

# **LAPORAN PRAKTIKUM**

## **MODUL II ARRAY**



**Disusun oleh:**  
**Geranada Saputra Priambudi**  
**NIM: 2311102008**

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

**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS INFORMATIKA  
INSTITUT TEKNOLOGI TELKOM PURWOKERTO  
PURWOKERTO  
2023**

# **BAB I**

## **TUJUAN PRAKTIKUM**

- a. Mahasiswa dapat memahami konsep Array.
- b. Mahasiswa dapat mengetahui jenis dimensi Array dan cara penulisannya.
- c. Mahasiswa dapat mengimplementasikan Array pada kode program yang dibuat.

## **BAB II**

### **DASAR TEORI**

Array merupakan struktur data yang paling dasar. Sebelum mempelajari struktur data lainnya, alangkah baiknya, kita perlu mempelajari Array terlebih dahulu. Berikut ini adalah beberapa jenis array :

1. **Array Satu Dimensi** Array satu dimensi adalah tipe variabel yang terdiri dari kumpulan data dengan tipe yang sama yang disusun dalam satu baris atau satu dimensi. Setiap elemen di dalam array memiliki sebuah indeks atau nomor yang digunakan untuk mengakses elemen tersebut. Indeks dimulai dari 0 dan berakhir pada jumlah elemen dikurangi satu.

Contohnya, sebuah array satu dimensi yang berisi bilangan bulat {1, 2, 3, 4, 5} memiliki lima elemen dan indeksnya dimulai dari 0. Indeks 0 merujuk pada elemen pertama, indeks 1 merujuk pada elemen kedua, dan seterusnya hingga indeks 4 merujuk pada elemen kelima.

2. **Array Dua Dimensi** Array dua dimensi adalah variable yang terdiri dari kumpulan array satu dimensi dengan tipe yang sama yang disusun dalam baris dan kolom. Dalam array dua dimensi, setiap elemen memiliki dua indeks, yaitu indeks baris dan indeks kolom. Indeks baris menunjukkan posisi elemen dalam baris, sementara indeks kolom menunjukkan posisi elemen dalam kolom.
3. **Array Multidimensi** Array multidimensi memiliki kesamaan dengan array satu dimensi dan dua dimensi, namun memiliki kapasitas memori yang lebih besar. Array ini digunakan untuk merepresentasikan array dengan dimensi lebih dari dua atau array yang memiliki lebih dari dua indeks, seperti array tiga dimensi, array empat dimensi, array lima dimensi, dan seterusnya.

4. **Array Empat Dimensi**

Contoh :

```
int arr [3] [2] [4] [4];
```

5. **Array Lima Dimensi**

Contoh :

```
int arr [2] [4] [4] [3] [3];
```

## BAB III

### GUIDED

#### 1. Guided 1

##### Source code

```
#include <iostream>

using namespace std;

// PROGRAM INPUT ARRAY 3 DIMENSI
int main()
{
    // Deklarasi array
    int arr[2][3][3];
    // Input elemen
    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;
    }
}
```

**Screenshoot program**

```

2 (SI IF-11-A)\Praktikum Struktur data dan Algoritma\Modul2" ; if ($?) { g++ Guided_1.cpp -o Guided_1 } ; if ($?) { .\Guided_1 }
Input Array[0][0][0] = 1
Input Array[0][0][1] = 8
Input Array[0][0][2] = 4
Input Array[0][1][0] = 9
Input Array[0][1][1] = 3
Input Array[0][1][2] = 5
Input Array[0][2][0] = 8
Input Array[0][2][1] = 3
Input Array[0][2][2] = 4

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

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

184
935
834

739
561
583

```

## Deskripsi program

Program diatas merupakan program C++ yang memungkinkan pengguna untuk memasukkan elemen ke dalam array tiga dimensi dan kemudian menampilkan elemen-elemen tersebut. Berikut ini yang termasuk bagian dari program yaitu : Deklarasi Array, Input Elemen, Output Elemen dan Tampilan Array

## 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];
    }
}

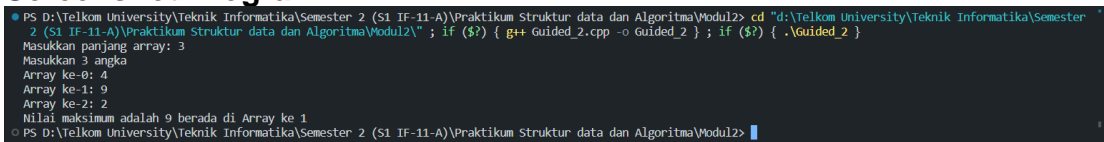
```

```

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



```

PS D:\Telkom University\Teknik Informatika\Semester 2 (S1 IF-11-A)\Praktikum Struktur data dan Algoritma\Modul2> cd "d:\Telkom University\Teknik Informatika\Semester
2 (S1 IF-11-A)\Praktikum Struktur data dan Algoritma\Modul2\" ; if ($?) { g++ Guided_2.cpp -o Guided_2 } ; if ($?) { .\Guided_2 }
Masukkan panjang array: 3
Masukkan 3 angka
Array ke-0: 4
Array ke-1: 9
Array ke-2: 2
Nilai maksimum adalah 9 berada di Array ke 1
PS D:\Telkom University\Teknik Informatika\Semester 2 (S1 IF-11-A)\Praktikum Struktur data dan Algoritma\Modul2>

```

### Deskripsi program

Program diatas merupakan program yang bertujuan untuk mencari nilai maksimum dari sejumlah angka yang dimasukkan pengguna ke dalam sebuah array. Program ini tidak memiliki perlindungan terhadap kasus-kasus tertentu, seperti jika pengguna memasukkan nilai tidak valid atau array kosong.

## LATIHAN KELAS - UNGUIDED

### 1. Unguided 1

#### Source code

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

using namespace std;

int main() {
    // Mendeklarasikan array dan variabel lainnya
    const int SIZE = 10;
    int numbers[SIZE];
    vector<int> oddNumbers, evenNumbers;

    // Meminta pengguna untuk memasukkan data array
    cout << "Masukkan 10 angka: ";
    for (int i = 0; i < SIZE; ++i) {
        cin >> numbers[i];
    }

    // Memisahkan angka ganjil dan genap ke dalam vektor yang sesuai
    for (int i = 0; i < SIZE; ++i) {
        if (numbers[i] % 2 == 0) {
            evenNumbers.push_back(numbers[i]);
        } else {
            oddNumbers.push_back(numbers[i]);
        }
    }

    // Menampilkan angka ganjil
    cout << "Nomor Ganjil: ";
```



```

        for (int i = 0; i < oddNumbers.size(); ++i) {
            cout << oddNumbers[i];
            if (i != oddNumbers.size() - 1) {
                cout << ", ";
            }
        }
        cout << endl;

        // Menampilkan angka genap
        cout << "Nomor Genap: ";
        for (int i = 0; i < evenNumbers.size(); ++i) {
            cout << evenNumbers[i];
            if (i != evenNumbers.size() - 1) {
                cout << ", ";
            }
        }
        cout << endl;

        return 0;
    }

```

## Screenshoot program

```

PS D:\Telkom University\Teknik Informatika\Semester 2 (S1 IF-11-A)\Praktikum Struktur data dan Algoritma\Modul2> cd "d:\Telkom University\Teknik Informatika\Semester 2 (S1 IF-11-A)\Praktikum Struktur data dan Algoritma\Modul2"; if ($?) { g++ Unguided_1.cpp -o Unguided_1 }; if ($?) { .\Unguided_1 }
Masukkan 10 angka: 12
43
57
90
52
31
99
12
36
72
Nomor Ganjil: 43, 57, 31, 99
Nomor Genap: 12, 90, 52, 12, 36, 72
PS D:\Telkom University\Teknik Informatika\Semester 2 (S1 IF-11-A)\Praktikum Struktur data dan Algoritma\Modul2>

```

## Deskripsi program

Program diatas merupakan program yang meminta pengguna untuk memasukkan 10 angka ke dalam array, kemudian memisahkan angka-angka ganjil dan genap ke dalam vector yang sesuai, dan terakhir menampilkan angka ganjil dan genap tersebut

## 2. Unguided 2

### Source code

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

int main() {
    int x_size, y_size, z_size;

    // Meminta pengguna untuk memasukkan ukuran array
    cout << "Masukkan ukuran array tiga dimensi:\n";
    cout << "Ukuran dimensi x: ";
    cin >> x_size;
    cout << "Ukuran dimensi y: ";
    cin >> y_size;
    cout << "Ukuran dimensi z: ";
    cin >> z_size;

    // Deklarasi array
    int arr[x_size][y_size][z_size];

    // Input elemen
    cout << "Masukkan elemen array:\n";
    for (int x = 0; x < x_size; x++) {
        for (int y = 0; y < y_size; y++) {
            for (int z = 0; z < z_size; z++) {
                cout << "Input Array[" << x << "][" << y << "][" << z << "] = ";
                cin >> arr[x][y][z];
            }
        }
    }

    // Output Array
    cout << "\nData Array:\n";
```

```

for (int x = 0; x < x_size; x++) {
    for (int y = 0; y < y_size; y++) {
        for (int z = 0; z < z_size; z++) {
            cout << "Data Array[" << x << "][" << y << "]["
<< z << "] = " << arr[x][y][z] << endl;
        }
    }
}

// Tampilan array
cout << "\nTampilan Array:\n";
for (int x = 0; x < x_size; x++) {
    for (int y = 0; y < y_size; y++) {
        for (int z = 0; z < z_size; z++) {
            cout << arr[x][y][z] << " ";
        }
        cout << endl;
    }
    cout << endl;
}

return 0;
}

```

## Screenshot program

```

PS D:\Telkom University\Teknik Informatika\Semester 2 (S1 IF-11-A)\Praktikum Struktur data dan Algoritma\Modul2> cd "d:\Telkom University\Teknik Informatika\Semester
2 (S1 IF-11-A)\Praktikum Struktur data dan Algoritma\Modul2\"; if ($?) { g++ Unguided_2.cpp -o Unguided_2 }; if ($?) { .\Unguided_2 }
Masukkan ukuran array tiga dimensi:
Ukuran dimensi x: 3
Ukuran dimensi y: 3
Ukuran dimensi z: 3
Masukkan elemen array:
Input Array[0][0][0] = 6
Input Array[0][0][1] = 4
Input Array[0][0][2] = 2
Input Array[0][1][0] = 5
Input Array[0][1][1] = 8
Input Array[0][1][2] = 2
Input Array[0][2][0] = 9
Input Array[0][2][1] = 3
Input Array[0][2][2] = 7
Input Array[1][0][0] = 5
Input Array[1][0][1] = 1
Input Array[1][0][2] = 5
Input Array[1][1][0] = 8
Input Array[1][1][1] = 9
Input Array[1][1][2] = 2
Input Array[1][2][0] = 7
Input Array[1][2][1] = 2
Input Array[1][2][2] = 1
Input Array[2][0][0] = 4
Input Array[2][0][1] = 5
Input Array[2][0][2] = 7
Input Array[2][1][0] = 2
Input Array[2][1][1] = 8
Input Array[2][1][2] = 9
Input Array[2][2][0] = 4
Input Array[2][2][1] = 2
Input Array[2][2][2] = 8

```

```

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

```

```

Tampilan Array:

```

```

6 4 2
5 8 2
9 3 7

5 1 5
8 9 2
7 2 1

4 5 7
2 8 9
4 2 8

```

PS D:\Telkom University\Teknik Informatika\Semester 2 (S1 IF-11-A)\Praktikum Struktur data dan Algoritma\Modul2>

## Deskripsi program

Program diatas merupakan program memungkinkan pengguna untuk memasukkan elemen-elemen ke dalam array tiga dimensi dan kemudian menampilkan data tersebut serta menampilkan tampilan array secara visual dalam bentuk matriks. Program ini memungkinkan pengguna untuk bekerja dengan array tiga dimensi dan memberikan tampilan visual dari array tersebut dalam bentuk matriks, sehingga memudahkan untuk memahami struktur data array tiga dimensi.

### 3. Unguided 3

#### Source code

```
#include <iostream>

using namespace std;

int main() {
    int size;

    // Meminta pengguna untuk memasukkan ukuran array
    cout << "Masukkan jumlah elemen array: ";
    cin >> size;

    // Mendeklarasikan array sesuai dengan ukuran yang
    // dimasukkan pengguna
    int arr[size];

    // Meminta pengguna untuk memasukkan elemen array
    cout << "Masukkan elemen-elemen array:\n";
    for (int i = 0; i < size; ++i) {
        cout << "Elemen ke-" << i + 1 << ": ";
        cin >> arr[i];
    }

    // Menampilkan menu
    cout << "\nPilih operasi yang ingin Anda lakukan:\n";
    cout << "1. Cari nilai maksimum\n";
    cout << "2. Cari nilai minimum\n";
    cout << "3. Cari nilai rata-rata\n";

    int choice;
    cout << "Masukkan pilihan: ";
    cin >> choice;

    // Proses pemilihan
    switch (choice) {
        case 1: {
            // Mencari nilai maksimum
            int max = arr[0];
            for (int i = 1; i < size; ++i) {
                if (arr[i] > max) {
                    max = arr[i];
                }
            }
            cout << "Nilai maksimum dalam array adalah: " <<
max << endl;
            break;
        }
        case 2: {
            // Mencari nilai minimum
            int min = arr[0];
```

```

        for (int i = 1; i < size; ++i) {
            if (arr[i] < min) {
                min = arr[i];
            }
        }
        cout << "Nilai minimum dalam array adalah: " << min
<< endl;
        break;
    }
    case 3: {
        // Menghitung nilai rata-rata
        int sum = 0;
        for (int i = 0; i < size; ++i) {
            sum += arr[i];
        }
        double average = static_cast<double>(sum) / size;
        cout << "Nilai rata-rata dari array adalah: " <<
average << endl;
        break;
    }
    default:
        cout << "Pilihan tidak valid." << endl;
}

return 0;
}

```

## Screenshot program

### - Menu cari nilai maksimum

```

PS D:\Telkom University\Teknik Informatika\Semester 2 (S1 IF-11-A)\Praktikum Struktur data dan Algoritma\Modul2> cd "d:\Telkom University\Teknik Informatika\Semester
2 (S1 IF-11-A)\Praktikum Struktur data dan Algoritma\Modul2\" ; if ($?) { g++ Unguided_3.cpp -o Unguided_3 } ; if ($?) { .\Unguided_3 }
Masukkan jumlah elemen array: 4
Masukkan elemen-elemen array:
Elemen ke-1: 5
Elemen ke-2: 3
Elemen ke-3: 8
Elemen ke-4: 1

Pilih operasi yang ingin Anda lakukan:
1. Cari nilai maksimum
2. Cari nilai minimum
3. Cari nilai rata-rata
Masukkan pilihan: 1
Nilai maksimum dalam array adalah: 8
PS D:\Telkom University\Teknik Informatika\Semester 2 (S1 IF-11-A)\Praktikum Struktur data dan Algoritma\Modul2>

```

### - Menu cari nilai minimum

```

PS D:\Telkom University\Teknik Informatika\Semester 2 (S1 IF-11-A)\Praktikum Struktur data dan Algoritma\Modul2> cd "d:\Telkom University\Teknik Informatika\Semester
2 (S1 IF-11-A)\Praktikum Struktur data dan Algoritma\Modul2\" ; if ($?) { g++ Unguided_3.cpp -o Unguided_3 } ; if ($?) { .\Unguided_3 }
Masukkan jumlah elemen array: 5
Masukkan elemen-elemen array:
Elemen ke-1: 10
Elemen ke-2: 46
Elemen ke-3: 62
Elemen ke-4: 39
Elemen ke-5: 19

Pilih operasi yang ingin Anda lakukan:
1. Cari nilai maksimum
2. Cari nilai minimum
3. Cari nilai rata-rata
Masukkan pilihan: 2
Nilai minimum dalam array adalah: 10
PS D:\Telkom University\Teknik Informatika\Semester 2 (S1 IF-11-A)\Praktikum Struktur data dan Algoritma\Modul2>

```

## - Menu cari nilai rata-rata

```
PS D:\Telkom University\Teknik Informatika\Semester 2 (S1 IF-11-A)\Praktikum Struktur data dan Algoritma\Modul2> cd "d:\Telkom University\Teknik Informatika\Semester 2 (S1 IF-11-A)\Praktikum Struktur data dan Algoritma\Modul2\" ; if ($?) { g++ Unguided_3.cpp -o Unguided_3 } ; if ($?) { .\Unguided_3 }
Masukkan jumlah elemen array: 6
Masukkan elemen-elemen array:
Elemen ke-1: 68
Elemen ke-2: 99
Elemen ke-3: 36
Elemen ke-4: 97
Elemen ke-5: 29
Elemen ke-6: 34

Pilih operasi yang ingin Anda lakukan:
1. Cari nilai maksimum
2. Cari nilai minimum
3. Cari nilai rata-rata
Masukkan pilihan: 3
Nilai rata-rata dari array adalah: 60.5
PS D:\Telkom University\Teknik Informatika\Semester 2 (S1 IF-11-A)\Praktikum Struktur data dan Algoritma\Modul2>
```

### Deskripsi program

Program diatas merupakan program yang meminta pengguna untuk memasukkan sejumlah elemen ke dalam array dan kemudian memilih operasi yang ingin dilakukan terhadap array tersebut. Pilihan operasi yang tersedia adalah mencari nilai maksimum, nilai minimum dan nilai rata-rata dari elemen array.

## **BAB IV**

### **KESIMPULAN**

Array adalah struktur data yang menyimpan sejumlah elemen dengan tipe data yang sama. Elemen-elemen ini dapat diakses menggunakan indeks yang mengidentifikasi posisi relative mereka dalam array. Array dideklarasikan dengan menyebutkan tipe data elemen array diikuti dengan nama array dan ukuran array. Ukuran array bisa ditentukan pada saat kompilasi (statis) atau pada saat runtime (dinamis). Elemen-elemen array dapat dimasukkan dan diakses menggunakan indeks, indeks array dimulai dari 0 hingga ukuran array dikurangi satu. Beberapa operasi yang umum dilakukan pada array meliputi mencari nilai maksimum, minimum dan rata-rata dari elemen array. Operasi ini dapat dilakukan dengan menggunakan iterasi melalui array dan membandingkan nilai-nilai elemen array.

Selain array satu dimensi, C++ juga mendukung array multi dimensi, seperti array dua dimensi (matriks) dan array tiga dimensi. Array multi dimensi memungkinkan penyimpanan data dalam struktur yang lebih kompleks, seperti tabel atau kubus. Kelebihan array adalah kemampuannya untuk menyimpan sejumlah elemen dengan tipe data yang sama dan memungkinkan akses cepat ke elemen-elemennya menggunakan indeks. Namun, kelemahannya adalah ukuran array harus diketahui sebelumnya dan tidak dapat diubah setelah deklarasi, serta tidak memiliki fleksibilitas untuk menangani penambahan atau pengurangan elemen

Dalam praktikum, array digunakan untuk menyimpan data dalam jumlah besar, seperti data sensor, nilai-nilai mahasiswa atau elemen-elemen lain yang memiliki hubungan atau kesamaan tipe data. Operasi-operasi pada array dapat membantu dalam analisis data sederhana, seperti mencari nilai ekstrem atau nilai rata-rata. Array cocok digunakan Ketika kita memiliki sekumpulan data dengan tipe data yang sama dan ukuran array yang tetap. Namun, jika kita memerlukan struktur data yang lebih fleksibel dengan kemampuan penambahan atau pengurangan elemen, maka mungkin struktur data lain seperti vector atau daftar terhubung, lebih sesuai.