

LAPORAN PRAKTIKUM

MODUL II ARRAY



Disusun oleh:
Imelda Fajar Awalina Crisyanti
NIM: 2311102004

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.

Contoh:

```
#include <iostream>
using namespace std;
int main() {
    int arr[5] = {9, 3, 5, 2, 1}; //deklarasi array
    cout<< arr[1] << endl;
    cout<< arr[4];
}
```

Output

```
3
1
```

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. Contoh :

```
#include <iostream>
using namespace std;
int main() {
    int arr[2][2] = {{3, 2}, {2, 5}};
    for (int i=0; i<2; i++) { //baris
        for(int j=0; j<2; j++) { //kolom
            cout<< arr[i][j] << ends;
        }
        cout << endl;
    }
}
```

Output

```
3  2
2  5
```

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

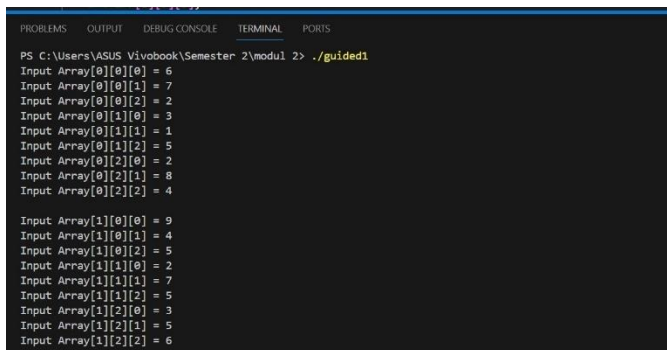
```
#include <iostream>
using namespace std;

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;
    }

    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;
}
```

```
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\ASUS Vivobook\Semester 2\modul 2> ./guided1
Input Array[0][0][0] = 6
Input Array[0][0][1] = 7
Input Array[0][0][2] = 2
Input Array[0][1][0] = 3
Input Array[0][1][1] = 1
Input Array[0][1][2] = 5
Input Array[0][2][0] = 2
Input Array[0][2][1] = 8
Input Array[0][2][2] = 4

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

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

```
672
315
284

945
275
356

PS C:\Users\ASUS Vivobook\Semester 2\modul 2> |
```

Deskripsi program

Program di atas merupakan deklarasi array dengan tiga dimensi. Ini berarti ada dua lapisan, masing-masing lapisan memiliki tiga baris, dan setiap baris memiliki tiga elemen.

Program meminta untuk memasukkan nilai-nilai ke dalam array dengan menggunakan loop for bertingkat. Loop pertama mengakses lapisan, loop kedua mengakses baris, dan loop ketiga mengakses elemen dalam baris tersebut. Selanjutnya yaitu memasukkan nilai melalui cin, dan nilai tersebut disimpan di dalam array.

Setelah semua nilai dimasukkan, program mencetak kembali nilai-nilai array yang telah dimasukkan dengan menggunakan loop yang serupa.

Ada satu loop tambahan yang mencetak nilai-nilai array tanpa label atau spasi tambahan di antara elemen-elemen yang berdekatan.

Program ini berguna untuk mendemonstrasikan cara menggunakan array multidimensi dalam bahasa pemrograman C++, serta cara membaca dan menulis nilai-nilai ke dalamnya.

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



The screenshot shows a terminal window with the following output:

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Masukkan panjang array: 5
Masukkan 5 angka
Array ke-0: 4
Array ke-1: 5
Array ke-2: 8
Array ke-3: 2
Array ke-4: 7
Nilai maksimum adalah 8 berada di Array ke 2
PS C:\Users\ASUS Vivobook\Semester 2\modul 2> |
```


Deskripsi program

Program ini digunakan untuk memasukkan panjang array melalui cin. Panjang array ini akan digunakan untuk mendeklarasikan array dengan ukuran yang sesuai.

Setelah panjang array dimasukkan, program mendeklarasikan array dengan menggunakan nilai yang dimasukkan pengguna.

Setelah itu masukkan nilai-nilai array satu per satu menggunakan loop for. Setiap nilai array dimasukkan oleh pengguna akan disimpan di dalam array.

Setelah semua nilai array dimasukkan, program melakukan iterasi melalui array untuk mencari nilai maksimum. Variabel maks digunakan untuk menyimpan nilai maksimum yang ditemukan, sedangkan variabel lokasi digunakan untuk menyimpan indeks dimana nilai maksimum tersebut berada.

Setelah nilai maksimum ditemukan, program mencetak nilai maksimum beserta indeksnya ke layar.

LATIHAN KELAS - UNGUIDED

1. Unguided 1

Source code

```
#include <iostream>

using namespace std;

int daftar_bil[10] = {21, 22, 23, 24, 25, 26, 27, 28, 29, 30};

int main() {
    int indeks;
    int ganjil = daftar_bil [0];
    int genap = daftar_bil[0];

    cout << "Bilangan = ";
    for (indeks = 0; indeks < 10; indeks++) {
        cout << daftar_bil[indeks] << ", ";

        if (daftar_bil[indeks] < ganjil) {
            ganjil = daftar_bil[indeks];
        }

        if (daftar_bil[indeks] > genap) {
            genap = daftar_bil[indeks];
        }
    }

    cout << endl;
    cout << "Nomor Genap : ";
    for (indeks = 0; indeks < 10; indeks++) {
        if (daftar_bil[indeks] % 2 == 0) {
            cout << daftar_bil[indeks] << ", ";
        }
    }

    cout << endl;

    cout << "Nomor Ganjil : ";
```

```

    for (indeks = 0; indeks < 10; indeks++) {
        if (daftar_bil[indeks] % 2 != 0) {
            cout << daftar_bil[indeks] << ", ";
        }
    }

    cout << endl;

    return 0;
}

```

Screenshoot program

```

PS C:\Users\ASUS Vivobook\Semester 2\modul 2\.vscode> g++ unguided1.cpp -o unguided1
PS C:\Users\ASUS Vivobook\Semester 2\modul 2\.vscode> ./unguided1
Bilangan = 21, 22, 23, 24, 25, 26, 27, 28, 29, 30,
Nomor Genap : 22, 24, 26, 28, 30,
Nomor Ganjil : 21, 23, 25, 27, 29,
PS C:\Users\ASUS Vivobook\Semester 2\modul 2\.vscode>

```

Deskripsi program

Langkah pertama Program mendefinisikan array daftar_bil yang berisi 10 bilangan. Program memulai fungsi main(). Program mendeklarasikan variabel indeks untuk digunakan dalam loop. Program mendeklarasikan variabel ganjil dan genap dan menginisialisasinya dengan elemen pertama dari array daftar_bil. Program mencetak semua bilangan dalam array daftar_bil menggunakan loop for. Selama iterasi, program juga memperbarui nilai ganjil dan genap jika ditemukan bilangan yang lebih kecil atau lebih besar, sesuai kebutuhan.

Setelah mencetak semua bilangan, program mencetak bilangan genap dengan melakukan iterasi melalui array dan memeriksa apakah setiap bilangan habis dibagi dua (bilangan % 2 == 0). Bilangan yang memenuhi kondisi tersebut dicetak.

Program kemudian mencetak bilangan ganjil dengan melakukan iterasi melalui array dan memeriksa apakah setiap bilangan tidak habis dibagi dua (bilangan % 2 != 0). Bilangan yang memenuhi kondisi tersebut dicetak.

2. Unguided 2

Source code

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

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

    cout << "Masukkan jumlah elemen untuk dimensi X: ";
    cin >> x_size;
    cout << "Masukkan jumlah elemen untuk dimensi Y: ";
    cin >> y_size;
    cout << "Masukkan jumlah elemen untuk dimensi Z: ";
    cin >> z_size;

    int arr[x_size][y_size][z_size];

    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];
            }
        }
        cout << endl;
    }

    cout << "Data Array:" << endl;
    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;
            }
        }
    }

    cout << endl << "Array dalam format ringkas:" << endl;
    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;
}

```

Screenshoot program

```

Masukkan jumlah elemen untuk dimensi X: 2
Masukkan jumlah elemen untuk dimensi Y: 2
Masukkan jumlah elemen untuk dimensi Z: 2
Input Array[0][0][0] = 3
Input Array[0][0][1] = 5
Input Array[0][1][0] = 7
Input Array[0][1][1] = 9

Input Array[1][0][0] = 2
Input Array[1][0][1] = 4
Input Array[1][1][0] = 6
Input Array[1][1][1] = 8

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

Array dalam format ringkas:
3 5
7 9

2 4
6 8

```

Deskripsi program

Pertama Program meminta kita untuk memasukkan jumlah elemen untuk dimensi X, Y, dan Z melalui cin. Array tiga dimensi arr dideklarasikan dengan ukuran yang sesuai berdasarkan jumlah elemen yang dimasukkan pengguna.

Program menggunakan tiga loop bersarang untuk meminta pengguna memasukkan nilai-nilai elemen array. Loop pertama digunakan untuk dimensi X, loop kedua untuk dimensi Y, dan loop ketiga untuk dimensi Z. Setiap nilai array dimasukkan oleh pengguna melalui cin.

Setelah semua nilai array dimasukkan, program mencetak nilai-nilai array dalam format lengkap menggunakan tiga loop bersarang yang serupa dengan yang digunakan untuk memasukkan nilai-nilai.

Program kemudian mencetak nilai-nilai array dalam format yang lebih ringkas, yaitu dengan mencetak semua nilai dalam satu baris untuk setiap dimensi X, diikuti dengan baris kosong sebagai pemisah antara dimensi X.

3. Unguided 3

Source code

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

int cariMaksimum(int arr[], int panjang) {
    int maksimum = arr[0];
    for (int i = 1; i < panjang; i++) {
        if (arr[i] > maksimum) {
            maksimum = arr[i];
        }
    }
    return maksimum;
}

int cariMinimum(int arr[], int panjang) {
    int minimum = arr[0];
    for (int i = 1; i < panjang; i++) {
        if (arr[i] < minimum) {
            minimum = arr[i];
        }
    }
    return minimum;
}

double hitungRataRata(int arr[], int panjang) {
    double total = 0;
    for (int i = 0; i < panjang; i++) {
        total += arr[i];
    }
    return total / panjang;
}

int main() {
    int panjang;
    cout << "Masukkan panjang array: ";
    cin >> panjang;

    int arr[panjang];
    cout << "Masukkan elemen-elemen array: ";
```

```
for (int i = 0; i < panjang; i++) {
    cin >> arr[i];
}

int pilihan;
cout << "Menu:" << endl;
cout << "1. Cari nilai Maksimum" << endl;
cout << "2. Cari nilai Minimum" << endl;
cout << "3. Hitung nilai rata-rata" << endl;
cout << "Pilih opsi: ";
cin >> pilihan;

switch (pilihan) {
    case 1:
        cout << "Nilai Maksimum: " << cariMaksimum(arr,
panjang) << endl;
        break;
    case 2:
        cout << "Nilai Minimum: " << cariMinimum(arr, panjang)
<< endl;
        break;
    case 3:
        cout << "Nilai rata-rata: " << hitungRataRata(arr,
panjang) << endl;
        break;
    default:
        cout << "Pilihan tidak valid." << endl;
}

return 0;
}
```


Screenshoot program

```
PS C:\Users\ASUS Vivobook\Semester 2\modul 2\.vscode> ./unguided3
Masukkan panjang array: 3
Masukkan elemen-elemen array: 5 7 9
Menu:
1. Cari nilai Maksimum
2. Cari nilai Minimum
3. Hitung nilai rata-rata
Pilih opsi: 1
Nilai Maksimum: 9
```

Deskripsi program

Pengguna diminta untuk memasukkan panjang array dan elemen-elemen array. Setelah memasukkan array, pengguna diminta untuk memilih opsi dari menu: mencari nilai maksimum, minimum, atau nilai rata-rata. Program memiliki tiga fungsi terpisah untuk mencari nilai maksimum, minimum, dan menghitung nilai rata-rata dari array. Hasil sesuai dengan pilihan pengguna kemudian ditampilkan ke layar.

BAB IV

KESIMPULAN

Jadi menurut konsep Array merupakan struktur data yang digunakan untuk menyimpan kumpulan elemen yang memiliki tipe data yang sama dalam satu variabel. Konsep array memungkinkan kita untuk menyimpan beberapa nilai dalam satu variabel dengan nama yang sama dan mengakses nilai-nilai tersebut dengan menggunakan indeks.

Di dalam array ada berbagai dimensi, seperti satu dimensi (1D), dua dimensi (2D), atau bahkan dimensi yang lebih tinggi.

1. Array Satu Dimensi (1D):
 - Terdiri dari satu baris elemen.
 - Ditulis dengan menuliskan tipe data elemen diikuti oleh nama array dan jumlah elemen dalam tanda kurung siku.
 - Contoh: `int numbers[5];`
2. Array Dua Dimensi (2D):
 - Terdiri dari baris dan kolom.
 - Ditulis dengan menuliskan tipe data elemen diikuti oleh nama array dan jumlah baris dan kolom dalam tanda kurung siku.
 - Contoh: `int matrix[3][3];`
3. Array Tiga Dimensi (3D) dan Lebih Tinggi:
 - Terdiri dari tiga dimensi atau lebih, seperti baris, kolom, dan tinggi.
 - Ditulis dengan menambahkan tanda kurung siku untuk setiap dimensi tambahan.
 - Contoh array tiga dimensi: `int threeD[3][3][3];`

Dalam inisialisasi, nilai-nilai elemen array ditempatkan dalam kurung kurawal `{ }`. Panjang array (jumlah elemen dalam setiap dimensi) biasanya ditentukan dalam deklarasi, tetapi dalam beberapa kasus, juga bisa dinamis tergantung pada bahasa pemrograman yang digunakan.