

LAPORAN PRAKTIKUM STRUKTUR DATA DAN ALGORITMA

MODUL II ARRAY



Disusun Oleh :

Muhammad Rusdiyanto Asatman
2311102053

Dosen :

Wahyu Andi Saputra, S.Pd., M.Eng.

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

A. Dasar Teori

1. Pengertian Array

Secara sederhana, array merupakan suatu variabel yang dapat menyimpan banyak nilai dengan tipe data yang sama. Jumlah dari data yang dapat disimpan oleh suatu array ditentukan melalui alokasi memori dalam kode program (array statis). Namun hal ini tidak selalu benar untuk semua bahasa pemrograman, karena ada beberapa yang menyediakan array yang dapat mengatur alokasi memori secara otomatis (array dinamis). Contoh bahasa pemrograman dengan array statis adalah C dan C++, sementara array dinamis bisa didapati dalam Python dan JavaScript.

2. Dimensi Array

Dimensi dalam array mengacu pada jumlah indeks yang digunakan untuk mengakses elemen-elemen di dalamnya. Secara sederhana, dimensi adalah cara untuk mengatur data dalam suatu struktur terorganisir, mirip dengan rak buku di perpustakaan. Misalnya, dalam array satu dimensi, data disusun dalam satu baris tunggal, diakses menggunakan satu indeks. Sementara array dua dimensi, data diatur dalam baris dan kolom, dengan masing-masing elemen dapat diakses menggunakan dua indeks. Semakin banyak dimensi yang ada dalam array, semakin kompleks cara kita mengakses dan mengatur datanya. Analogi yang sederhana adalah memikirkan dimensi sebagai langkah-langkah yang diperlukan untuk menemukan elemen tertentu dalam struktur data array. Semakin banyak dimensi, semakin rumit cara kita "menavigasi" melalui struktur data tersebut.

Berikut adalah contoh deklarasi dari suatu array :

```
int arr1[3]; // array satu dimensi
int arr2[3][2]; // array dua dimensi
int arr3[2][3][2]; // array multi dimensi
int arr4...
```

Berdasarkan contoh tersebut, semakin banyak dimensi suatu array maka akan semakin kompleks pula strukturnya. Sebagai catatan tambahan, suatu array dapat dikatakan array multi dimensi jika dimensinya lebih dari 2 (dimensi array ≥ 3).

B. Guided

Guided 1

```
#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[1][0][2] = 12
Input Array[1][1][0] = 13
Input Array[1][1][1] = 14
Input Array[1][1][2] = 15
Input Array[1][2][0] = 16
Input Array[1][2][1] = 17
Input Array[1][2][2] = 18

```

```

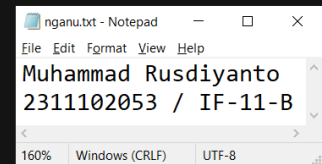
Data Array[0][0][0] = 1
Data Array[0][0][1] = 2
Data Array[0][0][2] = 3
Data Array[0][1][0] = 4
Data Array[0][1][1] = 5
Data Array[0][1][2] = 6
Data Array[0][2][0] = 7
Data Array[0][2][1] = 8
Data Array[0][2][2] = 9
Data Array[1][0][0] = 10
Data Array[1][0][1] = 11
Data Array[1][0][2] = 12
Data Array[1][1][0] = 13
Data Array[1][1][1] = 14
Data Array[1][1][2] = 15
Data Array[1][2][0] = 16
Data Array[1][2][1] = 17
Data Array[1][2][2] = 18

```

```

123
456
789

```



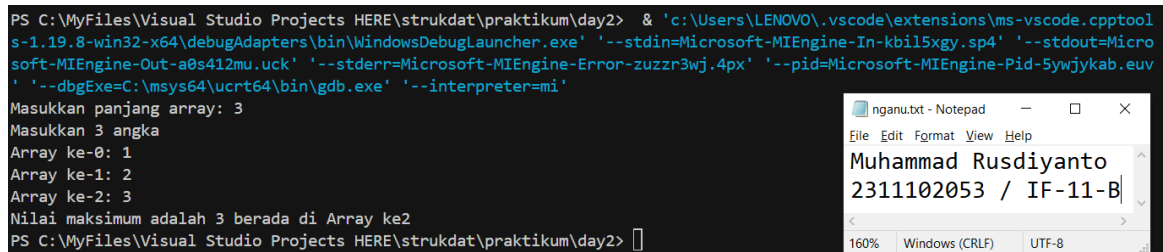
Deskripsi:

Program di atas adalah program simulasi input (pengisian) suatu array 3 dimensi. Dalam program tersebut, ukuran array sudah ditentukan melalui kode yaitu 3 untuk ukuran baris, 3 untuk ukuran kolom, dan 2 untuk banyaknya baris dan kolom. Saat program berjalan, array akan dideklarasikan lalu diakses menggunakan nested looping (perulangan bersarang), dimana terdapat perulangan di dalam perulangan. Dalam program ini, nested looping memiliki 3 tingkat / lapis, dimana yang pertama digunakan untuk mengakses kumpulan baris dan kolom, lalu yang kedua digunakan untuk mengakses baris, dan yang ketiga digunakan untuk mengakses kolom. Selama perulangan berjalan, pengguna akan diminta untuk mengisi nilai array satu per satu. Setelah semua nilai array diinputkan, program akan menjalankan nested looping yang serupa. Hanya saja, looping kali ini digunakan untuk menampilkan nilai - nilai yang sudah dimasukkan tersebut. Setelah semua nilai ditampilkan, program selesai,

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;
}
```

Screenshots Output



```
PS C:\MyFiles\Visual Studio Projects HERE\strukdat\praktikum\day2> & 'c:\Users\LENOVO\.vscode\extensions\ms-vscode.cpptools-1.19.8-win32-x64\debugAdapters\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-kbil5xgy.sp4' '--stdout=Microsoft-MIEngine-Out-a0s412mu.uck' '--stderr=Microsoft-MIEngine-Error-zuzr3wj.4px' '--pid=Microsoft-MIEngine-Pid-5ywjyab.euv' '--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
Masukkan panjang array: 3
Masukkan 3 angka
Array ke-0: 1
Array ke-1: 2
Array ke-2: 3
Nilai maksimum adalah 3 berada di Array ke2
PS C:\MyFiles\Visual Studio Projects HERE\strukdat\praktikum\day2> |
```

Deskripsi:

Program di atas adalah sebuah program yang membolehkan pengguna untuk mengatur ukuran array yang akan mereka isi. Proses pertama yang akan dilakukan program adalah meminta input ukuran dari array yang akan dibuat. Input tersebut akan digunakan saat deklarasi ukuran array. Setelah array dideklarasikan, program akan

looping meminta pengguna untuk memasukkan angka sebanyak panjang array yang mereka masukkan tadi (jika 10 maka memasukkan 10 angka, jika 5 maka memasukkan 5 angka, dst.). Setelah semua angka berhasil dimasukkan, program akan melakukan looping lagi dimana looping kali ini dilakukan untuk mencari nilai maksimum (nilai terbesar) dari kumpulan angka yang dimasukkan tadi. Jika semua angka dalam array telah dicek, maka nilai maksimum akan ditampilkan dan setelah itu program berhenti. Pengecekan disini dilakukan menggunakan looping, dimana pada setiap iterasi, program membandingkan nilai maksimum saat ini (nilai yang disimpan pada variabel maks) dengan nilai array pada indeks ke-i. Nilai array indeks i lebih besar, maka nilai variabel maks akan diubah menjadi nilai array indeks i. Proses ini akan dilakukan ke setiap elemen dalam array.

C. Unguided

Unguided 1

```
#include <iostream>

using namespace std;

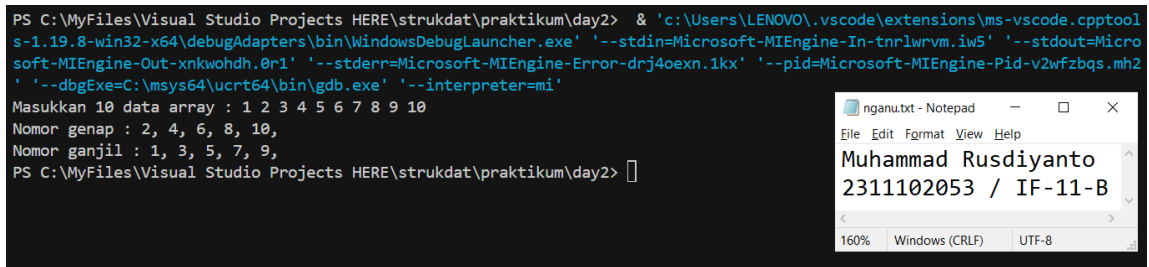
int main() {
    int arr[10];

    cout << "Masukkan 10 data array : ";
    for (int i = 0; i < 10; i++) {
        cin >> arr[i];
    }

    cout << "Nomor genap : ";
    for (int i = 0; i < 10; i++) {
        if (arr[i] % 2 == 0) {
            cout << arr[i] << ", ";
        }
    }

    cout << endl << "Nomor ganjil : ";
    for (int i = 0; i < 10; i++) {
        if (arr[i] % 2 != 0) {
            cout << arr[i] << ", ";
        }
    }
}
```

Screenshots Output



```
PS C:\MyFiles\Visual Studio Projects HERE\strukdat\praktikum\day2> & 'c:\Users\LENOVO\.vscode\extensions\ms-vscode.cpptools-1.19.8-win32-x64\debugAdapters\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-tnrlwrm.iw5' '--stdout=Microsoft-MIEngine-Out-xnkwohdh.0r1' '--stderr=Microsoft-MIEngine-Error-drj4oexn.1kx' '--pid=Microsoft-MIEngine-Pid-v2wfzbs.mh2' '--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
Masukkan 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,
PS C:\MyFiles\Visual Studio Projects HERE\strukdat\praktikum\day2> 
```

nganu.txt - Notepad

Muhammad Rusdiyanto
2311102053 / IF-11-B

160% Windows (CRLF) UTF-8

Deskripsi:

Program di atas adalah sebuah program yang dapat mengelompokkan semua angka dalam array berupa ganjil atau genap. Array yang dideklarasikan dalam program memiliki ukuran 10, jadi array dapat menyimpan 10 angka (integer). Setelah dideklarasikan, program akan looping untuk meminta input angka dari user sebanyak 10 kali (menyesuaikan dari ukuran array supaya semua slot array digunakan). Setelah itu, program akan looping lagi untuk menampilkan angka genap dalam array. Setelah semua nilai genap ditampilkan, program akan melanjutkan dengan looping array lagi untuk mencari nilai ganjil. Proses penentuan genap atau ganjilnya angka dalam program dilakukan menggunakan operator modulus (%), operator sisa bagi). Jadi, ketika suatu angka dibagi dengan dua dan hasilnya 0, maka angka tersebut adalah genap. Sebaliknya, jika hasilnya tidak sama dengan 0, maka angka tersebut ganjil.

Unguided 2

```
#include <iostream>

using namespace std;

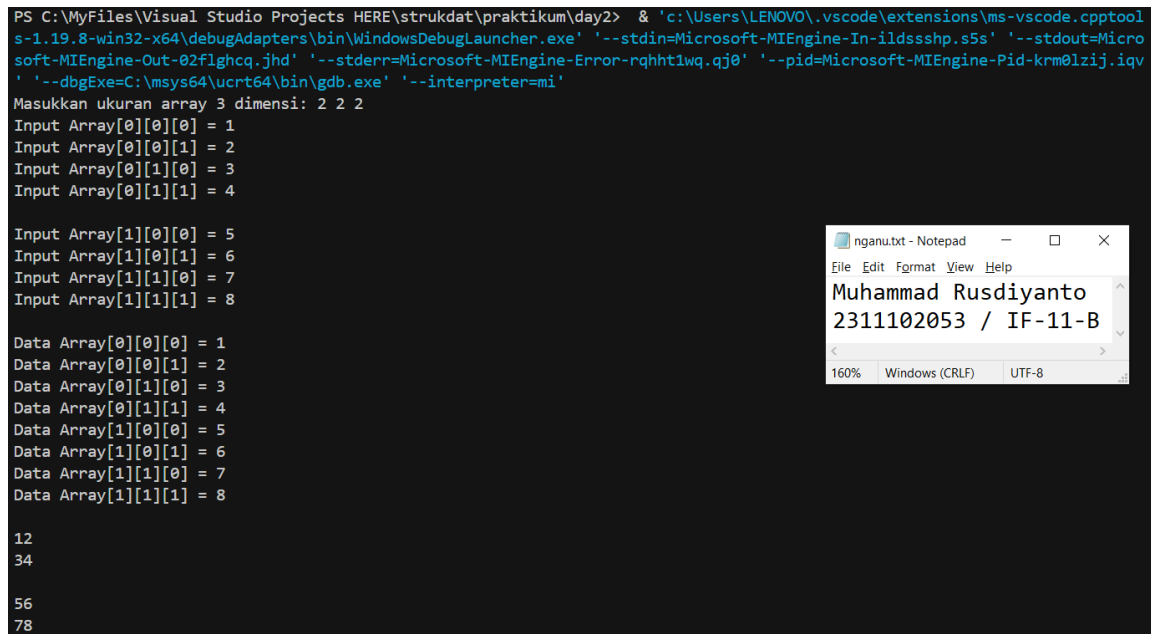
int main() {
    int arrSize[3];
    cout << "Masukkan ukuran array 3 dimensi: ";
    for (int i = 0; i < 3; i++) {
        cin >> arrSize[i];
    }
    int arr[arrSize[0]][arrSize[1]][arrSize[2]];
    for (int x = 0; x < arrSize[0]; x++)
    {
        for (int y = 0; y < arrSize[1]; y++)
        {
            for (int z = 0; z < arrSize[2]; z++)
            {
```

```

        cout << "Input Array[" << x << "]"[" << y << "]"[" << z <<
"] = ";
        cin >> arr[x][y][z];
    }
}
cout << endl;
}
// Output Array
for (int x = 0; x < arrSize[0]; x++)
{
    for (int y = 0; y < arrSize[1]; y++)
    {
        for (int z = 0; z < arrSize[2]; z++)
        {
            cout << "Data Array[" << x << "]"[" << y << "]"[" << z << "]
= " << arr[x][y][z] << endl;
        }
    }
}
cout << endl;
// Tampilan array
for (int x = 0; x < arrSize[0]; x++)
{
    for (int y = 0; y < arrSize[1]; y++)
    {
        for (int z = 0; z < arrSize[2]; z++)
        {
            cout << arr[x][y][z] << ends;
        }
        cout << endl;
    }
    cout << endl;
}
}
}

```


Screenshots Output



```
PS C:\MyFiles\Visual Studio Projects HERE\strukdat\praktikum\day2> & 'c:\Users\LENOVO\.vscode\extensions\ms-vscode.cpptools-1.19.8-win32-x64\debugAdapters\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-ildssshp.s5s' '--stdout=Microsoft-MIEngine-Out-02flghcq.jhd' '--stderr=Microsoft-MIEngine-Error-rqhhtlwq.qj0' '--pid=Microsoft-MIEngine-Pid-krm0lzij.iqv' '--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
Masukkan ukuran array 3 dimensi: 2 2 2
Input Array[0][0][0] = 1
Input Array[0][0][1] = 2
Input Array[0][1][0] = 3
Input Array[0][1][1] = 4

Input Array[1][0][0] = 5
Input Array[1][0][1] = 6
Input Array[1][1][0] = 7
Input Array[1][1][1] = 8

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

12
34

56
78
```

Deskripsi:

Program di atas adalah gabungan dan modifikasi dari program bagian guided 1 dan guided 2. Program ini menggabungkan konsep nested loop untuk mengakses array 3 dimensi dan konsep dimana pengguna dapat memasukkan ukuran array. Pertama, program akan meminta input ukuran array dari pengguna. Untuk program ini, data ukuran array disimpan di dalam array dengan ukuran tiga. Setiap slot elemen digunakan untuk menyimpan data, dimana [0] untuk banyaknya baris dan kolom, [1] untuk baris, dan [2] untuk ukuran kolom. Setelah program selesai memperoleh input dari pengguna, program akan mendeklarasikan array dengan ukuran sesuai dengan yang telah dimasukkan. Kemudian, array diisi menggunakan metode yang sama seperti program guided 1. Setelah itu, semua nilai array akan ditampilkan dengan metode yang sama pula. Yang membedakan nested looping dalam program guided 1 dan program ini nilai yang digunakan sebagai batas looping (semacam tanda supaya looping berhenti). Dalam program guided 1, batas ditulis menggunakan nilai statis seperti 2 dan 3, berhubung ukuran array tidak akan berubah. Karena ukuran array dalam program ini dapat berubah, maka batas yang digunakan menggunakan variabel (dalam hal ini menggunakan nilai input pengguna yang disimpan dalam array). Setelah semua nilai array dioutputkan, maka program selesai.

Unguided 3

```
#include <iostream>

using namespace std;

int main() {
    bool runApp = true;
    int arr[5];
    int max, min, avg = 0, choice;

    cout << "Masukkan 5 data array : ";
    for (int i = 0; i < 5; i++) {
        cin >> arr[i];
    }

    while (runApp) {
        cout << endl << "1. Tampilkan nilai maksimum";
        cout << endl << "2. Tampilkan nilai minimum";
        cout << endl << "3. Tampilkan nilai rata - rata";
        cout << endl << "Masukkan pilihan [1-3] : ";
        cin >> choice;

        switch (choice) {
            case 1 : {
                cout << "Nilai maksimum : ";
                max = arr[0];
                for (int i = 1; i < 5; i++) {
                    if (arr[i] > max) max = arr[i];
                }
                cout << max << endl;
                break;
            }
            case 2: {
                cout << "Nilai minimum : ";
                min = arr[0];
                for (int i = 1; i < 5; i++) {
                    if (arr[i] < min) min = arr[i];
                }
                cout << min << endl;
                break;
            }
            case 3: {
                cout << "Nilai rata - rata : ";
```

```

        for (int i = 0; i < 5; i++) {
            avg += arr[i];
        }
        avg /= 5;
        cout << avg << endl;
        break;
    }
    default: cout << "Input tidak valid!" << endl;
}
}
}

```

Screenshots Output

```

PS C:\MyFiles\Visual Studio Projects HERE\strukdat\praktikum\day2> & 'c:\Users\LENOVO\.vscode\extensions\ms-vscode.cpptools-1.19.8-win32-x64\debugAdapters\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-ogtyg3gb.wbv' '--stdout=Microsoft-MIEngine-Out-og5dvagk.ycs' '--stderr=Microsoft-MIEngine-Error-ymeevt2y.tpa' '--pid=Microsoft-MIEngine-Pid-g3d44krn.3vy' '--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
Masukkan 5 data array : 12 4 7 19 8

1. Tampilkan nilai maksimum
2. Tampilkan nilai minimum
3. Tampilkan nilai rata - rata
Masukkan pilihan [1-3] : 1
Nilai maksimum : 19

1. Tampilkan nilai maksimum
2. Tampilkan nilai minimum
3. Tampilkan nilai rata - rata
Masukkan pilihan [1-3] : 2
Nilai minimum : 4

1. Tampilkan nilai maksimum
2. Tampilkan nilai minimum
3. Tampilkan nilai rata - rata
Masukkan pilihan [1-3] : 3
Nilai rata - rata : 10

```

Deskripsi:

Program di atas adalah program yang berfungsi untuk mencari nilai maksimum, minimum, dan rata - rata dari suatu barisan angka. Barisan angka tersebut disimpan di dalam array dengan ukuran 5, jadi pengguna hanya bisa memasukkan 5 angka saja ke dalam program untuk diproses. Setelah pengguna memasukkan 5 angka, program akan menjalankan looping yang tidak akan pernah berhenti. Hal ini dilakukan supaya pengguna tidak perlu menjalankan program secara berkali - kali untuk menampilkan nilai maksimum, minimum, dan rata - rata. Di dalam looping, program akan menampilkan menu yang berisi pilihan proses untuk menampilkan nilai maksimum, minimum, atau rata - rata. Pengguna diminta untuk memilih salah satu dari pilihan yang ada. Jika pilihan pengguna sesuai, maka program akan menampilkan output yang sesuai pula (jika 1 maka tampil maksimum, jika 2 maka tampil minimum, jika 3 maka tampil rata - rata). Jika tidak sesuai, maka program akan menampilkan pesan error,

Dalam program ini, algoritma pencarian nilai maksimum kurang lebih sama dengan program di bagian guided 2. Begitu pula untuk pencarian nilai minimum, hanya saja ada perubahan sedikit. Dalam algoritma nilai maksimum, program mencari nilai yang lebih besar, sementara untuk algoritma nilai minimum, program mencari nilai yang lebih kecil. Untuk pencarian nilai rata - rata, program menjumlahkan semua angka dalam array dengan menggunakan looping, lalu jumlah tersebut dibagi dengan ukuran array setelah looping. Akhirnya, nilai rata - rata ditampilkan.

D. Kesimpulan

Array merupakan sebuah tipe data yang dapat menyimpan banyak value (nilai) dengan tipe data yang sama. Array memiliki dimensi yang dapat diatur sesuai dengan keperluan. Interaksi dengan array (baik itu input maupun output) pada umumnya dilakukan menggunakan looping (perulangan) untuk mengurangi beban penulisan kode, namun bukan berarti nilai array tidak dapat diakses tanpa looping. Semakin banyak dimensi array, maka semakin banyak pula nested looping (looping di dalam looping) yang diperlukan. Array akan menjadi sangat berguna ketika menghadapi kondisi dimana program perlu menangani banyak input.

E. Referensi

Asisten Praktikum, “Modul 2 Array”