

# Array

## Tujuan Praktikum

1. Memahami struktur Array dalam program,
2. Mampu menerapkan Array untuk menyelesaikan berbagai kasus.

## Dasar Teori

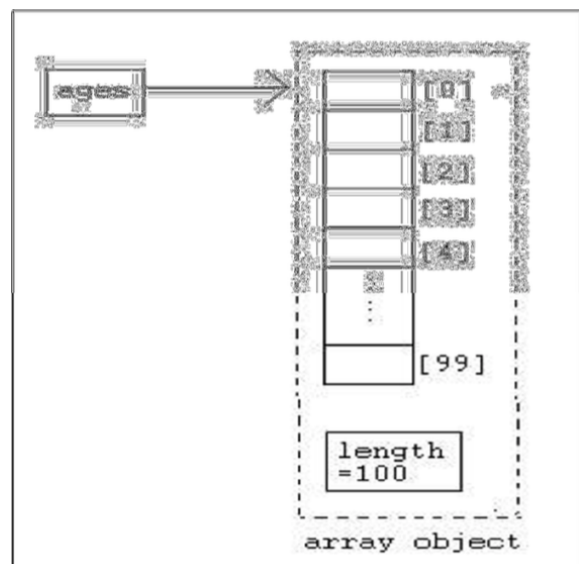
Sebuah larik/array akan menyimpan beberapa item data dengan tipe data yang sama di dalam sebuah blok memori yang berdekatan yang kemudian dibagi menjadi beberapa slot. Cara penyimpanan [struktur data] inilah yang disebut sebagai array.

Tipe data yang sama, disimpan dalam satu tempat yang sama dan diberi nomor indeks.

Untuk mendeklarasikan array, tulis tipe datanya diikuti dengan tanda kurung `[]`.

Contoh:

```
int ages[100];
```



## Praktik (Guided)

1. Mengisi nilai elemen-elemen array dengan nol dan mencetaknya.

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

int main()
{
    int n[10]; // n merupakan array bertipe data integer yang berukuran 10

    // Mengisi nilai nol pada array n dengan perulangan
    for (int i = 0; i < 10; i++)
    {
        n[i] = 0; // mengisi nilai elemen array pada index ke-i
    }

    cout << "Element" << setw(13) << "Value" << endl;

    // Mencetak nilai dari setiap elemen-elemen array
    for (int i = 0; i < 10; i++)
    {
        cout << setw(7) << i << setw(13) << n[i] << endl;
    }

    return 0;
} // akhir fungsi main
```

2. Mengisi nilai array pada saat deklarasi

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

int main()
{
    // Menggunakan initializer list untuk mengisi nilai pada array n
    int n[10] = {32, 27, 64, 18, 95, 14, 90, 70, 60, 37};

    cout << "Element" << setw(13) << "Value" << endl;

    // Mencetak nilai pada array
    for (int i = 0; i < 10; i++)
    {
        cout << setw(7) << i << setw(13) << n[i] << endl;
    }

    return 0;
} // akhir fungsi main
```

### 3. Mengisi array dengan bilangan genap dari 2 - 20

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

int main()
{
    // constant merupakan nilai yang tidak bisa diubah
    const int arraySize = 10;

    int s[arraySize]; // array s memiliki 10 elemen

    for (int i = 0; i < arraySize; i++) // mengatur nilai dari array s
    {
        s[i] = 2 + 2 * i;
    }

    cout << "Element" << setw(13) << "Value" << endl;

    // mencetak nilai pada array
    for (int i = 0; i < arraySize; i++)
    {
        cout << setw(7) << i << setw(13) << s[i] << endl;
    }

    return 0;
} // akhir fungsi main
```

### 4. Mengisi nilai variabel konstanta dengan cara yang benar (langsung diisi)

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

int main()
{
    const int x = 7; // Mengisi nilai dari variabel konstanta

    cout << "The value of constant variable x is: " << x << endl;

    return 0;
} // akhir fungsi main
```

### 5. Mengisi nilai variabel konstanta dengan cara yang salah (tidak langsung diisi)

```
#include <iostream>

int main()
{
    const int x; // Error: x harus diisi nilainya

    x = 7; // Error: x tidak boleh dimodifikasi

    return 0;
} // akhir fungsi main
```

## 6. Menjumlahkan elemen-elemen array

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

int main() {
    const int arraySize = 10; // Ukuran array dengan variabel konstanta
    int a[arraySize] = {87, 68, 94, 100, 83, 78, 85, 91, 76, 87};
    int total = 0;

    // Menjumlahkan isi dari array a
    for (int i = 0; i < arraySize; i++)
    {
        total += a[i];
    }

    cout << "Total of array elements: " << total << endl;

    return 0;
} // akhir fungsi main
```

## 7. Mengisi array dengan input dari user

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

int main()
{
    // Ukuran array menggunakan variabel
    int arraySize;

    cout << "Input ukuran array: ";
    cin >> arraySize;

    int s[arraySize]; // array s memiliki elemen sebanyak nilai dari arraySize
```

```

// melakukan perulangan
// untuk mengisi nilai array s berdasarkan input user
for (int i = 0; i < arraySize; i++)
{
    cout << "Input elemen array ke-" << i << ": ";
    cin >> s[i];
}

cout << "Element" << setw(13) << "Value" << endl;

// mencetak nilai pada array
for (int i = 0; i < arraySize; i++)
{
    cout << setw(7) << i << setw(13) << s[i] << endl;
}

return 0;
} // akhir fungsi main

```

## Latihan (Unguided)

1. Buatlah variabel array untuk menampung kode, nama barang, jumlah, harga dan total yang sudah ditentukan sebagai berikut! Total mula-mula adalah 0, dan didapatkan dari jumlah \* harga.

Output:

```

Masukkan Kode ke-1: 001
Masukkan Nama ke-1: Penghapus
Masukkan Jumlah ke-1: 4
Masukkan Harga ke-1: 1000

Masukkan Kode ke-2: 002
Masukkan Nama ke-2: Pensil
Masukkan Jumlah ke-2: 3
Masukkan Harga ke-2: 1500

Masukkan Kode ke-3: 003
Masukkan Nama ke-3: Buku
Masukkan Jumlah ke-3: 2
Masukkan Harga ke-3: 2000

Masukkan Kode ke-4: 004
Masukkan Nama ke-4: Rautan
Masukkan Jumlah ke-4: 3
Masukkan Harga ke-4: 1000

Masukkan Kode ke-5: 005
Masukkan Nama ke-5: Penggaris

```

```
Masukkan Jumlah ke-5: 5
Masukkan Harga ke-5: 500
```

Kode	Nama	Jumlah	Harga	Total
001	Penghapus	4	1000	4000
002	Pensil	3	1500	4500
003	Buku	2	2000	4000
004	Rautan	3	1000	3000
005	Penggaris	5	500	2500

Jumlah item = 17  
Total pembelian = 18000

2. Buatlah program untuk menghitung banyak data, rata-rata, jumlah dari sekumpulan data yang dimasukkan!

Output:

```
Masukan banyaknya data = 5
Data[1] = 5
Data[2] = 2
Data[3] = 4
Data[4] = 4
Data[5] = 5
Banyaknya data = 5
Rata-rata = 4
Jumlah = 20
```

3. Tambahkan standar deviasi pada soal no 2!

Rumus standar deviasi (sd):

$$\begin{aligned} \sigma &= \sigma + \text{pow}(\text{data}[i] - \text{rataRata}, 2) \\ \text{sd} &= \text{sqrt}(\sigma/n) \end{aligned}$$

Catatan:

- Dalam C++, fungsi `pow()` digunakan untuk mengangkat suatu bilangan. Maksud dari `pow(data[i] - rataRata, 2)` di atas adalah  $(\text{data}[i] - \text{rataRata})^2$ .
- Jangan lupa untuk meng-include file header `math.h`.

```
#include <math.h>
```

Output:

```
Masukan banyaknya data = 5  
Data[1] = 5  
Data[2] = 2  
Data[3] = 4  
Data[4] = 4  
Data[5] = 5  
Standar Deviasi = 1.09545
```