

LAPORAN PRAKTIKUM

MODUL II ARRAY



Disusun oleh:
Muhammad Rifki Fadhilah
NIM: 2311102032

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. Memahami konsep Array
2. Mengetahui jenis dimensi Array dan cara penulisannya
3. Dapat mengimplementasikan Array pada kode program yang dibuat.

BAB II

DASAR TEORI

Array adalah struktur data yang digunakan untuk mengumpulkan sekumpulan elemen dalam satu tempat seperti kita menyimpan pakaian dalam lemari. Setiap elemen dalam array memiliki indeks yang mengidentifikasi posisi dalam array, indeks dimulai dari nol(0) dan panjang Array bisa ditentukan panjangnya seperti 10 atau 50. Array memudahkan pengolahan sejumlah besar data dengan efisien.

Penggolongan Array berdasarkan dimensi,

1. Array Dimensi 1

Array satu dimensi adalah array yang menyimpan elemen-elemen dalam satu baris atau satu deretan . Contoh array 1 dimensi:

```
#include <iostream>
using namespace std;
int main() {
    int angka[5]; // Deklarasi array angka dengan 5
    elemen
    angka[0] = 1;
    angka[1] = 2;
    angka[2] = 3;
    angka[3] = 4;
    angka[4] = 5;
    for (int i = 0; i < 5; i++) {
        cout << "angka[" << i << "] = " << angka[i] <<
endl;
    }
    return 0;
}
```

2. Array Dimensi 2

Array dua dimensi adalah array yang memiliki dua indeks untuk mengakses elemen-elemen dalam bentuk matriks atau tabel. Indeks pertama digunakan untuk mengakses baris, dan indeks kedua digunakan untuk mengakses kolom. Contoh array dimensi 2:

```
#include <iostream>
using namespace std;
int main() {
    int matriks[3][3] = {
        {1, 2, 3},
        {4, 5, 6},
        {7, 8, 9}
    };
    for (int i = 0; i < 3; i++) {
        for (int j = 0; j < 3; j++) {
            cout << "matriks[" << i << "][" << j << "] = "
                << matriks[i][j] << " ";
        }
        cout << endl;
    }
    return 0;
}
```

3. Array Dimensi 3

Array banyak dimensi adalah array dengan lebih dari 2 dimensi. Dapat digunakan untuk data yang lebih kompleks. Indeks pertama digunakan untuk lapisan, indeks kedua untuk baris dan indeks ketiga untuk kolom. Contoh array banyak dimensi:

```
#include <iostream>
using namespace std;
int main() {
```

```
int kubus[2][3][4] = {  
    {  
        {1, 2, 3, 4},  
        {5, 6, 7, 8},  
        {9, 10, 11, 12}  
    },  
    {  
        {13, 14, 15, 16},  
        {17, 18, 19, 20},  
        {21, 22, 23, 24}  
    }  
};  
  
for (int i = 0; i < 2; i++) {  
    for (int j = 0; j < 3; j++) {  
        for (int k = 0; k < 4; k++) {  
            cout << "kubus[" << i << "][" << j <<  
                "]" << k << "] = " << kubus[i][j][k] << " ";  
        }  
        cout << endl;  
    }  
    cout << endl;  
}  
  
return 0;  
}
```

BAB III

GUIDED

1. 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] << " ";  
        }  
        cout << endl;  
    }  
    cout << endl;  
}  
}
```

Screenshoot program

```
guide1.cpp

PS D:\Project VS Code\C++\Semester 2\Praktikum Struktur Data> cd 'd:\Project VS Code\C++\Semester 2\Praktikum Struktur Data\modul2\output'
PS D:\Project VS Code\C++\Semester 2\Praktikum Struktur Data\modul2\output> & .\'guide1.exe'
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] = 10
Input Array[1][0][1] = 11
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

1 2 3
4 5 6
7 8 9

10 11 12
13 14 15
16 17 18
```


Deskripsi program

Program diatas adalah program menginput nilai ke array 3 dimensi. Dimulai dari mendeklarasikan array 3 dimensi dengan ukuran 2(matriks) x3(baris)x3(kolom). Setelah itu menggunakan nested loop for untuk menginputkan nilai ke dalam array. Loop pertama untuk mengakses indeks array pertama. Loop kedua untuk indeks array kedua. Dan loop ketiga untuk indeks array ketiga. Lalu menggunakan nested loop for lagi untuk menampilkan nilai dari setiap elemen array yang diinputkan pengguna.

2. 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



```

guide2.cpp
PS D:\Project VS Code\C++\Semester 2\Praktikum Struktur Data> cd 'd:\Project VS Code\C++\Semester 2\Praktikum
Struktur Data\modul2\output'
PS D:\Project VS Code\C++\Semester 2\Praktikum Struktur Data\modul2\output> & .\guide2.exe'
Masukkan panjang array: 1
Masukkan 1 angka
Array ke-0: 1
Nilai maksimum adalah 1 berada di Array ke -2
PS D:\Project VS Code\C++\Semester 2\Praktikum Struktur Data\modul2\output> cd 'd:\Project VS
Code\C++\Semester 2\Praktikum Struktur Data\modul2\output'
PS D:\Project VS Code\C++\Semester 2\Praktikum Struktur Data\modul2\output> & .\guide2.exe'
Masukkan panjang array: 3
Masukkan 3 angka
Array ke-0: 1
Array ke-1: 2
Array ke-2: 6
Nilai maksimum adalah 6 berada di Array ke 2

```

Deskripsi program

Program diatas adalah program mencari nilai maksimum dalam sebuah array. Dimulai dari mendeklarasikan variabel maks untuk menyimpan nilai maksimum, a untuk menyimpan panjang array yang akan diinputkan, l untuk perulangan, lokasi untuk menyimpan indeks untuk nilai maksimum dalam array. Lalu pengguna diminta untuk memasukkan panjang array. Setelah itu array dengan panjang a dideklarasikan. Lalu pengguna memasukkan angka ke dalam array menggunakan perulangan for. Variabel maks diisi dengan nilai pertama dalam array. Lalu perulangan for untuk mencari nilai maksimum dalam array. Jika nilai pada indeks ke l lebih

besar dari nilai maksimum saat ini. Maka nilai maksimum diperbarui dan indeksinya disimpan dalam variabel lokasi.

LATIHAN KELAS - UNGUIDED

1. Unguided 1

Source code

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

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

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

    cout << "Data array: ";
    for (int i = 0; i < a; i++) {
        cout << array[i];
        if (i < a - 1) {
            cout << ",";
        }
    }
    cout << endl;

    cout << "Nilai ganjil: ";
    for (int i = 0; i < a; i++) {
        if (array[i] % 2 != 0) {
            cout << array[i];
            if (i < a - 2) {
```

```
        cout << ",";

    }

}

cout << endl;

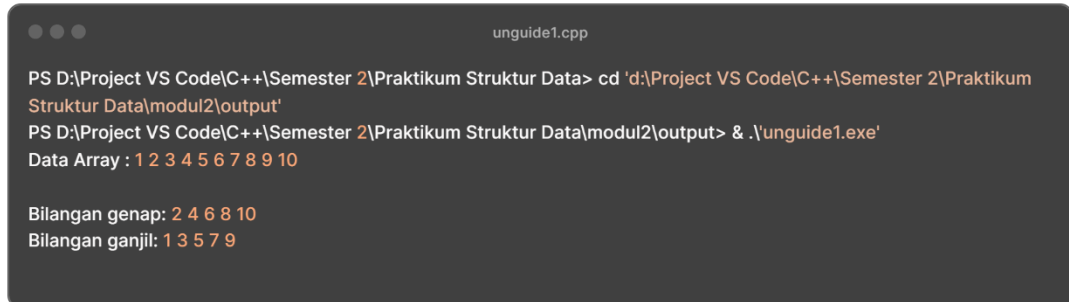
cout << "Nilai genap: ";
for (int i = 0; i < a; i++) {
    if (array[i] % 2 == 0) {
        cout << array[i];
        if (i < a - 2) {
            cout << ",";
        }
    }
}

cout << endl;

return 0;

}
```

Screenshoot program



```
unguide1.cpp
PS D:\Project VS Code\C++\Semester 2\Praktikum Struktur Data> cd 'd:\Project VS Code\C++\Semester 2\Praktikum
Struktur Data\modul2\output'
PS D:\Project VS Code\C++\Semester 2\Praktikum Struktur Data\modul2\output> & .\unguide1.exe
Data Array : 1 2 3 4 5 6 7 8 9 10

Bilangan genap: 2 4 6 8 10
Bilangan ganjil: 1 3 5 7 9
```

Deskripsi program

Program diatas adalah program yang mengambil inputan 10 bilangan dan memisahkan bilangan-bilangan tersebut menjadi bilangan genap dan ganjil. Pertama, deklarasikan `arr[100]` untuk menyimpan data yang dimasukkan oleh pengguna, `ganjil[100]` untuk menyimpan bilangan ganjil dan `genap[100]` untuk menyimpan bilangan genap, `j` untuk penghitung bilangan genap, `k` untuk penghitung bilangan ganjil, `n` untuk menunjukkan jumlah data yang akan dimasukkan. Program meminta pengguna memasukkan 10 bilangan dan nanti 10 bilangan tersebut dimasukkan ke dalam array `arr`. Loop `for` digunakan untuk mengulangi proses input sebanyak `n` kali. Setelah memasukkan data, program melakukan iterasi melalui array `arr` dan memeriksa setiap elemennya. Jika sebuah elemen habis dibagi 2 maka termasuk bilangan genap jika tidak habis dibagi 2 maka termasuk bilangan ganjil. Setelah memisahkan bilangan ganjil dan genap maka program menampilkan bilangan ganjil dan genap tersebut menggunakan loop `for`.

2. Unguided 2

Source code

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

int main()
{
    int x, y, z;
    cout << "Masukkan jumlah elemen ke-1: ";
    cin >> x;
    cout << "Masukkan jumlah elemen ke-2: ";
    cin >> y;
    cout << "Masukkan jumlah elemen ke-3: ";
    cin >> z;

    int arr[x][y][z];

    // Input Array
    for (int i = 0; i < x; i++)
    {
        for (int j = 0; j < y; j++)
        {
            for (int k = 0; k < z; k++)
            {
                cout << "Input Array[" << i << "][" << j << "]["
<< k << "] = ";
                cin >> arr[i][j][k];
            }
        }
        cout << endl;
    }

    // Output Array
```

```
for (int i = 0; i < x; i++)
{
    for (int j = 0; j < y; j++)
    {
        for (int k = 0; k < z; k++)
        {
            cout << "Data Array[" << i << "][" << j << "]["
<< k << "] = " << arr[i][j][k] << endl;
        }
    }

    cout << endl;

    // Tampilan array
    for (int i = 0; i < x; i++)
    {
        for (int j = 0; j < y; j++)
        {
            for (int k = 0; k < z; k++)
            {
                cout << arr[i][j][k] << " ";
            }
            cout << endl;
        }
        cout << endl;
    }

    return 0;
}
```


Screenshoot program

```
unguided2.cpp

PS D:\Project VS Code\C++\Semester 2\Praktikum Struktur Data> cd 'd:\Project VS Code\C++\Semester 2\Praktikum
Struktur Data\modul2\output'
PS D:\Project VS Code\C++\Semester 2\Praktikum Struktur Data\modul2\output> & .\unguided2.exe'
Masukkan jumlah elemen ke-1: 2
Masukkan jumlah elemen ke-2: 4
Masukkan jumlah elemen ke-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[0][3][0] = 10
Input Array[0][3][1] = 11
Input Array[0][3][2] = 12

Input Array[1][0][0] = 13
Input Array[1][0][1] = 14
Input Array[1][0][2] = 15
Input Array[1][1][0] = 16
Input Array[1][1][1] = 17
Input Array[1][1][2] = 18
Input Array[1][2][0] = 19
Input Array[1][2][1] = 20
Input Array[1][2][2] = 21
Input Array[1][3][0] = 22
Input Array[1][3][1] = 23
Input Array[1][3][2] = 24

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[0][3][0] = 10
Data Array[0][3][1] = 11
Data Array[0][3][2] = 12
Data Array[1][0][0] = 13
Data Array[1][0][1] = 14
Data Array[1][0][2] = 15
Data Array[1][1][0] = 16
Data Array[1][1][1] = 17
Data Array[1][1][2] = 18
Data Array[1][2][0] = 19
Data Array[1][2][1] = 20
Data Array[1][2][2] = 21
Data Array[1][3][0] = 22
Data Array[1][3][1] = 23
Data Array[1][3][2] = 24

1 2 3
4 5 6
7 8 9
10 11 12

13 14 15
16 17 18
19 20 21
22 23 24
```

Deskripsi program

Program diatas adalah program menginputkan jumlah elemen dan nilai ke array 3 dimensi. Pertama, mendeklarasiakn 3 variabel x,y,z untuk menentukan ukuran array 3 dimensi. Array 3 dimensi arr dengan ukuran x,y,z dideklarasikan. Pengguna memasukkan nilai untuk setiap elemen array menggunakan loop for yang berjalan sebanyak x,y,z kali. Di mana setiap iterasi akan meminta input untuk 1 elemen array. Setelah selesai input, program akan mengoutputkan tampilan array secara keseluruhan.

3. Unguided 3

Source code

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

int main() {
    int menu;
    int arr[10];
    bool inputDone = false;

    do {
        // Tampilkan menu
        cout << "MENU" << endl;
        cout << "1. Input Array" << endl;
        cout << "2. Tampil Array" << endl;
        cout << "3. Cari Nilai Minimum" << endl;
        cout << "4. Cari Nilai Maksimum" << endl;
        cout << "5. Hitung Rata-rata" << endl;
        cout << "6. Keluar" << endl;
        cout << "Masukkan Pilihan: ";
        cin >> menu;
```

```

switch (menu) {
    case 1:
        cout << "Masukkan 10 bilangan bulat:" << endl;
        for (int i = 0; i < 10; i++) {
            cout << "Bilangan ke-" << i + 1 << ": ";
            cin >> arr[i];
        }
        inputDone = true;
        break;

    case 2:
        if (inputDone) {
            cout << "Isi Array: ";
            for (int i = 0; i < 10; i++) {
                cout << arr[i] << " ";
            }
            cout << endl;
        } else {
            cout << "Anda belum menginputkan array." <<
endl;

        }
        break;

    case 3:
        if (inputDone) {
            int minVal = arr[0];
            for (int i = 1; i < 10; i++) {
                if (arr[i] < minVal) {
                    minVal = arr[i];
                }
            }
            cout << "Nilai Minimum: " << minVal << endl;
        } else {

```

```

        cout << "Anda belum menginputkan array." <<
endl;

        }
        break;

    case 4:
        if (inputDone) {
            int maxVal = arr[0];
            for (int i = 1; i < 10; i++) {
                if (arr[i] > maxVal) {
                    maxVal = arr[i];
                }
            }
            cout << "Nilai Maksimum: " << maxVal << endl;
        } else {
            cout << "Anda belum menginputkan array." <<
endl;
        }
        break;

    case 5:
        if (inputDone) {
            int total = 0;
            for (int i = 0; i < 10; i++) {
                total += arr[i];
            }
            double rataRata = static_cast<double>(total)
/ 10;

            cout << fixed << setprecision(2);
            cout << "Rata-rata: " << rataRata << endl;
        } else {
            cout << "Anda belum menginputkan array." <<
endl;
        }

```

```
        break;

        case 6:
            cout << "Terima kasih!" << endl;
            break;

        default:
            cout << "Pilihan tidak valid. Silakan coba lagi."
<< endl;
    }
} while (menu != 6);

return 0;
}
```

Screenshoot program

```
unguide3.cpp

PS D:\Project VS Code\C++\Semester 2\Praktikum Struktur Data\modul2\output> cd 'd:\Project VS
Code\C++\Semester 2\Praktikum Struktur Data\modul2\output'
PS D:\Project VS Code\C++\Semester 2\Praktikum Struktur Data\modul2\output> & .\unguide3.exe'
MENU
1. Input Array
2. Tampil Array
3. Cari Nilai Minimum
4. Cari Nilai Maksimum
5. Hitung Rata-rata
6. Keluar
Masukkan Pilihan: 1
Masukkan 10 bilangan bulat:
Bilangan ke-1: 1
Bilangan ke-2: 2
Bilangan ke-3: 3
Bilangan ke-4: 4
Bilangan ke-5: 5
Bilangan ke-6: 6
Bilangan ke-7: 7
Bilangan ke-8: 8
Bilangan ke-9: 9
Bilangan ke-10: 10
MENU
1. Input Array
2. Tampil Array
3. Cari Nilai Minimum
4. Cari Nilai Maksimum
5. Hitung Rata-rata
6. Keluar
Masukkan Pilihan: 2
Isi Array: 1 2 3 4 5 6 7 8 9 10
MENU
1. Input Array
2. Tampil Array
3. Cari Nilai Minimum
4. Cari Nilai Maksimum
5. Hitung Rata-rata
6. Keluar
Masukkan Pilihan: 3
Nilai Minimum: 1
MENU
1. Input Array
2. Tampil Array
3. Cari Nilai Minimum
4. Cari Nilai Maksimum
5. Hitung Rata-rata
6. Keluar
Masukkan Pilihan: 4
Nilai Maksimum: 10
MENU
1. Input Array
2. Tampil Array
3. Cari Nilai Minimum
4. Cari Nilai Maksimum
5. Hitung Rata-rata
6. Keluar
Masukkan Pilihan: 5
Rata-rata: 5.50
MENU
1. Input Array
2. Tampil Array
3. Cari Nilai Minimum
4. Cari Nilai Maksimum
5. Hitung Rata-rata
6. Keluar
Masukkan Pilihan: 6
Terima kasih!
```

Deskripsi program

Program ini membuat menu untuk input, menampilkan array, mencari nilai minimum dan maksimum, rata rata, dan opsi keluar. Penjelasan alur program dibawah ini:

1. menu: Ini adalah variabel yang digunakan untuk menyimpan pilihan menu yang dimasukkan oleh pengguna.
2. arr[10]: Ini adalah array yang digunakan untuk menyimpan 10 bilangan bulat yang akan dimasukkan oleh pengguna.
3. inputDone: Variabel boolean yang digunakan untuk melacak apakah pengguna sudah menginputkan array atau belum.
4. Program memulai sebuah loop do-while, yang akan berjalan sampai pengguna memilih opsi "Keluar" (menu 6).
5. Program menampilkan menu yang berisi enam opsi kepada pengguna, dan pengguna diminta untuk memasukkan pilihan mereka dengan menggunakan variabel menu.
6. Setelah pengguna memilih menu tertentu, program menggunakan struktur switch-case untuk mengeksekusi tindakan yang sesuai dengan pilihan pengguna.
 - a. Jika pengguna memilih menu 1, program meminta pengguna untuk memasukkan 10 bilangan bulat ke dalam array arr.
 - b. Jika pengguna memilih menu 2, program akan menampilkan isi array arr jika pengguna sudah menginputkan datanya; jika belum, program akan memberikan pesan kesalahan.
 - c. Jika pengguna memilih menu 3, program akan mencari nilai minimum dalam array arr dan menampilkannya.
 - d. Jika pengguna memilih menu 4, program akan mencari nilai maksimum dalam array arr dan menampilkannya.
 - e. Jika pengguna memilih menu 5, program akan menghitung rata-rata dari nilai-nilai dalam array arr dan menampilkannya.
 - f. Jika pengguna memilih menu 6, program akan menampilkan pesan "Terima kasih!" dan keluar dari loop.

7. Program menggunakan variabel `inputDone` untuk memastikan bahwa pengguna telah menginputkan data array sebelum menjalankan menu tertentu. Jika belum, program akan memberikan pesan kesalahan.
8. Program mengulangi loop `do-while` sampai pengguna memilih untuk keluar (menu 6). Ini memungkinkan pengguna untuk melakukan beberapa operasi terkait array dalam satu sesi program.

BAB IV

KESIMPULAN

Array adalah struktur data penting dalam pemrograman yang memungkinkan penyimpanan dan manipulasi data dalam satu variabel. Array satu dimensi merupakan koleksi linear dari elemen-elemen yang disusun dalam satu baris. Elemen-elemen ini diakses menggunakan satu indeks dan cocok digunakan untuk menyimpan data sederhana seperti daftar nama, nilai-nilai ujian, atau bilangan bulat.

Sementara itu, array dua dimensi adalah kumpulan elemen yang disusun dalam bentuk baris dan kolom. Elemen-elemen diakses menggunakan dua indeks, satu untuk baris dan satu lagi untuk kolom. Cocok digunakan untuk menyimpan data yang memiliki representasi dalam bentuk tabel atau matriks, seperti data dari spreadsheet, gambar, atau matriks matematika. Dengan adanya array multidimensi, seperti array tiga dimensi, pengguna dapat menyimpan data yang memiliki struktur yang lebih kompleks dan lebih dari dua level, seperti data citra 3D, data waktu ruang, atau matriks dalam konteks ilmu pengetahuan dan rekayasa. Dengan pemahaman yang baik tentang karakteristik dan penggunaan dari setiap jenis array ini, pengembang dapat memilih jenis array yang tepat sesuai dengan kebutuhan program yang sedang dikembangkan.