

**LAPORAN PRAKTIKUM STRUKTUR DATA DAN  
PEMROGRAGAMAN**

**MODUL 2**

**ARRAY**



**Disusun oleh :**

Rafa Aldhino Fatin

2311102023

IF-11-A

**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**

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 adalah beberapa jenis array :

1. Array satu dimensi
2. Array dua dimensi
3. Array multidimensi
4. Array empat dimensi
5. Array lima dimensi

#### **ARRAY SATU DIMENSI**

Array satu dimensi adalah tipe variabel yang terdiri dari kumpulan data dengan tipe data 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.

#### **ARRAY DUA DIMENSI**

Array dua dimensi adalah variabel yang terdiri dari kumpulan array satu dimensi dengan tipe yang sama 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.

#### **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 dan seterusnya.

## BAB III

### GUIDED

#### GUIDED 1

#### 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] << endl;
            }
        }
        cout << endl;
    }
}
```

## SCREENSHOOT PROGRAM

The screenshot shows a Visual Studio IDE with a C++ project named 'pertemuan 3 array'. The Explorer pane on the left shows the project structure with files like 'modul1.cpp', 'modul1.exe', 'modul2.cpp', 'modul2.exe', 'test.cpp', and 'test.exe'. The main editor displays the source code for 'modul1.cpp', which includes a 3D array 'arr' of size 3x3x3. The code uses nested loops to iterate through the array and print its elements. The output window at the bottom shows the execution results, including the command prompt path and the output of the program, which displays the values of the array elements in a row-major order.

```
int main()
{
    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] << endl;
            }
            cout << endl;
        }
        cout << endl;
    }
}
```

```
PS E:\KULIAH\SOURCE CODE\SEMESTER 2\praktikum struktur data\pertemuan 3 array> cd "e:\KULIAH\SOURCE CODE\SEMESTER 2\praktikum struktur data\pertemuan 3 array" & if ($?) { g++ modul1.cpp -o modul1 } ; if ($?) { .\modul1 }
Input Array[0][0][0] = 5
Input Array[0][0][1] = 2
Input Array[0][0][2] = 6
Input Array[0][1][0] = 2
Input Array[0][1][1] = 7
Input Array[0][1][2] = 3
Input Array[0][2][0] = 1
Input Array[0][2][1] = 5
Input Array[0][2][2] = 6
Input Array[1][0][0] = 8
Input Array[1][0][1] = 3
Input Array[1][0][2] = 4
Input Array[1][1][0] = 2
Input Array[1][1][1] = 5
Input Array[1][1][2] = 6
Input Array[1][2][0] = 2
Input Array[1][2][1] = 7
Input Array[1][2][2] = 8
Data Array[0][0][0] = 5
```

## DESKRIPSI PROGRAM

Program c++ ini mendemonstrasikan cara kerja array tiga dimensi. Pengguna diminta untuk mengisi nilai-nilai kedalam array tiga dimensi berukuran 2x3x3 melalui input keyboard. Setelah data dimasukkan, program menampilkan nilai-nilai yang telah diinput. Program juga menampilkan isi array secara langsung tanpa label, di mana nilai nilai disusun berurutan tanpa spasi di antara elemen dalam satu baris dan menggunakan baris baru untuk setiap dimensi. Baris `int arr[2][3][3]` mendeklarasikan array `arr` dengan 3 dimensi.

## GUIDED 2

## SOURCE CODE

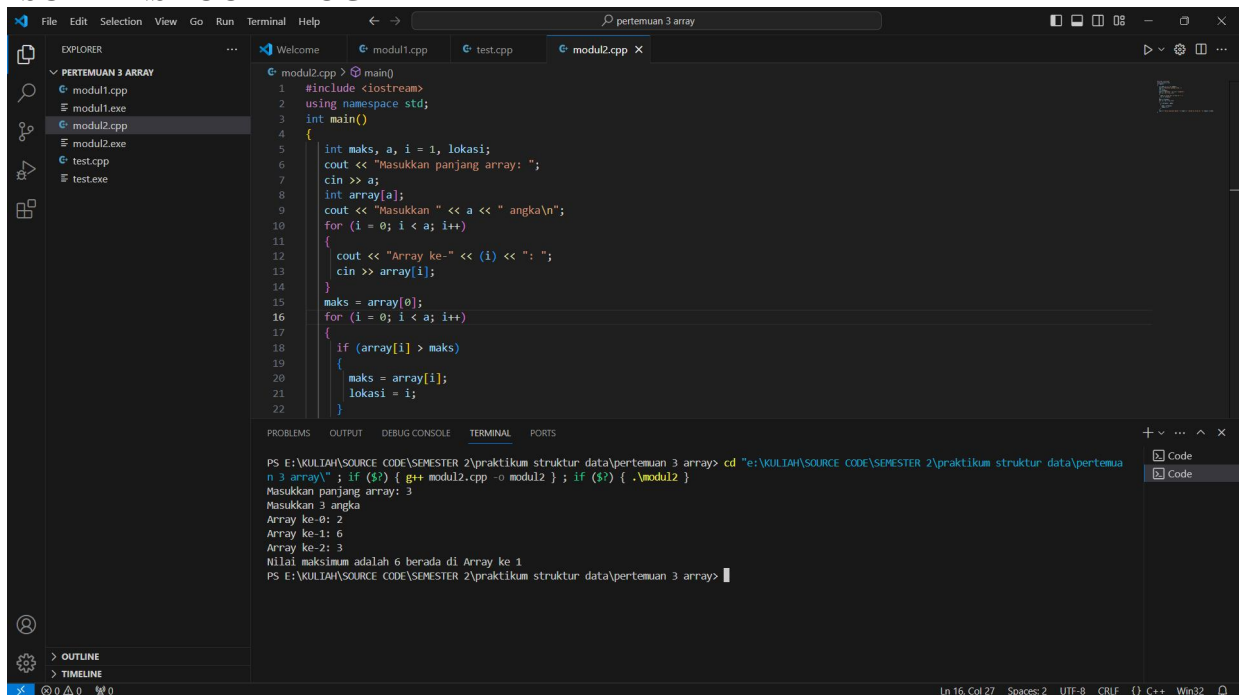
```
#include <iostream>
using namespace std;
int main()
{
    int maks, a, i = 1, lokasi;
    cout << "Masukkan panjang array: ";
    cin >> a;
    int array[a];
    cout << "Masukkan " << a << " angka\n";
    for (i = 0; i < a; i++)
    {
        cout << "Array ke-" << (i) << ": ";
        cin >> array[i];
    }
    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



## DESKRIPSI PROGRAM

Pertama, program meminta pengguna untuk memasukkan panjang array yang diinginkan. Setelah itu, program meminta pengguna untuk memasukkan nilai-nilai ke dalam array tersebut. Kemudian, program mencari nilai maksimum dari array tersebut dengan melakukan iterasi pada setiap elemen array dan membandingkannya dengan nilai maksimum sementara. Jika nilai elemen array tersebut lebih besar dari nilai maksimum sementara, maka nilai maksimum sementara akan diperbarui beserta dengan indeks lokasinya dalam array. Setelah selesai mencari, program akan menampilkan nilai maksimum beserta dengan lokasinya dalam array.

# BAB IV

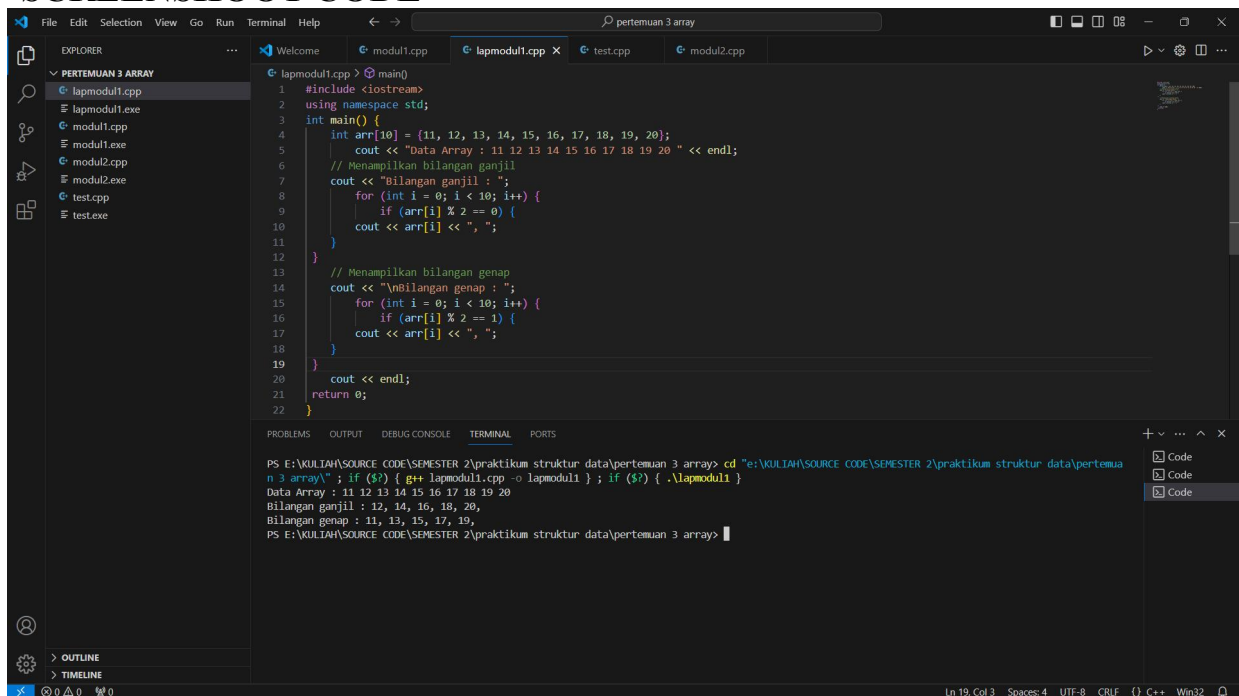
## UNGUIDED

### UNGUIDED 1

### SOURCE CODE

```
#include <iostream>
using namespace std;
int main() {
    int arr[10] = {11, 12, 13, 14, 15, 16, 17, 18, 19, 20};
    cout << "Data Array : 11 12 13 14 15 16 17 18 19 20 " << endl;
    // Menampilkan bilangan ganjil
    cout << "Bilangan ganjil : ";
    for (int i = 0; i < 10; i++) {
        if (arr[i] % 2 == 0) {
            cout << arr[i] << ", ";
        }
    }
    // Menampilkan bilangan genap
    cout << "\nBilangan genap : ";
    for (int i = 0; i < 10; i++) {
        if (arr[i] % 2 == 1) {
            cout << arr[i] << ", ";
        }
    }
    cout << endl;
    return 0;
}
```

### SCREENSHOOT CODE



## DESKRIPSI PROGRAM

Program ini menampilkan elemen bilangan dari sebuah array dan mengelompokkannya berdasarkan bilangan genap dan ganjil. Pertama, program mendeklarasikan array arr berisi 10 elemen angka. Kemudian, program menampilkan seluruh isi arr. Selanjutnya, program menggunakan dua perulangan for terpisah. Perulangan pertama mencari bilangan genap dengan mengecek apakah elemen arr[i] habis dibagi 2 ( $\% 2 == 0$ ). Jika genap, program akan mencetak angkanya. Perulangan kedua mencari bilangan ganjil dengan mengecek sisa bagi 1 ( $\% 2 == 1$ ).

## UNGUIDED 2

### SOURCE CODE

```
#include <iostream>
using namespace std;
int main() {
    int x_RAFA, y_ALDHINO, z_FATIN;
    cout << "Masukkan jumlah atau ukuran array dalam dimensi x: ";
    cin >> x_RAFA;
    cout << "Masukkan jumlah atau ukuran array dalam dimensi y: ";
    cin >> y_ALDHINO;
    cout << "Masukkan jumlah atau ukuran array dalam dimensi z: ";
    cin >> z_FATIN;

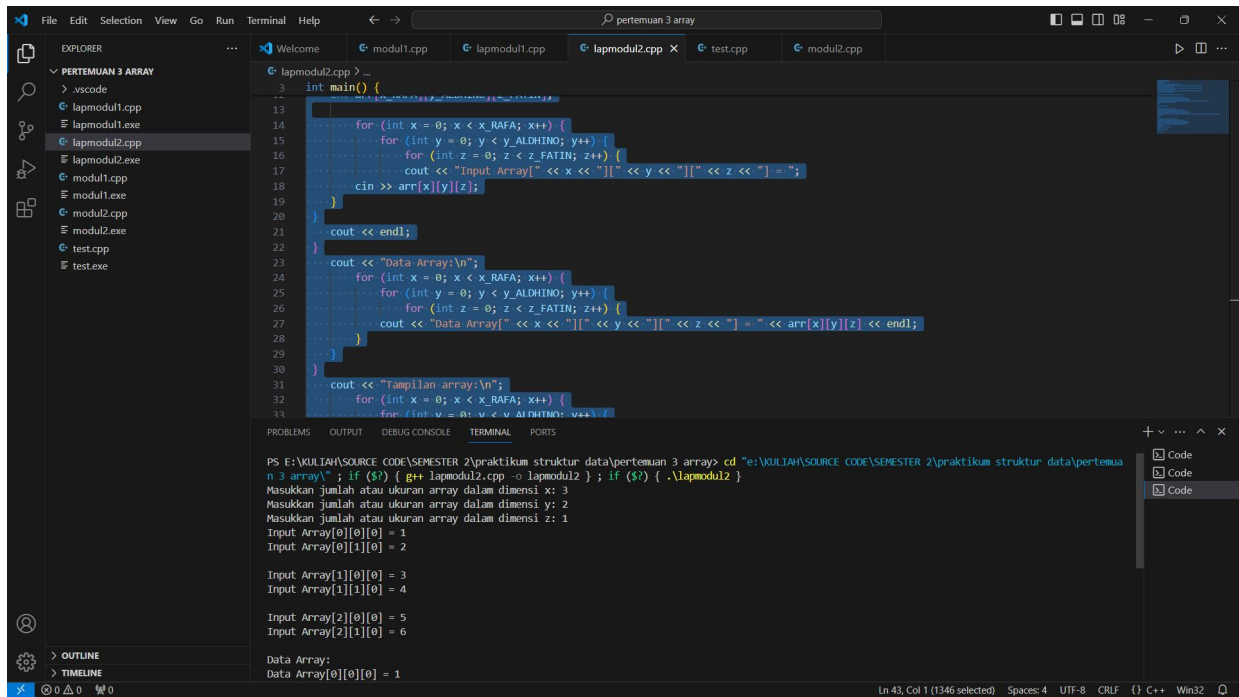
    int arr[x_RAFA][y_ALDHINO][z_FATIN];

    for (int x = 0; x < x_RAFA; x++) {
        for (int y = 0; y < y_ALDHINO; y++) {
            for (int z = 0; z < z_FATIN; z++) {
                cout << "Input Array[" << x << "][" << y << "][" << z << "] = ";
                cin >> arr[x][y][z];
            }
        }
    }
    cout << endl;
}

cout << "Data Array:\n";
for (int x = 0; x < x_RAFA; x++) {
    for (int y = 0; y < y_ALDHINO; y++) {
        for (int z = 0; z < z_FATIN; z++) {
            cout << "Data Array[" << x << "][" << y << "][" << z << "] = " << arr[x][y][z] << endl;
        }
    }
}
```



# SCREENSHOOT PROGRAM



```
1  int main() {
2      int x, y, z;
3      int RAFA = 3, ALDHINO = 3, FATIN = 3;
4      int arr[RAFA][ALDHINO][FATIN];
5
6      for (int x = 0; x < RAFA; x++) {
7          for (int y = 0; y < ALDHINO; y++) {
8              for (int z = 0; z < FATIN; z++) {
9                  cout << "Input Array[" << x << "][" << y << "][" << z << "] = ";
10                 cin >> arr[x][y][z];
11             }
12         }
13     }
14     cout << endl;
15
16     cout << "Data Array:\n";
17     for (int x = 0; x < RAFA; x++) {
18         for (int y = 0; y < ALDHINO; y++) {
19             for (int z = 0; z < FATIN; z++) {
20                 cout << "Data Array[" << x << "][" << y << "][" << z << "] = " << arr[x][y][z] << endl;
21             }
22         }
23     }
24
25     cout << "Tampilan array:\n";
26     for (int x = 0; x < RAFA; x++) {
27         for (int y = 0; y < ALDHINO; y++) {
28             for (int z = 0; z < FATIN; z++) {
29                 cout << arr[x][y][z] << " ";
30             }
31         }
32     }
33 }
```

```
PS E:\KULIAH\SOURCE CODE\SEMESTER 2\praktikum struktur data\pertemuan 3 array> cd "e:\KULIAH\SOURCE CODE\SEMESTER 2\praktikum struktur data\pertemuan 3 array\" ; if ($?) { g++ lapmodul2.cpp -o lapmodul2 } ; if ($?) { .\lapmodul2 }
Masukkan jumlah atau ukuran array dalam dimensi x: 3
Masukkan jumlah atau ukuran array dalam dimensi y: 2
Masukkan jumlah atau ukuran array dalam dimensi z: 1
Input Array[0][0][0] = 1
Input Array[0][1][0] = 2

Input Array[1][0][0] = 3
Input Array[1][1][0] = 4

Input Array[2][0][0] = 5
Input Array[2][1][0] = 6

Data Array:
Data Array[0][0][0] = 1
```

## DESKRIPSI / FUNGSI PROGRAM

Program menggunakan tiga perulangan for berlapis untuk mengisi setiap elemen array arr. Perulangan ini berjalan dari 0 hingga nilai dimensi masing-masing. Di dalam perulangan, program meminta pengguna untuk memasukkan nilai untuk setiap elemen array. Program menampilkan data array dalam dua format. Pertama, program menampilkan data array dalam bentuk tabel dengan koordinat setiap elemen. Kedua, program menampilkan data array dalam bentuk tabel sederhana tanpa koordinat.

## UNGUIDED 3

## SOURCE

## CODE

```

#include <iostream>
using namespace std;
int main() {
    int panjang, i = 0;
    double total = 0;
    cout << "Masukkan panjang array: ";
    cin >> panjang;

    int array[panjang];

    cout << "Masukkan " << panjang << " angka\n";

    // Memasukkan elemen array
    for (i = 0; i < panjang; i++) {
        cout << "Array ke-" << (i + 1) << ": ";
        cin >> array[i];
        total += array[i];
    }

    // Mencari nilai maksimum
    int maks = array[0];
    int lokasi_max = 0;
    for (i = 1; i < panjang; i++) {
        if (array[i] > maks) {
            maks = array[i];
            lokasi_max = i;
        }
    }

    // Mencari nilai minimum
    int min = array[0];
    int lokasi_min = 0;
    for (i = 1; i < panjang; i++) {
        if (array[i] < min) {
            min = array[i];
            lokasi_min = i;
        }
    }

    // Menghitung rata-rata
    double rata_rata = total / panjang;

    // Output hasil
    cout << "Nilai maksimum adalah " << maks << " berada di Array ke-" << (lokasi_max + 1) << endl;
    cout << "Nilai minimum adalah " << min << " berada di Array ke-" << (lokasi_min + 1) << endl;
    cout << "Nilai rata-rata adalah " << rata_rata << endl;
    return 0;
}

```

## SCREENSHOOT PROGRAM

```
3 int main() {
18
19
20 // Mencari nilai maksimum
21 int maks = array[0];
22 int lokasi_max = 0;
23 for (i = 1; i < panjang; i++) {
24     if (array[i] > maks) {
25         maks = array[i];
26         lokasi_max = i;
27     }
28 }
29
30 // Mencari nilai minimum
31 int min = array[0];
32 int lokasi_min = 0;
33 for (i = 1; i < panjang; i++) {
34     if (array[i] < min) {
35         min = array[i];
36         lokasi_min = i;
37     }
38 }
```

```
PS E:\KULIAH\SOURCE CODE\SEMESTER 2\praktikum struktur data\pertemuan 3 array> cd "E:\KULIAH\SOURCE CODE\SEMESTER 2\praktikum struktur data\pertemuan 3 array\"; if ($?) { g++ lapmodul3.cpp -o lapmodul3 }; if ($?) { .\lapmodul3 }
Masukkan panjang array: 3
Masukkan 3 angka
Array ke-1: 4
Array ke-2: 3
Array ke-3: 7
Nilai maksimum adalah 7 berada di Array ke-3
Nilai minimum adalah 3 berada di Array ke-2
Nilai rata-rata adalah 4.66667
PS E:\KULIAH\SOURCE CODE\SEMESTER 2\praktikum struktur data\pertemuan 3 array>
```

## DESKRIPSI PROGRAM

Di dalam perulangan, program membandingkan setiap elemen array dengan nilai maksimum dan minimum saat ini. Jika elemen lebih besar dari nilai maksimum saat ini, maka elemen tersebut menjadi nilai maksimum baru. Lalu Program menghitung rata-rata dengan menjumlahkan semua elemen array dan membaginya dengan panjang array. Dan Program menampilkan nilai maksimum, minimum, dan rata-rata yang telah dihitung.

## BAB V

## KESIMPULAN

Array adalah alat yang sangat penting untuk mengorganisir dan mengelola data dalam program komputer. Array memungkinkan kita untuk menyimpan data secara efisien dan mudah diakses. Memahami jenis-jenis array dan cara kerjanya adalah langkah penting dalam mempelajari struktur data dan algoritma. Ada beberapa jenis array yang pertama adalah array satu dimensi yang dimana menyimpan data dalam satu baris, seperti daftar nama. Yang kedua adalah array dua dimensi yang dimana menyimpan data dalam tabel, seperti tabel nilai mata pelajaran. Yang ketiga adalah array multi dimensi yang dimana menyimpan data dalam struktur yang lebih kompleks, seperti data spasial (3 dimensi) atau data waktu (4 dimensi).

## DAFTAR PUSTAKA

Karumanchi, N. (2016). Data Structures and algorithms made easy: Concepts, problems, Interview Questions. CareerMonk Publications. TylerMSFT. (n.d.). Collections (C++/CX). diakses dari <https://learn.microsoft.com/en-us/cpp/cppcx/collections-c-cx?view=msvc-170>