

**LAPORAN PRAKTIKUM  
STRUKTUR DATA DAN ALGORITMA**

**MODUL II  
ARRAY**



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

**Disusun oleh:**

**FALAH ASBI PANGESTU (2311102045)**

**IF-11-B**

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

**2024**

# BAB I

## DASAR TEORI

### A. 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

## 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 II

### GUIDED

#### LATIHAN – GUIDED

##### 1. Guided 1

Program input array tiga dimensi.

##### 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;

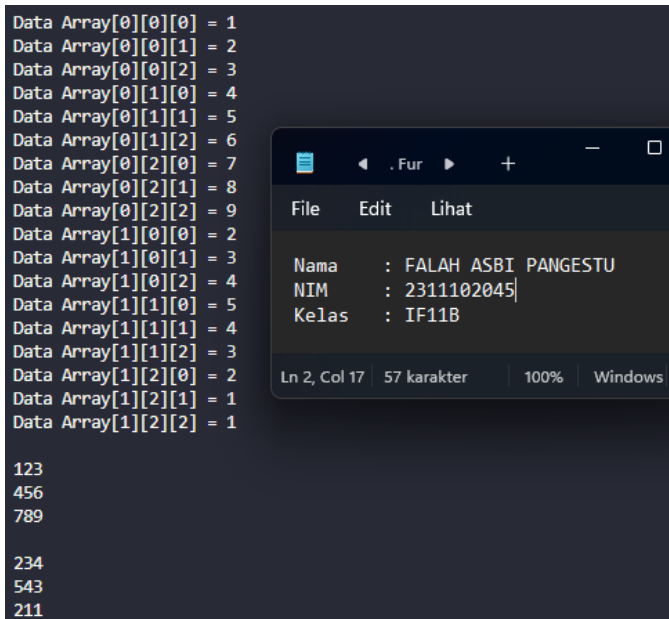
    // Tampilan array
    for (int x = 0; x < 2; x++)
    {
        for (int y = 0; y < 3; y++)
        {
            for (int z = 0; z < 3; z++)
            {
                cout << arr[x][y][z] << ends;
            }

            cout << endl;
        }

        cout << endl;
    }
}

```

## Screenshoot program



The screenshot shows the output of a C++ program and a Notepad window. The program output displays a 2x3x3 array of integers, with values ranging from 1 to 9. The Notepad window shows the text "Fur" and a menu with "File", "Edit", and "Lihat". The text "Nama : FALAH ASBI PANGESTU", "NIM : 2311102045", and "Kelas : IF11B" is visible in the Notepad window.

```

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] = 2
Data Array[1][0][1] = 3
Data Array[1][0][2] = 4
Data Array[1][1][0] = 5
Data Array[1][1][1] = 4
Data Array[1][1][2] = 3
Data Array[1][2][0] = 2
Data Array[1][2][1] = 1
Data Array[1][2][2] = 1

123
456
789

234
543
211

```

### Deskripsi program

Program menciptakan sebuah array tiga dimensi berukuran 2 x 3 x 3. Kemudian, dengan menggunakan loop, meminta pengguna memasukkan nilai ke dalam array sesuai dengan indeksinya. Setelah semua nilai dimasukkan, program menampilkan kembali nilai-nilai tersebut, terlebih dahulu menampilkan indeks masing-masing nilai, dan kemudian menampilkan semua nilai tanpa memperhatikan struktur tiga dimensinya.

#### 2. Guided 2

Program mencari nilai maksimal pada array

### 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 + 1) << ": ";

        cin >> array[i];

    }

    maks = array[0];

    for (i = 0; i < a; i++) {

        if (array[i] > maks) {

            maks = array[i];

            lokasi = i;

        }

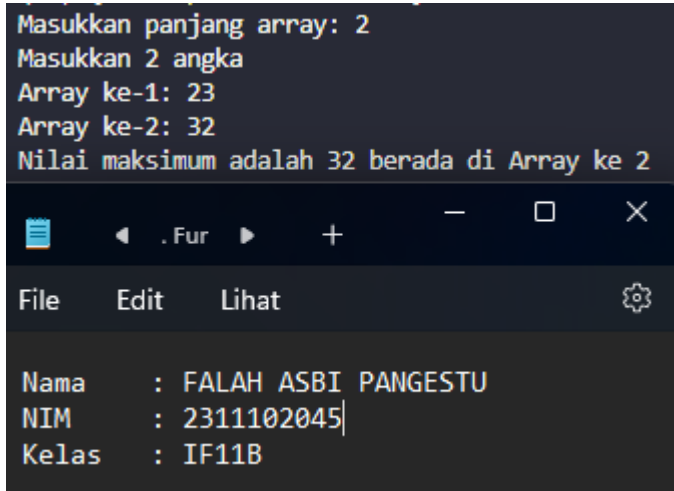
    }

    cout << "Nilai maksimum adalah " << maks << " berada di Array ke " << (lokasi + 1) << endl;

    return 0;

}
```

### Screenshoot program



```
Masukkan panjang array: 2
Masukkan 2 angka
Array ke-1: 23
Array ke-2: 32
Nilai maksimum adalah 32 berada di Array ke 2

Nama      : FALAH ASBI PANGESTU
NIM       : 2311102045
Kelas    : IF11B
```

### Deskripsi program

Pertama, program meminta pengguna untuk menentukan panjang array, kemudian membuat array dengan ukuran yang sesuai, dan kemudian meminta pengguna untuk memasukkan nilai ke dalamnya. Setelah semua nilai dimasukkan, program menggunakan sebuah loop untuk membandingkan setiap elemen array dengan nilai maksimum yang disimpan sebelumnya; jika sebuah elemen dari array lebih besar dari nilai maksimum yang disimpan sebelumnya, maka nilai maksimum akan ditambahkan.



## BAB III

### UNGUIDED

#### TUGAS – UNGUIDED

##### 1. Unguided 1

Buatlah program untuk menampilkan Output seperti berikut dengan data yang diinputkan oleh user!

##### Source Code

```
#include <iostream>

using namespace std;

int main() {
    int numb[10] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};

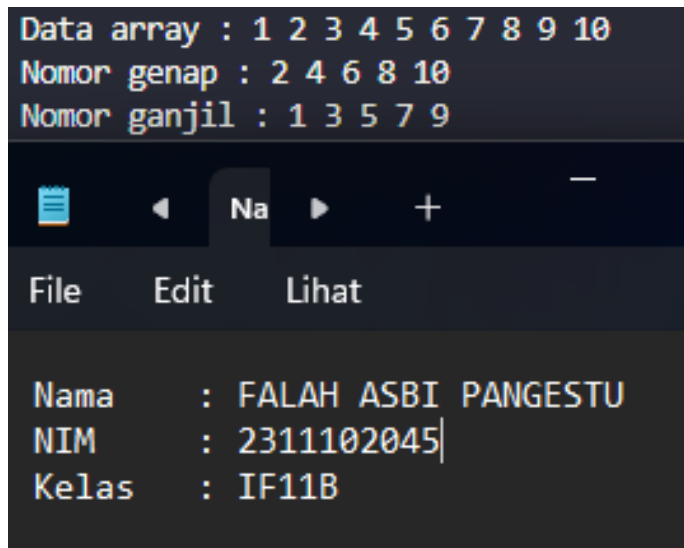
    cout << "Data array :";
    for (int i = 0; i < 10; ++i) {
        cout << " " << numb[i];
    }
    cout << endl;

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

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

```
    return 0;
}
```

### Screenshoot program



### Deskripsi program

Program memulai dengan meminta pengguna untuk menentukan panjang sebuah array, membuat array dengan ukuran yang sesuai, dan kemudian meminta pengguna untuk memasukkan nilai ke dalamnya. Setelah semua nilai dimasukkan, program menggunakan sebuah loop untuk membandingkan setiap elemen array dengan nilai maksimum yang disimpan sebelumnya; jika sebuah elemen dari array lebih besar dari nilai maksimum yang disimpan sebelumnya, maka nilai maksimum akan ditambahkan.

### 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 (x y z): ";
    cin >> x_size >> y_size >> z_size;
```

```

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

cout << "Masukkan elemen array:\n";
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 <<
"]: ";
            cin >> arr[x][y][z];
        }
    }
}

cout << "Elemen array yang dimasukkan:\n";
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;
}

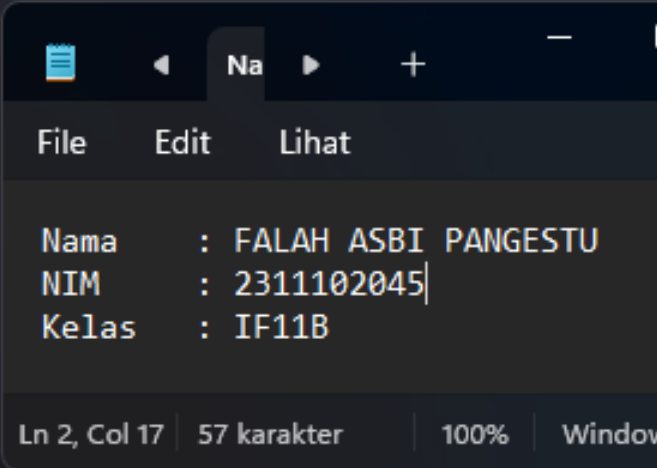
```

### Screenshoot program

```
Elemen array yang dimasukkan:
2 3 1
2 3 2
3 2 5

6 7 5
4 3 6
43 3 4

3 2 5
6 3 2
6 4 3
```



The screenshot shows a C++ IDE window. On the left, there is a list of 3D array elements. On the right, a menu is open with 'Na' selected. Below the menu, there is a form with three fields: 'Nama' (filled with 'FALAH ASBI PANGESTU'), 'NIM' (filled with '2311102045'), and 'Kelas' (filled with 'IF11B'). The status bar at the bottom indicates 'Ln 2, Col 17', '57 karakter', '100%', and 'Window'.

### Deskripsi program

Program meminta pengguna untuk memasukkan ukuran array tiga dimensi (x, y, z), membuat array dengan ukuran yang sesuai, dan kemudian meminta pengguna untuk memasukkan nilai untuk setiap elemen array. Setelah semua nilai dimasukkan, program menampilkan kembali nilai-nilai untuk verifikasi.

### 3. Unguided 3

Buatlah program menu untuk mencari nilai Maksimum, Minimum dan Nilai rata-rata dari suatu array dengan input yang dimasukan oleh user!

### Source code

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

int main() {
    int arr[100];
    int n;

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

    cout << "Masukkan elemen-elemen array:\n";
    for (int i = 0; i < n; ++i) {
        cout << "Elemen ke-" << i+1 << ": ";
        cin >> arr[i];
    }
}
```

```
char choice;

cout << "\nPilih operasi:\n";

cout << "1. Mencari nilai maksimum\n";
cout << "2. Mencari nilai minimum\n";
cout << "3. Mencari nilai rata-rata\n";
cout << "Pilihan Anda (1/2/3): ";
cin >> choice;

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

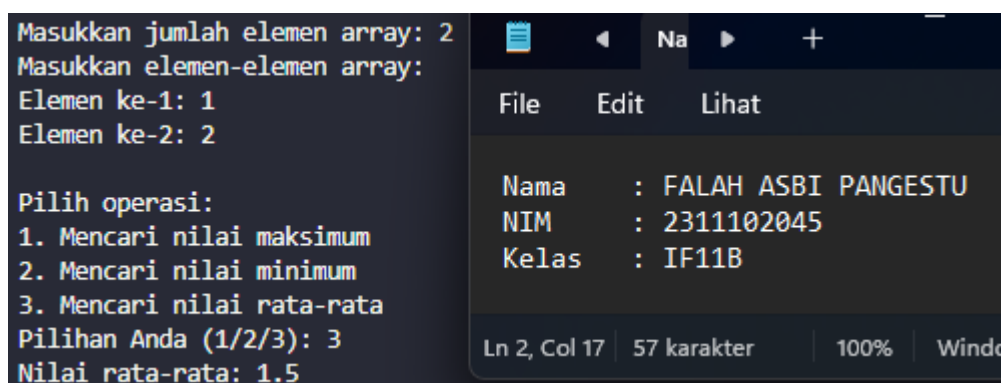
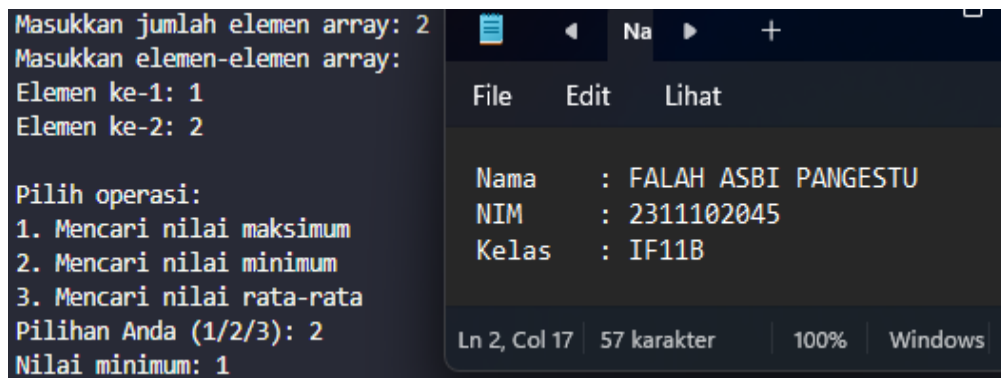
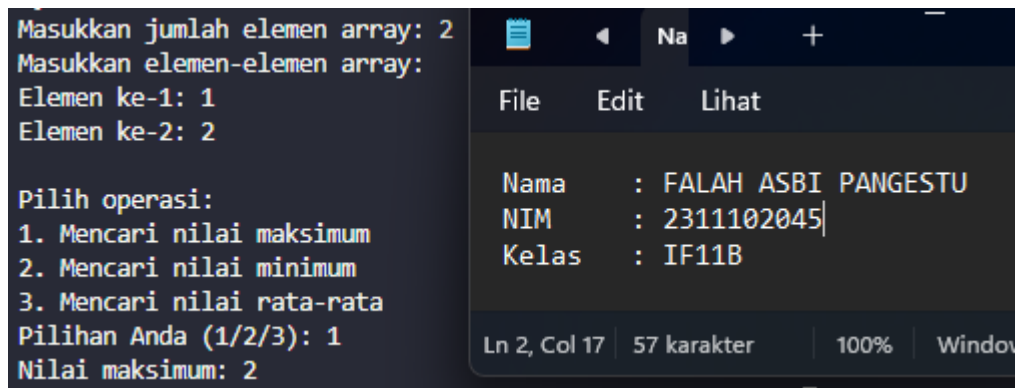
```

        double average = static_cast<double>(sum) / n;
        cout << "Nilai rata-rata: " << average << endl;
        break;
    }
    default:
        cout << "Pilihan tidak valid!\n";
    }

    return 0;
}

```

### Screenshoot program



**Deskripsi program**

Pengguna diminta untuk memasukkan jumlah elemen array dan elemen-elemennya sendiri. Setelah memasukkan semua elemen, program menggunakan struktur switch-case untuk menjalankan operasi yang dipilih oleh pengguna. Untuk operasi mencari nilai maksimum, program melakukan iterasi melalui array untuk menemukan nilai maksimum; untuk operasi mencari nilai minimum, program melakukan hal yang sama, tetapi mencari nilai minimum. Untuk operasi mencari nilai rata-rata, program menghitung total nilai semua elemen dalam array, kemudian membagi hasilnya dengan jumlah elemen untuk mendapatkan nilai rata-rata.

## **BAB IV**

### **KESIMPULAN**

Array adalah salah satu struktur data paling dasar dalam bahasa pemrograman C++ yang digunakan untuk menyimpan kumpulan data yang mirip dalam satu variabel. Setiap elemen dalam array dapat diakses melalui indeks numerik yang dimulai dari 0, yang memudahkan akses data. Saat mendeklarasikan array, perlu diperhatikan bahwa ukuran array harus sesuai dengan jumlah elemen yang akan disimpan di dalamnya. Ukuran ini tetap dan tidak dapat diubah setelah dideklarasikan, jadi harus dipilih dengan hati-hati agar tidak ada pemborosan memori atau kekurangan memori yang dapat menyebabkan program tidak stabil.

Karena setiap elemen dapat diakses secara terpisah dan operasi dapat diterapkan pada setiap elemen secara terpisah, array juga memungkinkan pemrosesan data paralel. Namun, perlu diingat bahwa array memiliki beberapa keterbatasan, seperti ukurannya yang tetap dan kesulitan mengubah ukurannya setelah deklarasi. Oleh karena itu, gunakan dengan hati-hati, terutama untuk aplikasi yang membutuhkan fleksibilitas penyimpanan data.



## **DAFTAR PUSTAKA**

Asisten Praktikum, “Modul 2 Array”, Learning Management System, 2024