

# **LAPORAN PRAKTIKUM**

## **MODUL II ARRAY**



**Disusun oleh:**  
**Annisa Al Jauhar**  
**NIM: 2311102014**

**Dosen Pengampu:**  
Muhammad Afrizal Amrustian, S. Kom., M. Kom

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

# **BAB I**

## **TUJUAN PRAKTIKUM**

### **A. TUJUAN PRAKTIKUM**

1. Mahasiswa dapat mempelajari tipe data primitif, abstrak, dan kolektif.
2. Mahasiswa dapat mengetahui jenis dimensi array dan cara penulisannya.
3. Mahasiswa dapat mengimplementasikan Array pada kode program yang dibuat.

## **BAB II**

### **DASAR TEORI**

#### **B. DASAR TEORI**

Dalam beberapa literatur, array sering disebut sebagai larik. Array adalah kumpulan dari nilai- nilai data bertipe sama dalam urutan tertentu yang menggunakan sebuah nama yang sama. Nilai nilai data suatu array ditunjukkan oleh suatu subscript atau indeks.

Array bisa berupa array berdimensi satu, dua, tiga atau lebih. Array berdimensi satu mewakili bentuk vektor. Array berdimensi dua mewakili bentuk suatu matriks atau table. Array berdimensi tiga mewakili bentuk suatu ruang.

## 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

```

PS C:\Users\annis\OneDrive\Desktop\Semester 2\C++ VsCode Praktikum\Struktur Data rabu\Modul 2> cd "c:\Users\annis\OneDrive\Desktop\Semester 2\C++ VsCode Praktikum\Struktur Data rabu\Modul 2\" ; if ($?) { g++ guided1.cpp -o guided1 } ; if ($?) { .\guided1 }
Input Array[0][0][0] = 1
Input Array[0][0][1] = 2
Input Array[0][0][2] = 3
Input Array[0][1][0] = 4
Input Array[0][1][1] = 5
Input Array[0][1][2] = 6
Input Array[0][2][0] = 7
Input Array[0][2][1] = 8
Input Array[0][2][2] = 5

```

```

Input Array[1][0][0] = 4
Input Array[1][0][1] = 6
Input Array[1][0][2] = 3
Input Array[1][1][0] = 4
Input Array[1][1][1] = 5
Input Array[1][1][2] = 6
Input Array[1][2][0] = 7
Input Array[1][2][1] = 7
Input Array[1][2][2] = 5

```

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

123
456
785

463
456
775

PS C:\Users\annis\OneDrive\Desktop\Semester 2\C++ VsCode Praktikum\Struktur Data rabu\Modul 2>
```

## Deskripsi program

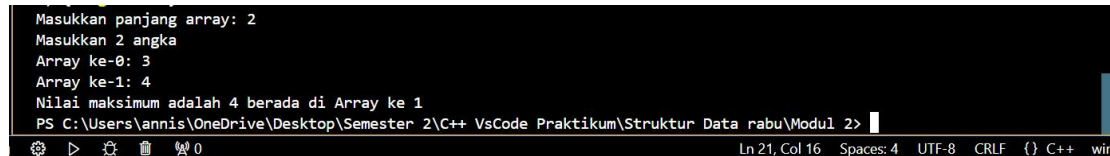
Program di atas merupakan program sederhana yang mengimplementasikan array tiga dimensi. Program ini meminta pengguna untuk memasukkan nilai-nilai ke dalam array tiga dimensi dengan ukuran 2x3x3. Pertama-tama, program mendeklarasikan array `arr` dengan ukuran 2x3x3. Selanjutnya, program menggunakan tiga loop bersarang untuk mengiterasi melalui semua elemen array. Dalam loop pertama, nilai `x` digunakan untuk mengakses indeks array pada dimensi pertama, kemudian loop kedua dan ketiga digunakan untuk mengakses indeks pada dimensi kedua dan ketiga. Pada setiap iterasi, program meminta pengguna untuk memasukkan nilai untuk setiap elemen array menggunakan `cin`. Setelah semua elemen array diisi, program mencetak nilai dari setiap elemen array menggunakan loop yang serupa dengan loop input, tetapi kali ini untuk menampilkan nilai dari setiap elemen array. Akhirnya, program juga mencetak tampilan array secara keseluruhan dengan menampilkan elemen-elemen array dalam format yang rapi.

## 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;
}
```

## Screenshoot program



```
Masukkan panjang array: 2
Masukkan 2 angka
Array ke-0: 3
Array ke-1: 4
Nilai maksimum adalah 4 berada di Array ke 1
PS C:\Users\annis\OneDrive\Desktop\Semester 2\C++ VsCode Praktikum\Struktur Data rabu\Modul 2>
```

## Deskripsi program

Program di atas merupakan program sederhana yang bertujuan untuk mencari nilai maksimum dalam sebuah array yang diinput oleh pengguna. Program ini dimulai dengan mendeklarasikan beberapa variabel seperti `maks` untuk menyimpan nilai maksimum, `a` untuk menyimpan panjang array, `i` sebagai variabel iterator, dan `lokasi` untuk menyimpan indeks dari nilai maksimum dalam array. Selanjutnya, program meminta pengguna untuk memasukkan panjang array melalui `cin`. Kemudian, array dengan panjang yang sesuai dideklarasikan. Program kemudian meminta pengguna untuk memasukkan nilai-nilai array satu per satu dalam loop `for`. Di dalam loop tersebut, setiap elemen array diminta melalui `cin`. Setelah semua nilai array dimasukkan, program melakukan iterasi melalui array untuk mencari nilai maksimum. Setiap kali nilai yang lebih besar ditemukan, nilai maksimum (`maks`) dan indeksnya (`lokasi`) diperbarui. Setelah selesai mencari nilai maksimum, program mencetak nilai maksimum beserta indeksnya dalam array.



## LATIHAN KELAS - UNGUIDED

### 1. Unguided 1

#### Source code

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

int main() {
    cout << "===== " <<
endl;
    cout << " Menentukan Bilangan Ganjil dan Genap " << endl;
    cout << "===== " <<
endl;
    cout << endl;

    int b;
    cout << "Masukkan Panjang Array: ";
    cin >> b;

    int bilangan[b];
    for(int i = 0; i < b; i++) {
        cout << "Array ke-" << i << ": ";
        cin >> bilangan[i];
    }

    cout << endl;
    cout << "Data Array: ";
    for(int i = 0; i < b; i++) {
        cout << bilangan[i] << " ";
    }
    cout << endl;

    cout << "Bilangan Genap: ";
```

```

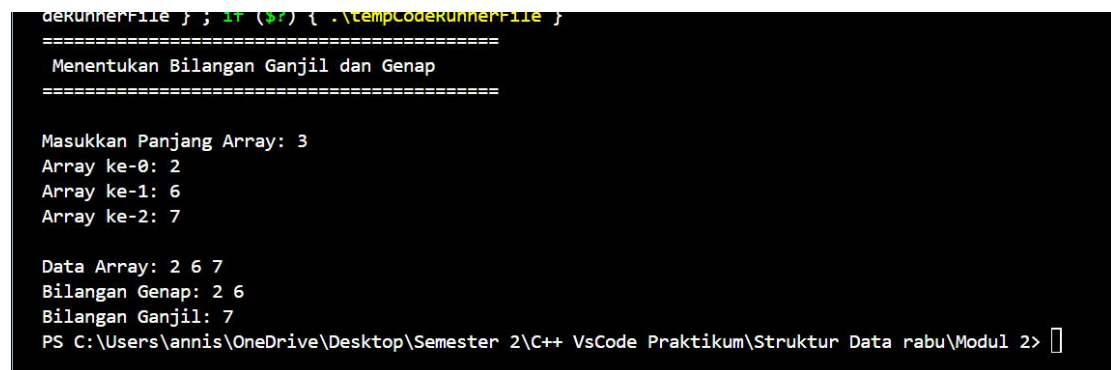
    for (int i = 0; i < b; i++) {
        if (bilangan[i] % 2 == 0) {
            cout << bilangan[i] << " ";
        }
    }
    cout << endl;

    cout << "Bilangan Ganjil: ";
    for (int i = 0; i < b; i++) {
        if (bilangan[i] % 2 != 0) {
            cout << bilangan[i] << " ";
        }
    }
    cout << endl;

    return 0;
}

```

## Screenshoot program



```

=====
Menentukan Bilangan Ganjil dan Genap
=====

Masukkan Panjang Array: 3
Array ke-0: 2
Array ke-1: 6
Array ke-2: 7

Data Array: 2 6 7
Bilangan Genap: 2 6
Bilangan Ganjil: 7
PS C:\Users\annis\OneDrive\Desktop\Semester 2\C++ VsCode Praktikum\Struktur Data rabu\Modul 2>

```

## Deskripsi program

Program di atas adalah program sederhana yang memungkinkan pengguna untuk memasukkan sejumlah bilangan bulat ke dalam sebuah array. Setelah memasukkan bilangan-bilangan tersebut, program akan menampilkan kembali bilangan-bilangan tersebut serta memisahkan antara bilangan genap dan ganjil.

Pertama-tama, pengguna diminta untuk memasukkan panjang array yang diinginkan. Kemudian, pengguna diminta untuk memasukkan nilai-nilai array satu per satu. Setelah itu, program akan menampilkan kembali nilai-nilai array yang telah dimasukkan. Selanjutnya, program akan menampilkan bilangan-bilangan genap yang terdapat dalam array, diikuti oleh bilangan-bilangan ganjil. Program ini berguna untuk memahami konsep loop, array, dan penggunaan operator modulus untuk menentukan bilangan ganjil dan genap dalam array.

## 2. Unguided 2

### Source code

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

int main() {
    int x, y, z;

    cout << "Masukkan jumlah elemen untuk dimensi pertama: ";
    cin >> x;
    cout << "Masukkan jumlah elemen untuk dimensi kedua: ";
    cin >> y;
    cout << "Masukkan jumlah elemen untuk dimensi ketiga: ";
    cin >> z;

    int arr[x][y][z];
    cout << "Masukkan elemen-elemen array:\n";
    for (int i = 0; i < x; ++i) {
        for (int j = 0; j < y; ++j) {
            for (int k = 0; k < z; ++k) {
                cout << "Array[" << i << "][" << j << "][" << k
<< "]: ";
                cin >> arr[i][j][k];
            }
        }
    }

    cout << "\nElemen-elemen array yang dimasukkan:\n";
    for (int i = 0; i < x; ++i) {
        for (int j = 0; j < y; ++j) {
            for (int k = 0; k < z; ++k) {
                cout << "Array[" << i << "][" << j << "][" << k
<< "]: " << arr[i][j][k] << endl;
            }
        }
    }
}
```

```
    }  
    }  
}  
  
return 0;  
}
```

### Screenshoot program

#### ▼ TERMINAL

```
Masukkan jumlah elemen untuk dimensi pertama: 2  
Masukkan jumlah elemen untuk dimensi kedua: 3  
Masukkan jumlah elemen untuk dimensi ketiga: 4  
Masukkan elemen-elemen array:  
Array[0][0][0]: 1  
Array[0][0][1]: 2  
Array[0][0][2]: 3  
Array[0][0][3]: 4  
Array[0][1][0]: 5  
Array[0][1][1]: 6  
Array[0][1][2]: 7  
Array[0][1][3]: 8  
Array[0][2][0]: 9  
Array[0][2][1]: 1  
Array[0][2][2]: 2  
Array[0][2][3]: 3  
Array[1][0][0]: 4  
Array[1][0][1]: 5  
Array[1][0][2]: 6  
Array[1][0][3]: 7  
Array[1][1][0]: 8  
Array[1][1][1]: 9  
Array[1][1][2]: 1  
Array[1][1][3]: 2  
Array[1][2][0]: 3  
Array[1][2][1]: 3  
Array[1][2][2]: 4  
Array[1][2][3]: 5
```

```
Elemen-elemen array ya
Array[0][0][0]: 1
Array[0][0][1]: 2
Array[0][0][2]: 3
Array[0][0][3]: 4
Array[0][1][0]: 5
Array[0][1][1]: 6
Array[0][1][2]: 7
Array[0][1][3]: 8
Array[0][2][0]: 9
Array[0][2][1]: 1
Array[0][2][2]: 2
Array[0][2][3]: 3
Array[1][0][0]: 4
Array[1][0][1]: 5
Array[1][0][2]: 6
Array[1][0][3]: 7
Array[1][1][0]: 8
Array[1][1][1]: 9
Array[1][1][2]: 1
Array[1][1][3]: 2
Array[1][2][0]: 3
Array[1][2][1]: 3
Array[1][2][2]: 4
Array[1][2][3]: 5
PS C:\Users\annis\OneD
```

### Deskripsi program

Program di atas adalah program sederhana yang memungkinkan pengguna untuk memasukkan jumlah elemen untuk tiga dimensi array. Pertama-tama, program meminta pengguna untuk memasukkan jumlah elemen untuk masing-masing dimensi array, yaitu dimensi pertama, kedua, dan ketiga. Setelah menerima input, program mendeklarasikan sebuah array tiga dimensi dengan ukuran yang sesuai berdasarkan input pengguna. Kemudian, program meminta pengguna untuk memasukkan nilai-nilai elemen array satu per satu melalui loop bersarang, dengan setiap elemen dimasukkan sesuai dengan indeksnya. Setelah semua elemen dimasukkan, program menampilkan kembali semua elemen array yang dimasukkan oleh pengguna dengan mencetak setiap elemen beserta indeksnya dalam array. Program ini memberikan

fleksibilitas kepada pengguna untuk menentukan ukuran array tiga dimensi sesuai kebutuhan mereka, dan kemudian memasukkan nilai-nilai sesuai dengan ukuran yang dimasukkan.

### 3. Unguided 3

#### Source code

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

int main() {
    vector<int> arr;
    int n, num;
    double sum = 0;

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

    cout << "Masukkan elemen array:\n";
    for (int i = 0; i < n; i++) {
        cin >> num;
        arr.push_back(num);
        sum += num;
    }
    char choice;
    cout << "\nPilih operasi yang ingin dilakukan:\n";
    cout << "1. Mencari nilai maksimum\n";
    cout << "2. Mencari nilai minimum\n";
    cout << "3. Menghitung nilai rata-rata\n";
    cin >> choice;

    switch (choice) {
        case '1':
            cout << "Nilai maksimum: " <<
                *max_element(arr.begin(), arr.end()) << endl;
            break;
        case '2':
```



```

        cout << "Nilai minimum: " <<
        *min_element(arr.begin(), arr.end()) << endl;
        break;
    case '3':
        cout << "Nilai rata-rata: " << sum / n << endl;
        break;
    default:
        cout << "Pilihan tidak valid.\n";
    }

    return 0;
}

```

### Screenshoot program

```

codeRunnerFile } ; if ($?) { .\tempCodeRunnerFile }
Masukkan jumlah elemen array: 2
Masukkan elemen array:
1
2

Pilih operasi yang ingin dilakukan:
1. Mencari nilai maksimum
2. Mencari nilai minimum
3. Menghitung nilai rata-rata
3
Nilai rata-rata: 1.5
PS C:\Users\annis\OneDrive\Desktop\Semester 2\C++ VsCode Praktikum\Struktur Data rabu\Modul 2>

```

### Deskripsi program

Program di atas adalah sebuah program yang memanfaatkan vektor untuk menyimpan elemen-elemen array dinamis. Pertama-tama, program meminta pengguna untuk memasukkan jumlah elemen array yang diinginkan melalui `cin`. Setelah itu, program meminta pengguna untuk memasukkan nilai-nilai elemen array satu per satu dalam loop `for`. Selama memasukkan nilai-nilai, setiap nilai dimasukkan ke dalam vektor `arr`, dan pada saat yang bersamaan, nilai-nilai tersebut ditambahkan ke dalam variabel `sum` untuk menghitung jumlah keseluruhan elemen array. Setelah semua elemen array dimasukkan, pengguna diminta untuk memilih operasi yang ingin dilakukan melalui pilihan yang tersedia. Pilihan tersebut adalah mencari nilai maksimum, minimum, atau menghitung rata-rata dari elemen-elemen

array. Bergantung pada pilihan pengguna, program menggunakan pernyataan `'switch'` untuk menjalankan operasi yang sesuai. Jika pengguna memilih mencari nilai maksimum, program menggunakan fungsi `'max_element'` dari library `'<algorithm>'` untuk menemukan nilai maksimum dalam vektor `'arr'`, dan hasilnya dicetak. Jika pengguna memilih mencari nilai minimum, program menggunakan fungsi `'min_element'` untuk menemukan nilai minimum dan mencetaknya. Jika pengguna memilih menghitung nilai rata-rata, program membagi jumlah total elemen dengan jumlah elemen dan mencetak hasilnya. Jika pengguna memilih opsi lain, program memberikan pesan bahwa pilihan tidak valid.

## **BAB IV**

### **KESIMPULAN**

Struktur adalah sekumpulan elemen-elemen data yang digabung menjadi suatu kesatuan. Struktur array adalah kumpulan elemen-elemen data yang digabungkan menjadi suatu kesatuan yang memiliki tipe homogen (sama). Array merupakan bagian dari struktur data yaitu termasuk ke dalam struktur data sederhana yang dapat didefinisikan sebagai pemesanan alokasi memori sementara pada komputer.

Apabila kita membuat program dengan data yang sudah kita ketahui batasnya, maka kita menggunakan array (tipe data statis), namun apabila datanya belum kita ketahui batasnya maka gunakan pointer (tipe data dinamis). Elemen-elemen array tersusun secara sekuensial dalam memori komputer. Array dapat berupa satu dimensi, dua dimensi, ataupun multidimensi.