

LAPORAN PRAKTIKUM

MODUL 2 ARRAY



Disusun oleh:
Denny Budiansyach NIM: 2311102022

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

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

BAB I

TUJUAN PRAKTIKUM

1. Mahasiswa dapat memahami konsep Array.
2. Mahasiswa dapat mengetahui jenis dimensi Array dan cara penulisannya.
3. 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.

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;
    };
}
```

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

1. GUIDED 1

```
#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

```

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

123
456
789

098
765
432

```

Deskripsi program

Program diatas adalah sebuah program yang akan melakukan output 2 array berukuran ordo 3×3 , yang mana nilai elemen dari array multidimensi tersebut berasal dari input user. Program ini mengimplementasikan bentuk array $[2][3][3]$ yang berarti 2 buah array dengan ukuran 3 baris dan 3 kolom. Program menggunakan 3 perulangan for bersarang untuk input elemen kedalam array yang dimulai dengan memenuhi kolom pada baris pertama dan seterusnya hingga baris terakhir.

2. GUIDED 2

```
#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: 4
Masukkan 4 angka
Array ke-0: 1
Array ke-1: 2
Array ke-2: 3
Array ke-3: 4
Nilai maksimum adalah 4 berada di Array ke 3
```

Deskripsi program

Program diatas adalah sebuah program yang akan mencari nilai maksimum elemen yang ada dalam array. Panjang dan nilai elemen setiap indeks dari array ditentukan oleh user melalui input program. Setelah baris array terbentuk melalui input user, program akan akan membandingkan elemen pertama array dengan elemen pada indeks selanjutnya untuk menemukan nilai maksimum.

LATIHAN KELAS - UNGUIDED

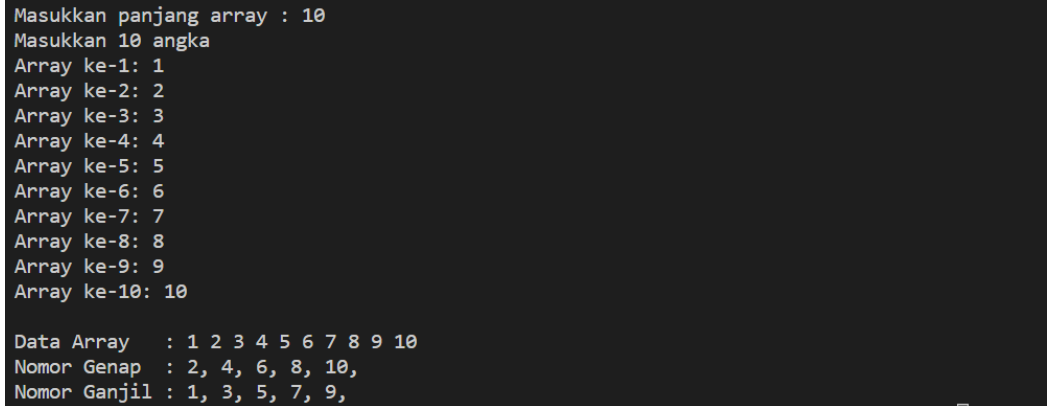
1. UNGUIDED 1

Source code

```
#include <iostream>
using namespace std;
int main()
{
    int a;
    cout << "Masukkan panjang array : ";
    cin >> a;
    int genap[a], ganjil[a], array[a];
    int tgenap = 0, tganjil = 0;

    cout << "Masukkan " << a << " angka\n";
    for (int i = 0; i < a; i++)
    {
        cout << "Array ke-" << (i+1) << ": ";
        cin >> array[i];
        if (array[i] % 2 == 0) {
            genap[tgenap] = array[i];
            tgenap++;
        } else {
            ganjil[tganjil] = array[i];
            tganjil++;
        }
    }
    cout << "\nData Array   : ";
    for (int i = 0; i < a; i++) {
        cout << array[i] << " ";
    }
    cout << "\nNomor Genap   : ";
    for (int i = 0; i < tgenap; i++) {
        cout << genap[i] << ", ";
    }
    cout << "\nNomor Ganjil  : ";
    for (int i = 0; i < tganjil; i++) {
        cout << ganjil[i] << ", ";
    }
    return 0;
}
```

Screenshoot Program



```
Masukkan panjang array : 10
Masukkan 10 angka
Array ke-1: 1
Array ke-2: 2
Array ke-3: 3
Array ke-4: 4
Array ke-5: 5
Array ke-6: 6
Array ke-7: 7
Array ke-8: 8
Array ke-9: 9
Array ke-10: 10

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

Program diatas adalah program yang akan menentukan angka ganjil dan angka genap dalam sebuah baris array. Panjang dan nilai elemen setiap indeks array ditentukan oleh user melalui input program. Ketika nilai elemen masuk melalui input user, program akan melakukan klasifikasi jenis angka melalui percabangan if else. Jika nilai elemen habis dibagi dengan 2 maka elemen pada indeks tersebut adalah genap dan akan ditampung pada variabel genap, jika tidak maka angka akan ditampung dalam variabel ganjil. Lalu pada akhir iterasi, program akan mengoutputkan semua data array, dan angka yang ada pada variabel genap dan angka yang ada pada variabel ganjil.

2. UNGUIDED 2

Source Code

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

int main() {
    int siz, baris, kolom;
    cout << "Masukkan ukuran array [x y z]: ";
    cin >> siz >> baris >> kolom;
    int arr[siz][baris][kolom];
    for (int x = 0; x < siz; x++) {
        for (int y = 0; y < baris; y++) {
            for (int z = 0; z < kolom; z++) {
                cout << "Input Array[" << x << "][" << y << "][" <<
z << "] = ";
                cin >> arr[x][y][z];
            }
        }
        cout << endl;
    }
    cout << "Data Array:\n";
    for (int x = 0; x < siz; x++) {
        for (int y = 0; y < baris; y++) {
            for (int z = 0; z < kolom; z++) {
                cout << "Data Array[" << x << "][" << y << "][" <<
z << "] = " << arr[x][y][z] << endl;
            }
        }
    }
    cout << endl;
    cout << "Data Array:\n";
    for (int x = 0; x < siz; x++) {
        for (int y = 0; y < baris; y++) {
            for (int z = 0; z < kolom; z++) {
                cout << arr[x][y][z] << ' ';
            }
            cout << endl;
        }
        cout << endl;
    }
    return 0;
}
```

Screenshoot Program

```
Masukkan ukuran array [x y z]: 2 3 3
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] = 98
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
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

Data Array:
1 2 3
4 5 6
7 8 9

98 8 7
6 5 4
3 2 1
```

Deskripsi program

Pada dasarnya program diatas adalah program yang sama dengan Guided 1, yang membedakan hanyalah variabel x, y, z yang merepresentasikan besar ukuran array dibuat sebagai variabel dinamis yang nilainya ditentukan oleh user. Dengan begitu jumlah dan ukuran dari array multidimensi bergantung pada nilai yang diinputkan oleh user.

3. UNGUIDED 3

Source Code

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

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

    int array[a];
    cout << "Masukan " << a << " angka\n";
    for (int i = 0; i < a; i++) {
        cout << "Array ke-" << (i + 1) << ": ";
        cin >> array[i];
    }

    int pilih;
    cout << "\nMenu:\n";
    cout << "1. Nilai Maksimum\n";
    cout << "2. Nilai Minimum\n";
    cout << "3. Rata-rata\n";
    cout << "Pilih menu : ";
    cin >> pilih;

    switch(pilih) {
        case 1: {
            int max = array[0];
            for (int i = 1; i < a; i++) {
                if (array[i] > max) {
                    max = array[i];
                }
            }
            cout << "Nilai Maksimum adalah " << max << endl;
            break;
        }
        case 2: {
            int min = array[0];
            for (int i = 1; i < a; i++) {
                if (array[i] < min) {
                    min = array[i];
                }
            }
            cout << "Nilai Minimum adalah " << min << endl;
            break;
        }
        case 3: {
            double sum = 0;
            for (int i = 0; i < a; i++) {
                sum += array[i];
            }
            double rata = sum / a;
            cout << "Rata-rata adalah " << rata << endl;
            break;
        }
    }
}
```

```

    }
}
cout << "Nilai Minimum adalah " << min << endl;
break;
}
case 3: {
    float sum = 0;
    for (int i = 0; i < a; i++) {
        sum += array[i];
    }
    float avg = sum / a;
    cout << "Rata-rata adalah " << avg << endl;
    break;
}
default:
    cout << "pilihan tidak validf\n";
}
return 0;
}

```

Screenshoot Program

Masukan panjang array: 2 Masukan 2 angka Array ke-1: 13 Array ke-2: 67 Menu: 1. Nilai Maksimum 2. Nilai Minimum 3. Rata-rata Pilih menu : 3 Rata-rata adalah 40	Masukan panjang array: 3 Masukan 3 angka Array ke-1: 12 Array ke-2: 23 Array ke-3: 31 Menu: 1. Nilai Maksimum 2. Nilai Minimum 3. Rata-rata Pilih menu : 1 Nilai Maksimum adalah 31	Masukan panjang array: 3 Masukan 3 angka Array ke-1: 12 Array ke-2: 23 Array ke-3: 31 Menu: 1. Nilai Maksimum 2. Nilai Minimum 3. Rata-rata Pilih menu : 2 Nilai Minimum adalah 12
--	---	--

Deskripsi program

Program diatas adalah program yang akan mencari nilai maksimum, minimum, dan rata rata dari elemen sebuah array. Panjang dan nilai elemen setiap indeks dari array ditentukan oleh user melalui input program. Dengan menggunakan switch case, program kemudian akan mengoutputkan menu untuk memproses array. Dalam menu maksimum dan minimum, program akan membandingkan setiap elemen dengan elemen lainnya hingga menemukan nilai terkecil atau terbesar dalam array. Dalam menu rata rata, program akan menjumlahkan semua elemen array kemudian membagi hasil jumlah tersebut dengan panjang array tersebut.

BAB IV

KESIMPULAN

Array adalah kumpulan dari nilai-nilai data bertipe yang sama. Nilai-nilai data di suatu array disebut dengan elemen-elemen array dan letak elemen array disebut dengan index yang dimulai dari 0 dan berakhir pada jumlah elemen dikurangi satu. Jumlah isi dari suatu array dapat di atur dengan memberikan tanda kurung siku- jumlah elemen yang ingin dimasukan pada array “[5]”. Array dapat dibedakan lagi menjadi array satu dimensi hingga multidimensi.

Array dua dimensi adalah struktur data yang terdiri dari kumpulan array satu dimensi dengan tipe yang sama yang disusun dalam baris dan kolom. Setiap elemen dalam array dua dimensi memiliki dua indeks, yaitu indeks untuk baris dan indeks untuk kolom, yang digunakan untuk mengakses elemen tersebut. Array dua dimensi sering digunakan untuk merepresentasikan data yang bersifat tabular, seperti matriks atau tabel.

Array multidimensi merupakan generalisasi dari array satu dimensi dan dua dimensi, yang memiliki kemampuan untuk merepresentasikan array dengan dimensi lebih dari dua. Array multidimensi dapat memiliki lebih dari dua indeks, seperti array tiga dimensi, array empat dimensi, array lima dimensi, dan seterusnya. Penggunaan array multidimensi seringkali diperlukan dalam pemrograman untuk menangani data yang kompleks dan memiliki struktur yang lebih kompleks.

DAFTAR PUSTAKA

- Chandra, J. A. (2017, November). *LAPORANG_PRAKTIKUM_IV_ARRAY(LARIK)*. Diambil kembali dari academia.edu: https://www.academia.edu/37423239/LAPORAN_PRAKTIKUM_IV_ARRA_Y_LARIK_Oleh
- sagar99. (2023, Agustus 23). *C++ Array Multidimensi*. Diambil kembali dari geeksforgeeks.org: <https://www.geeksforgeeks.org/cpp-multidimensional-array/>