

**LAPORAN PRAKTIKUM STRUKTUR
DATA DAN ALGORITMA**

**MODUL 2
ARRAY**



**Disusun Oleh :
Muhammad Hamzah Haifan Ma'ruf
231102091**

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

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

A. Dasar Teori

Array adalah kumpulan-kumpulan variabel yang menyimpan data dengan tipe yang sama atau data-data yang tersusun secara linear dimana di dalamnya terdapat elemen dengan tipe yang sama. Indeks dalam array menyatakan elemen yang disimpan dan panjang atau length menyatakan total elemen yang tersimpan.

Indeks dari elemen array baik bahasa Java maupun C++ dimulai dari 0, bukan 1. Dalam array, untuk membedakan satu variabel dengan variabel lain berdasarkan subscript, bilangan dalam kurung siku [...] disebut subscript, dengan subscript masing-masing elemen dapat diakses.

Dalam menyusun fungsi array ada tiga struktur, yaitu kumpulan data dengan tipe sama, gunakan indeks untuk mengakses setiap elemen, dan simpan di tempat yang bersambungan. Bisa menggunakan salah satu cara deklarasi seperti diatas. Lebih banyak yang menggunakan cara pertama, cara kedua membuat larik di bahasa pemrograman C yang selanjutnya akan diadopsi di Java.

Array: Mempunyai sebuah nilai, misalkan ada 4 data maka cukup menggunakan satu nama variabel `x[4]`

Variabel biasa: Dihubungkan dengan sebuah nilai, misal ada 4 data maka menggunakan nama variabel `x1, x2, x3, x4`

Tiga hal penting dalam deklarasi array :

- Nama array
- Range dari subskrip
- Type data array

Cara mengakses elemen dengan data secara acak menggunakan array, operasi yang digunakan pada array, yaitu:

1. Operasi penyisipan atau insertion: Memasukkan data ke dalam elemen array, dengan menggunakan operator assignment.
2. Operasi pengaksesan atau access: dengan indeks tertentu pada array dilakukan upaya pengambilan nilai dari elemen
3. Operasi pencarian atau searching: menentukan nilai tertentu pada array dengan cara membandingkan elemen yang diakses dengan nilai yang akan dicari

Jenis-jenis Array

1. Array satu dimensi

Array satu dimensi adalah jenis array dasar yang terdiri dari beberapa kolom elemen. Dalam satu baris tersusun dari beberapa elemen-elemen yang sama. Keunggulan dari array satu dimensi adalah mudah digunakan dan mudah dibaca sehingga paling umum digunakan. Perlu diingat elemen pertama ditandai dengan indeks 0, elemen keempat di indeks 3, dan seterusnya.

TipeData NamaArray [jumlah elemen]

2. Array dua dimensi

Array dua dimensi merupakan perluasan dari array satu dimensi sehingga array dua dimensi terdiri dari kolom dan baris atau berbentuk matrix. Maka harus perhatikan nomor baris dan kolom.

TipeData NamaArray [jumlah baris][jumlah kolom]

Contohnya algoritma [5][4] menyatakan nilai 5 berarti jumlah baris, nilai 4 berarti kolom. artinya array tersebut menampung 20 elemen.

Kelebihan dari array dua dimensi yaitu dapat menyimpan data yang lebih kompleks.

3. Array Multidimensi

Array multidimensi digunakan untuk ke array yang lebih dari dua dimensi atau lebih. Bentuknya memiliki banyak dimensi sehingga untuk menentukan posisi elemen data tidak menggunakan indeks namun menggunakan key atau string. String merupakan array dari karakter.

Dalam penulisan string menggunakan tanda petik ganda "...", sedangkan karakter menggunakan tanda petik tunggal '...'. Pada string selalu diakhiri dengan angka nol dalam penulisan. Contohnya :

Count <<"jklmn\0opqrs".

Karena ada angka nol sebelum huruf o maka yang akan muncul dimonitor yaitu jklmn.

B. 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 << "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;
}
return 0;

```

Screenshots Output :

```

Input Array[0][0][0]: 1
Input Array[0][0][1]: 1
Input Array[0][0][2]: 1
Input Array[0][1][0]: 1
Input Array[0][1][1]: 1
Input Array[0][1][2]: 1
Input Array[0][2][0]: 1
Input Array[0][2][1]: 1
Input Array[0][2][2]: 1

Input Array[1][0][0]: 1
Input Array[1][0][1]: 1
Input Array[1][0][2]: 1
Input Array[1][1][0]: 1
Input Array[1][1][1]: 1
Input Array[1][1][2]: 1
Input Array[1][2][0]: 1
Input Array[1][2][1]: 1
Input Array[1][2][2]: 1

Array[0][0]: 1
Array[0][1]: 1
Array[0][2]: 1
Array[0][1]: 1
Array[0][1]: 1
Array[0][2]: 1
Array[0][2]: 1
Array[0][2]: 1
Array[1][0]: 1
Array[1][0]: 1
Array[1][0]: 1
Array[1][1]: 1
Array[1][1]: 1
Array[1][1]: 1
Array[1][2]: 1
Array[1][2]: 1
Array[1][2]: 1

```

Deskripsi :

Program di atas merupakan sebuah program C++ yang mendemonstrasikan penggunaan array tiga dimensi. Program tersebut meminta pengguna untuk memasukkan elemen-elemen ke dalam array tiga dimensi dengan ukuran 2x3x3, kemudian mencetak elemen-elemen tersebut ke layar dalam format yang terstruktur. Pertama, program meminta input dari pengguna untuk setiap elemen array menggunakan tiga perulangan bersarang. Setelah itu, program mencetak isi array ke layar dua kali: pertama-tama dalam format yang terstruktur dengan menampilkan indeks setiap elemen, dan kedua dalam format yang lebih sederhana hanya menampilkan isi array tanpa indeks.

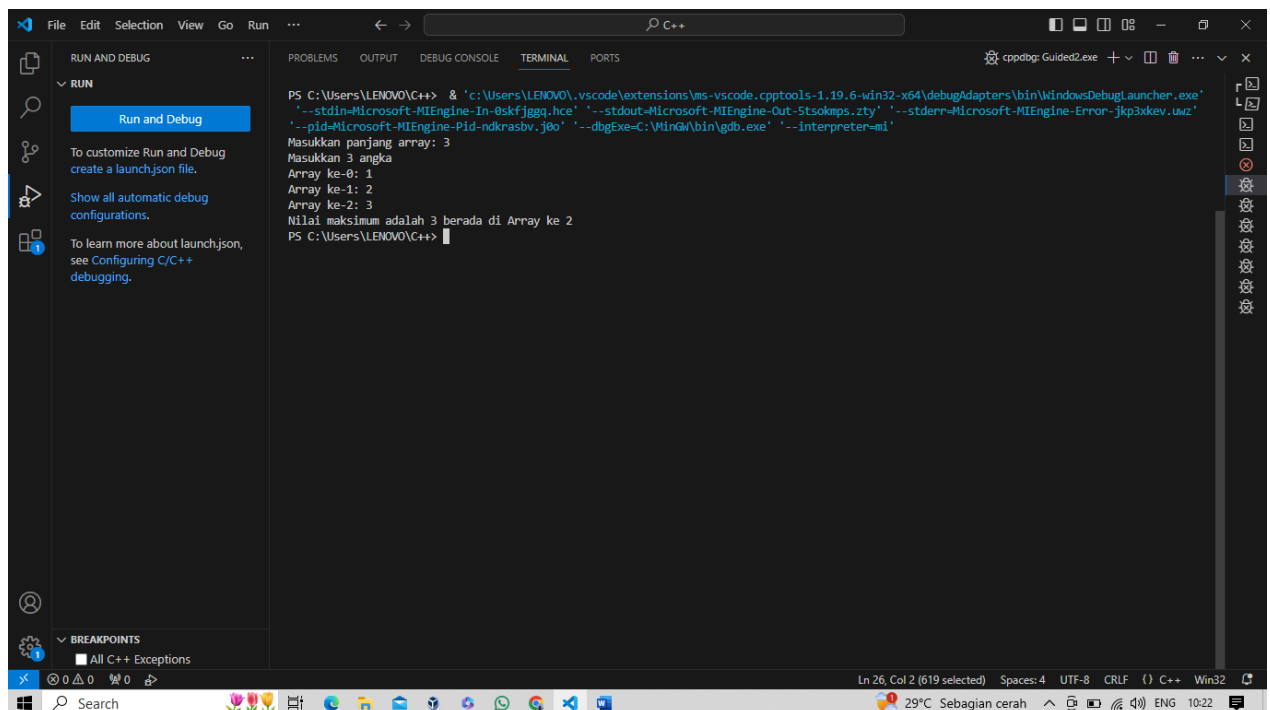
Guided 2

```
#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 :



Deskripsi :

Program di atas merupakan sebuah program C++ yang bertujuan untuk mencari nilai maksimum dalam sebuah array yang diinputkan oleh pengguna. Pertama, program meminta pengguna untuk memasukkan panjang array, kemudian meminta pengguna untuk memasukkan angka-angka tersebut satu per satu. Selanjutnya, program melakukan iterasi melalui array untuk menemukan nilai maksimum dan mencatat lokasi indeks di mana nilai maksimum tersebut ditemukan. Setelah selesai, program mencetak nilai maksimum beserta lokasi indeksnya ke layar.

C. Unguided

Unguided 1

Source Code

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

int main() {
    int bilangan_091 [10];

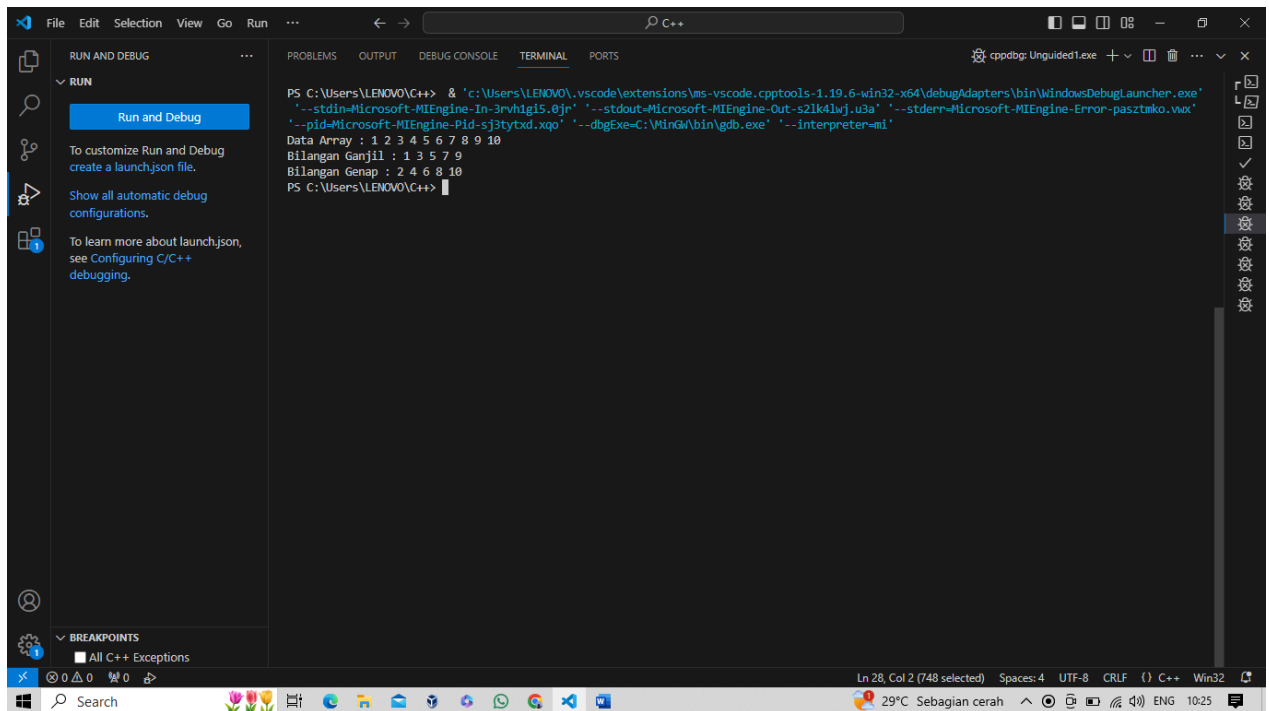
    //data array yang akan di proses
    cout << "Data Array : " ;
    for (int i=0; i<10; i++){
        cin >> bilangan_091 [i] ;
    }

    //proses pemilihan nomor ganjil yang akan di outputkan
    cout << "Bilangan Ganjil : " ;
    for (int i=0; i<10; i++){
        if (bilangan_091 [i]%2 != 0){
            cout << bilangan_091 [i] << " " ;
        }
    }

    //proses pemilihan nomor genap yang akan di outputkan
    cout << "\nBilangan Genap : " ;
    for (int i=0; i<10; i++){
        if (bilangan_091 [i]%2 == 0){
            cout << bilangan_091 [i] << " " ;
        }
    }

    return 0;
}
```

Screenshots Output :



Deskripsi :

Program di atas menggunakan C++ untuk mengambil input sepuluh bilangan dari pengguna dan kemudian memisahkan bilangan ganjil dan genap dari array. Pertama, program meminta pengguna untuk memasukkan sepuluh bilangan dan menyimpannya dalam array, kemudian melakukan dua iterasi berbeda melalui array. Program pertama kali menghitung bilangan ganjil dari array dan memeriksa apakah sisa pembagian bilangan dengan 2 tidak sama dengan 0. Kemudian, program menghitung bilangan genap dari array dan mengakhiri eksekusi.

Unguided 2 Source Code

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

int main() {
    int a_091, b, c;
    cout << "Masukkan jumlah elemen matriks: ";
    cin >> a_091;
    cout << "Masukkan ukuran matriks (y,z): ";
    cin >> b >> c;

    int arr[a_091][b][c];
    //Input elemen
    for (int i = 0; i < a_091; i++) {
        for (int j = 0; j < b; j++) {
```

```

        for (int k = 0; k < c; k++) {
            cout << "Input Array[" << i << "][" << j << "][" << k
<< "] = ";
            cin >> arr[i][j][k];
        }
    }
    cout << endl;
}
//Output Array
for (int i = 0; i < a_091; i++) {
    for (int j = 0; j < b; j++) {
        for (int k = 0; k < c; k++) {
            cout << "Data Array[" << i << "][" << j << "][" << k <<
"] = " << arr[i][j][k] << endl;
        }
    }
}
cout << endl;
//Tampilan array
for (int i = 0; i < a_091; i++) {
    cout << "Matriks ke-" << i+1 << ":" << endl;
    for (int j = 0; j < b; j++) {
        for (int k = 0; k < c; k++) {
            cout << arr[i][j][k] << " ";
        }
        cout << endl;
    }
    cout << endl;
}
return 0;
}

```

Screenshots Output :

```
File Edit Selection View Go Run ...  
C/C++: g++  
RUN...  
Masukkan jumlah elemen matriks: 3  
Masukkan ukuran matriks (y,z): 3 3  
Input Array[0][0][0] = 0  
Input Array[0][0][1] = 0  
Input Array[0][0][2] = 0  
Input Array[0][1][0] = 0  
Input Array[0][1][1] = 0  
Input Array[0][1][2] = 0  
Input Array[0][2][0] = 0  
Input Array[0][2][1] = 0  
Input Array[0][2][2] = 0  
Input Array[1][0][0] = 9  
Input Array[1][0][1] = 9  
Input Array[1][0][2] = 9  
Input Array[1][1][0] = 9  
Input Array[1][1][1] = 9  
Input Array[1][1][2] = 9  
Input Array[1][2][0] = 9  
Input Array[1][2][1] = 9  
Input Array[1][2][2] = 9  
Input Array[2][0][0] = 1  
Input Array[2][0][1] = 1  
Input Array[2][0][2] = 1  
Input Array[2][1][0] = 1  
Input Array[2][1][1] = 1  
Input Array[2][1][2] = 1  
Input Array[2][2][0] = 1  
Input Array[2][2][1] = 1  
Input Array[2][2][2] = 1  
Data Array[0][0][0] = 0  
Data Array[0][0][1] = 0  
Data Array[0][0][2] = 0  
Data Array[0][1][0] = 0  
Data Array[0][1][1] = 0  
Ln 44, Col 2 (1260 selected) Spaces: 4 UTF-8 CRLF C++ Win32
```

```
File Edit Selection View Go Run ...  
C/C++: g++  
RUN...  
Data Array[0][2][1] = 0  
Data Array[0][2][2] = 0  
Data Array[1][0][0] = 9  
Data Array[1][0][1] = 9  
Data Array[1][0][2] = 9  
Data Array[1][1][0] = 9  
Data Array[1][1][1] = 9  
Data Array[1][1][2] = 9  
Data Array[1][2][0] = 9  
Data Array[1][2][1] = 9  
Data Array[1][2][2] = 9  
Data Array[2][0][0] = 1  
Data Array[2][0][1] = 1  
Data Array[2][0][2] = 1  
Data Array[2][1][0] = 1  
Data Array[2][1][1] = 1  
Data Array[2][1][2] = 1  
Data Array[2][2][0] = 1  
Data Array[2][2][1] = 1  
Data Array[2][2][2] = 1  
Matriks ke-1:  
0 0 0  
0 0 0  
0 0 0  
Matriks ke-2:  
9 9 9  
9 9 9  
9 9 9  
Matriks ke-3:  
1 1 1  
1 1 1  
1 1 1  
Ln 44, Col 2 (1260 selected) Spaces: 4 UTF-8 CRLF C++ Win32
```

Deskripsi :

Program C++ di atas memungkinkan pengguna memasukkan elemen ke dalam matriks multidimensi dengan ukuran yang telah ditentukan. Pertama, program meminta pengguna untuk memasukkan jumlah elemen matriks (ukuran matriks dalam dimensi pertama), serta ukuran matriks dalam dimensi kedua dan ketiga (z). Selanjutnya, program menggunakan tiga perulangan bersarang untuk mengisi elemen matriks dengan ukuran yang telah ditentukan. Program mencetak elemen-elemen matriks dalam format yang terstruktur ke layar dan menampilkan indeks dan nilai setiap elemen setelah mengisi matriks. Kemudian, program mencetak tampilan keseluruhan matriks, menampilkan matriks ke layar dengan

baris dan spasi kosong di antara baris dan matriks. Matriks multi-dimensi dengan berbagai ukuran dapat dimasukkan dan ditampilkan melalui program ini sesuai dengan

Unguided 3

Source Code

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

int main() {
    int a_091;
    cout << "Masukkan jumlah elemen array yang diinginkan : ";
    cin >> a_091;

    //Input elemen array
    int arr[a_091];
    for (int i = 0; i < a_091; i++) {
        cout << "Input elemen array ke-" << i+1 << ": ";
        cin >> arr[i];
    }
    //Pilih menu array
    int choose;
    if (true) {
        cout << endl;
        cout << "Pilih menu yang diinginkan : " << endl;
        cout << "1. Cari nilai maksimum" << endl;
        cout << "2. Cari nilai minimum" << endl;
        cout << "3. Cari nilai rata-rata" << endl;
        cout << "Pilih Menu : ";
        cin >> choose;
        switch(choose) {
            //Proses dan output nilai maksimum
            case 1: {
                int maks = arr[0];
                for (int i = 1; i < a_091; i++) {
                    if (arr[i] > maks) {
                        maks = arr[i];
                    }
                }
                cout << "Nilai maksimum : " << maks << endl;
                break;
            }
            //Proses dan output nilai minimum
            case 2: {
                int min = arr[0];
                for (int i = 1; i < a_091; i++) {
                    if (arr[i] < min) {
                        min = arr[i];
                    }
                }
                cout << "Nilai minimum : " << min << endl;
                break;
            }
        }
    }
}
```

```

    }
}
cout << "Nilai minimum : " << min << endl;
break;
}
//Proses dan output rata-rata
case 3: {
    int sum = 0;
    for (int i = 0; i < a_091; i++) {
        sum += arr[i];
    }
    double ratarata = (double)sum / a_091;
    cout << "Nilai rata-rata : " << ratarata <<
endl;

    break;
}
//Output default
default: {
    cout << "Pilihan tidak valid" << endl;
    break;
}
}
}
return 0;
}

```

Screenshots Output :

```

Input elemen array ke-3: 1

Pilih menu yang diinginkan :
1. Cari nilai maksimum
2. Cari nilai minimum
3. Cari nilai rata-rata
Pilih Menu : 1
Nilai maksimum : 9
PS C:\Users\LENOVO\C++ > & 'c:\Users\LENOVO\.vscode\extensions\ms-vscode.cpptools-1.19.6-win32-x64\debugAdapters\bin\WindowsDebugLauncher.exe'
'--stdin=Microsoft-MIEngine-In-gdbmkrnt.egt' '--stdout=Microsoft-MIEngine-Out-1la0mjy1.kom' '--stderr=Microsoft-MIEngine-Error-rorbufyw.qbn'
'--pid=Microsoft-MIEngine-Pid-cbhupsg.22d' '--dbgExe=C:\WinGDB\bin\gdb.exe' '--interpreter=mi'
Masukkan jumlah elemen array yang diinginkan : 3
Input elemen array ke-1: 0
Input elemen array ke-2: 9
Input elemen array ke-3: 1

Pilih menu yang diinginkan :
1. Cari nilai maksimum
2. Cari nilai minimum
3. Cari nilai rata-rata
Pilih Menu : 2
Nilai minimum : 0
PS C:\Users\LENOVO\C++ > & 'c:\Users\LENOVO\.vscode\extensions\ms-vscode.cpptools-1.19.6-win32-x64\debugAdapters\bin\WindowsDebugLauncher.exe'
'--stdin=Microsoft-MIEngine-In-taojjgsc.15g' '--stdout=Microsoft-MIEngine-Out-fa0jlwhl.gy5' '--stderr=Microsoft-MIEngine-Error-wacehdga.31x'
'--pid=Microsoft-MIEngine-Pid-i4vgs4fp.cam' '--dbgExe=C:\WinGDB\bin\gdb.exe' '--interpreter=mi'
Masukkan jumlah elemen array yang diinginkan : 3
Input elemen array ke-1: 0
Input elemen array ke-2: 9
Input elemen array ke-3: 1

Pilih menu yang diinginkan :
1. Cari nilai maksimum
2. Cari nilai minimum
3. Cari nilai rata-rata
Pilih Menu : 3
Nilai rata-rata : 3.33333
PS C:\Users\LENOVO\C++ >

```

Deskripsi :

Program di atas adalah program C++ yang memungkinkan pengguna memasukkan elemen ke dalam array dan kemudian memilih opsi untuk mencari nilai maksimum, minimum, atau rata-rata elemen. Program memulai dengan meminta pengguna untuk memasukkan jumlah elemen array yang mereka inginkan. Setelah itu, mereka memasukkan elemen-elemen array tersebut. Setelah itu, program menampilkan menu pilihan di mana pengguna dapat memilih operasi apa yang mereka inginkan; mereka dapat menemukan nilai maksimum, minimum, atau rata-rata dari array yang telah dimasukkan. Kemudian, program menggunakan struktur switch-case untuk menangani setiap opsi yang dipilih pengguna. Jika pengguna memilih opsi 1, program akan menghitung nilai tertinggi dari array yang dimasukkan dan mencetaknya ke layar. Jika pengguna memilih opsi 2, program akan menghitung nilai terkecil dari array yang dimasukkan dan mencetaknya ke layar.

D. Kesimpulan

Dari rangkaian program-program di atas, dapat disimpulkan bahwa array merupakan sebuah struktur data yang digunakan untuk menyimpan kumpulan nilai data dengan tipe yang sama dalam satu variabel. Setiap elemen dalam array diakses menggunakan indeks yang dimulai dari 0. Array memungkinkan penyimpanan data secara terstruktur dan efisien, serta memfasilitasi operasi-operasi seperti mencari nilai maksimum, minimum, dan rata-rata dari elemen-elemennya. Terdapat berbagai jenis array, seperti array satu dimensi, array dua dimensi (matriks), dan array multidimensi, yang digunakan sesuai kebutuhan dalam menyimpan dan memanipulasi data. Melalui penggunaan array, pemrogram dapat dengan mudah mengelola dan mengakses sejumlah besar data dalam program mereka.

E. Referensi

Contoh Program Array C++ dan Penjelasannya [Studi Kasus] (no date).

<https://kodedasar.com/blog/array-cpp/>.

Wulandari, S. (2023) 'Contoh program Array C++ seperti apa?,' *dibimbing.com*, 1 August.

<https://dibimbing.id/blog/detail/contoh-program->

[array#:~:text=Dalam%20C%2B%2B%2C%20Array%20adalah,satu%20dimensi%20dan%20multi%20dimensi.](https://dibimbing.id/blog/detail/contoh-program-array#:~:text=Dalam%20C%2B%2B%2C%20Array%20adalah,satu%20dimensi%20dan%20multi%20dimensi.)

Fajar (2018) *Array - Belajar C++*. <https://www-belajarcpp->

[com.translate.goog/tutorial/cpp/array/? x tr sl=id& x tr tl=en& x tr hl=en& x tr pto](https://www-belajarcpp-com.translate.goog/tutorial/cpp/array/? x tr sl=id& x tr tl=en& x tr hl=en& x tr pto=sc)
[=sc](https://www-belajarcpp-com.translate.goog/tutorial/cpp/array/? x tr sl=id& x tr tl=en& x tr hl=en& x tr pto=sc).