

**LAPORAN PRAKTIKUM
STRUKTUR DATA DAN ALGORITMA
MODUL 2
“ARRAY”**



DISUSUN OLEH:
Mohammad Harits Tantowi
2311102016
S1 IF-11-A

DOSEN PENGAMPU:
Wahyu Andi Saputra, S. Pd.,M. Eng

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

BAB I

TUJUAN PRAKTIKUM

1. Dapat Memahami Konsep Array.
2. Dapat Mengetahui Jenis Dimensi Array dan Cara Penulisannya.
3. 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.

Contoh :

```
#include <iostream>

using namespace std;

int main() {
    int arr[2][2][3] = {{{2, 8, 7}, {6, 5, 1}}, {{8, 5, 2}, {9, 2, 7}}};
    for (int i=0; i<2; i++) {
        for(int j=0; j<2; j++) {
            for(int k=0; k<3; k++) {
                cout<< arr[i][j][k] << ends;
            };
            cout<< endl;
        };
    };
    cout<< endl;
}
```

```
};  
}
```

Output :

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

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 : Program Input Array Tiga Dimensi

```
#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;
}
}

```

Screenshots Output

```

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] = 9

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

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

123
456
789

987
654
321

```

Deskripsi :

Program ini adalah contoh implementasi dari array tiga dimensi dalam bahasa pemrograman C++. Array tiga dimensi digunakan untuk menyimpan data dalam struktur tiga dimensi, di mana data tersebut tersusun dalam bentuk kubus atau balok tiga dimensi. Dalam program ini, terdapat array tiga dimensi dengan ukuran 2x3x3. Artinya, terdapat dua "layer" atau lapisan, setiap lapisan memiliki tiga "baris", dan setiap baris memiliki tiga "kolom".

Pertama, program meminta pengguna untuk memasukkan nilai-nilai ke dalam array menggunakan nested loop. Pengguna diminta untuk memasukkan nilai untuk setiap elemen array dengan mengikuti format "Input Array[x][y][z] =", di mana x, y, dan z mewakili indeks array. Setelah pengguna memasukkan semua nilai, program kemudian menampilkan nilai-nilai tersebut kembali dengan format "Data Array[x][y][z] = nilai".

Selanjutnya, program menampilkan tampilan array secara langsung dengan cara mencetak nilai-nilai array dalam bentuk matriks dua dimensi. Proses ini dilakukan dengan menggunakan nested loop yang mencetak nilai-nilai array dan menambahkan baris baru setiap selesai mencetak satu baris.

2. Guided 2 : Program Mencari Nilai Maksimal pada Array

```
#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;
}

```

Screenshots Output

```

Masukkan panjang array: 9
Masukkan 9 angka
Array ke-0: 5
Array ke-1: 6
Array ke-2: 9
Array ke-3: 7
Array ke-4: 8
Array ke-5: 1
Array ke-6: 3
Array ke-7: 2
Array ke-8: 4
Nilai maksimum adalah 9 berada di Array ke 2

```

Deskripsi :

Program ini adalah sebuah program sederhana yang ditulis dalam bahasa pemrograman C++. Tujuan utama dari program ini adalah untuk mencari nilai maksimum dari sebuah array yang diisi oleh pengguna. Program dimulai dengan meminta pengguna untuk memasukkan panjang array, kemudian menginisialisasi array dengan panjang yang telah dimasukkan. Setelah itu, pengguna diminta untuk memasukkan nilai-nilai array satu per satu.

Setelah semua nilai dimasukkan, program melakukan iterasi melalui seluruh elemen array untuk menemukan nilai maksimum. Proses ini dilakukan dengan membandingkan setiap elemen array dengan nilai maksimum yang saat ini disimpan dalam variabel maks. Jika nilai elemen array yang sedang diperiksa lebih besar dari nilai maksimum yang disimpan, maka nilai maksimum diperbarui bersamaan dengan lokasi (indeks) elemen tersebut dalam array.

Setelah itu, program mengeluarkan nilai maksimum beserta lokasi (indeks) di mana nilai tersebut ditemukan dalam array. Program kemudian selesai dieksekusi.

BAB IV

UNGUIDED

1. 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,
```

Source code

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

using namespace std;

int main() {

    const int JUMLAH_NOMOR = 10;
    int nomor[JUMLAH_NOMOR] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};

    cout << "Data Array: ";
    for (int i = 0; i < JUMLAH_NOMOR; ++i) {
        cout << nomor[i];
        if (i != JUMLAH_NOMOR - 1) {
            cout << ", ";
        }
    }
    cout << endl;

    vector<int> ganjil, genap;

    for (int i = 0; i < JUMLAH_NOMOR; ++i) {
        if (nomor[i] % 2 == 0) {
            genap.push_back(nomor[i]);
        } else {
            ganjil.push_back(nomor[i]);
        }
    }
```

```

    }

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

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

    return 0;
}

```

Screenshots output

```

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

```

Deskripsi :

Program ini adalah contoh implementasi dalam bahasa pemrograman C++ yang mengilustrasikan penggunaan array dan vector untuk memisahkan nomor genap dan ganjil dari sebuah array yang telah ditentukan. Program ini mencetak informasi pribadi, yaitu nama dan nomor identifikasi, kemudian menampilkan isi array yang telah diinisialisasi dengan angka 1 hingga 10.

Setelah itu, program melakukan iterasi melalui setiap elemen array untuk memisahkan angka genap dan ganjil ke dalam dua vector terpisah, yaitu `genap` dan `ganjil`. Elemen-elemen genap dimasukkan ke dalam vector `genap`, sedangkan elemen-elemen ganjil dimasukkan ke dalam vector `ganjil`.

Selanjutnya, program mencetak isi dari kedua vector tersebut, yaitu nomor genap dan ganjil secara terpisah. Ini dilakukan dengan menggunakan perulangan untuk mengakses setiap elemen dalam vector dan mencetaknya ke layar. Setelah selesai, program mengakhiri eksekusi dengan mengembalikan nilai 0.

2. Unguided 2

Buatlah program Input array tiga dimensi (seperti pada guided) tetapi jumlah atau ukuran elemennya diinputkan oleh user!

Source code

```
#include <iostream>

using namespace std;

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

    cout << "Masukkan ukuran array dalam tiga dimensi:" << endl;
    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
    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;
    }

    // Output Array (dengan indeks)
    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 << "Array[" << x << "][" << y << "][" << z
<<"] = " << arr[x][y][z] << endl;
    }
}

cout << endl;

// Tampilan array (format matriks)
cout << "Array dalam bentuk matriks:" << 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] << "\t";
        }
        cout << endl;
    }
    cout << endl;
}

return 0;
}

```

Screenshots output

```

Masukkan ukuran array dalam tiga dimensi:
Ukuran dimensi x: 2
Ukuran dimensi y: 3
Ukuran dimensi z: 4
Input Array[0][0][0] = 1
Input Array[0][0][1] = 2
Input Array[0][0][2] = 3
Input Array[0][0][3] = 4
Input Array[0][1][0] = 5
Input Array[0][1][1] = 6
Input Array[0][1][2] = 7
Input Array[0][1][3] = 8
Input Array[0][2][0] = 9
Input Array[0][2][1] = 2
Input Array[0][2][2] = 3
Input Array[0][2][3] = 4

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

```

```

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

Array dalam bentuk matriks:
1    2    3    4
5    6    7    8
9    2    3    4

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

```

Deskripsi :

Program ini adalah contoh sederhana dari penggunaan array tiga dimensi dalam bahasa pemrograman C++. Program ini meminta pengguna untuk memasukkan ukuran array dalam tiga dimensi, yaitu dimensi x, y, dan z. Setelah itu, program akan meminta pengguna untuk memasukkan elemen-elemen array tersebut satu per satu. Setelah semua elemen dimasukkan, program akan menampilkan data array beserta indeksnya, kemudian menampilkan array dalam bentuk matriks dengan format yang lebih terstruktur.

Secara akademik, program ini menunjukkan penggunaan array multidimensi dan konsep nested loop untuk mengakses dan mengelola data dalam array tersebut. Selain itu, program juga menunjukkan interaksi sederhana dengan pengguna melalui input dari keyboard (cin) dan output ke layar (cout).

3. 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;

int main() {
    int n;
    cout << "Masukkan jumlah bilangan: ";
    cin >> n;

    int bilangan[n];
    cout << "Masukkan " << n << " bilangan:\n";
    for (int i = 0; i < n; ++i) {
        cout << "Bilangan ke-" << i + 1 << ": ";
        cin >> bilangan[i];
    }

    int maksimum = bilangan[0];
    int minimum = bilangan[0];
    double total = 0;

    for (int i = 0; i < n; ++i) {
        if (bilangan[i] > maksimum) {
            maksimum = bilangan[i];
        }
        if (bilangan[i] < minimum) {
            minimum = bilangan[i];
        }
        total += bilangan[i];
    }

    double rata_rata = total / n;

    cout << "Nilai maksimum: " << maksimum << endl;
    cout << "Nilai minimum: " << minimum << endl;
    cout << "Nilai rata-rata: " << rata_rata << endl;

    return 0;
}
```

Screenshots Output

```
Masukkan jumlah bilangan: 5
Masukkan 5 bilangan:
Bilangan ke-1: 4
Bilangan ke-2: 6
Bilangan ke-3: 8
Bilangan ke-4: 9
Bilangan ke-5: 6
Nilai maksimum: 9
Nilai minimum: 4
Nilai rata-rata: 6.6
```

Deskripsi :

Program ini merupakan sebuah program dalam bahasa C++ yang bertujuan untuk menghitung nilai maksimum, minimum, dan rata-rata dari sejumlah bilangan yang dimasukkan oleh pengguna. Program ini meminta pengguna untuk memasukkan jumlah bilangan yang akan diinput, kemudian meminta pengguna untuk memasukkan bilangan-bilangan tersebut satu per satu. Setelah semua bilangan dimasukkan, program akan melakukan perhitungan nilai maksimum, minimum, dan rata-rata dari bilangan-bilangan yang dimasukkan.

Untuk menghitung nilai maksimum dan minimum, program menggunakan pendekatan iteratif dengan membandingkan setiap bilangan yang dimasukkan dengan nilai maksimum dan minimum yang telah disimpan. Nilai maksimum dan minimum akan diperbarui jika ditemukan bilangan yang lebih besar atau lebih kecil dari nilai yang telah disimpan sebelumnya.

Setelah itu, program akan menghitung nilai rata-rata dari seluruh bilangan dengan menjumlahkan semua bilangan yang dimasukkan dan membaginya dengan jumlah bilangan tersebut.

Terakhir, program akan mencetak nilai maksimum, minimum, dan rata-rata tersebut ke layar untuk ditampilkan kepada pengguna.