat\Guided 1> █

## Deskripsi Program

Program diatas merupakan program untuk membuat array 3 dimensi dengan menginput jumlah elemen, ukuran matriks, serta nilai elemen pada matriks sesuai kemauan pengguna. Pertama, deklarasikan variabel x, y, z. Kemudian, program akan meminta pengguna untuk menginput jumlah elemen dan ukuran matriks. Kemudian, program akan membuat array 3 dimensi sesuai hal yang diinputkan oleh pengguna. Setelah itu, program akan melakukan input nilai elemen pada matriks 3 dimensi dengan menggunakan looping for. Program akan melakukan output data pada array dengan looping for. Setelah itu, program akan menampilkan matriks dengan menggunakan looping for dan perulangan nested. Program berakhir dan mengembalikan nilai 0.

### 3. Unguided 3

#### Source Code

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

int main()
{
    int x;
    cout << "Masukkan jumlah elemen array yang anda inginkan: ";
    cin >> x;
    int arr[x];
    for (int i = 0; i < x; i++)
    {
        cout << "Input elemen array ke-" << i + 1 << ": ";
        cin >> arr[i];
    }
    int choose;
    if (true)
    {
        cout << endl;
        cout << "Pilih menu yang diinginkan: " << endl;
        cout << "1. Mencari nilai maksimum" << endl;
        cout << "2. Mencari nilai minimum" << endl;
        cout << "3. Mencari nilai rata-rata" << endl;
        cout << "Pilihan Anda: ";
        cin >> choose;
        switch (choose)
        {
            case 1:
            {
                int maks = arr[0];
                for (int i = 1; i < x; i++)
                {
                    if (arr[i] > maks)
                    {
                        maks = arr[i];
                    }
                }
                cout << "Nilai maksimum: " << maks << endl;
                break;
            }
            case 2:
```

```

    {
        int min = arr[0];
        for (int i = 1; i < x; i++)
        {
            if (arr[i] < min)
            {
                min = arr[i];
            }
        }
        cout << "Nilai minimum: " << min << endl;
        break;
    }
    case 3:
    {
        int sum = 0;
        for (int i = 0; i < x; i++)
        {
            sum += arr[i];
        }
        double ratarata = (double)sum / x;
        cout << "Nilai rata-rata: " << ratarata << endl;
        break;
    }
    default:
    {
        cout << "Pilihan yang anda masukkan tidak valid." <<
endl;
        break;
    }
}
}
return 0;
}

```

## Screenshot Program

\*hasil jika user menginputkan menu 1

```
PS C:\Laprak Sem 2 Strukdat\Guided 1> g++ arrayunguided3.cpp
PS C:\Laprak Sem 2 Strukdat\Guided 1> ./a
Masukkan jumlah elemen array yang anda inginkan: 4
Input elemen array ke-1: 5
Input elemen array ke-2: 6
Input elemen array ke-3: 1
Input elemen array ke-4: 2

Pilih menu yang diinginkan:
1. Mencari nilai maksimum
2. Mencari nilai minimum
3. Mencari nilai rata-rata
Pilihan Anda: 1
Nilai maksimum: 6
```

\*hasil jika user menginputkan menu 2

```
PS C:\Laprak Sem 2 Strukdat\Guided 1> g++ arrayunguided3.cpp
PS C:\Laprak Sem 2 Strukdat\Guided 1> ./a
Masukkan jumlah elemen array yang anda inginkan: 4
Input elemen array ke-1: 9
Input elemen array ke-2: 5
Input elemen array ke-3: 8
Input elemen array ke-4: 4

Pilih menu yang diinginkan:
1. Mencari nilai maksimum
2. Mencari nilai minimum
3. Mencari nilai rata-rata
Pilihan Anda: 2
Nilai minimum: 4
```

\*hasil jika user menginputkan menu 3

```
PS C:\Laprak Sem 2 Strukdat\Guided 1> g++ arrayunguided3.cpp
PS C:\Laprak Sem 2 Strukdat\Guided 1> ./a
Masukkan jumlah elemen array yang anda inginkan: 4
Input elemen array ke-1: 5
Input elemen array ke-2: 3
Input elemen array ke-3: 7
Input elemen array ke-4: 2
```

```
Pilih menu yang diinginkan:
1. Mencari nilai maksimum
2. Mencari nilai minimum
3. Mencari nilai rata-rata
Pilihan Anda: 3
Nilai rata-rata: 4.25
```

\*hasil jika user tidak menginputkan sesuai menu

```
Pilih menu yang diinginkan:
1. Mencari nilai maksimum
2. Mencari nilai minimum
3. Mencari nilai rata-rata
Pilihan Anda: 4
Pilihan yang anda masukkan tidak valid.
PS C:\Laprak Sem 2 Strukdat\Guided 1> |
```



## **Deskripsi Program**

Program diatas merupakan program yang dibuat untuk menentukan nilai maksimum, minimum, dan rata-rata pada data array yang diinputkan. Pertama, program akan meminta pengguna untuk memasukan jumlah elemen array yang diinginkan. Kemudian, program akan membuat array sebanyak yang diminta pengguna dan meminta pengguna untuk mengisi nilai di setiap elemennya. Program lalu akan menampilkan pilihan menu operasi yaitu, maksimum, minimum dan rata-rata. Pengguna akan menginputkan satu pilihan dari ketiganya. Kemudian, program akan melakukan operasi sesuai dengan pilihan user. Dalam pilihan menu, program dijalankan dengan switch dengan case and break pada setiap pilihannya. Hasil operasi akan ditampilkan di outputnya. Jika pilihan menu yang dimasukkan tidak sesuai, maka program akan memberi pesan bahwa angka yang dimasukkan tidak valid.

## KESIMPULAN

Deklarasi array dimulai dengan tipe data, diikuti dengan nama variabel array, dan kemudian jumlah elemen. Jika array harus diberi nilai awal, nilai awal ditulis ke penginisialisasi. Jika initializer kurang dari jumlah elemen dalam array, sisanya diinisialisasi ke nol. Array multidimensi dalam C++ adalah struktur data yang digunakan untuk menyimpan elemen dalam bentuk tabel atau array dengan dua ukuran atau lebih. Array ini dapat digunakan untuk menyimpan data yang lebih kompleks seperti gambar, teks atau data multidimensi. array multidimensi sendiri terdiri dari indeks representatif indeks yang mewakili baris dan kolom. Indeks di sebelah kiri adalah ukurannya jumlah baris dalam tabel, sedangkan indeks yang tepat adalah ukurannya jumlah kolom dalam tabel. Untuk array tiga dimensi, index berikutnya adalah indeks yang mewakili kolom dan baris yang diukur kembali atau sesuatu. Pada laporan praktikum kali ini, kita dapat belajar mengenai pembuatan program menu array, menentukan bilangan ganjil genap serta membuat matriks dengan menggunakan array multidimensi.