

02 Literal Data

Ringkasan Materi

Fungsi mendasar dari sebuah program adalah mengolah data. Pengolahan ini akan menghasilkan perubahan nilai atau bentuk yang berbeda dari data. Sebagai contoh sebuah program kalkulator melakukan operasi penjumlahan dengan input 2 bilangan maka output yang dihasilkan adalah berupa bilangan dengan nilai yang berbeda. Contoh lain adalah sebuah program untuk memutar lagu. Input program tersebut berupa sebuah dokumen dengan data numerik dan outputnya adalah bunyi-bunyian.

Ada berbagai macam operasi yang digunakan oleh program untuk melakukan pengolahan data. Operasi-operasi tersebut menyesuaikan bentuk dari data yang diolah. Sebagai contoh bila data berupa numerik maka dapat menggunakan operasi aritmatika seperti penjumlahan, pengurangan, perkalian, dan pembagian. Demikian halnya dengan bentuk-bentuk data yang lain maka operasinya juga menyesuaikan.

Pembahasan materi ini akan meliputi beberapa hal tentang data, di antaranya:

1. Tipe Data
2. Variabel dan konstanta
3. Operator

Pembahasan tentang data selengkapnya dapat dilihat pada slide [02 Literal Data.pptx](#).

Percobaan

LiteralTest

Tujuan dari project ini adalah untuk melihat beberapa penggunaan literal data dalam program. Dalam project juga akan melibatkan penggunaan variabel serta konstanta untuk menampung data.

Buatlah sebuah project baru dengan nama **LiteralTes** dan nama **Solution: 02**. Tambahkan file **main.cpp** pada folder **Source Files** di Solution Explorer. Tuliskan kode berikut ini.

```
#include <iostream>
#include <string>

using namespace std;

int main()
{
    // Integer literals
    int dec = 42;           // Decimal literal
    int oct = 052;         // Octal literal (base 8)
    int hex = 0x2A;        // Hexadecimal literal (base 16)
    int bin = 0b101010;    // Binary literal (C++14 and later)

    // Floating-point literals
    double pi = 3.14159;   // Double precision floating-point literal
    float e = 2.71828f;    // Single precision floating-point literal
    long double golden = 1.6180339887L; // Long double precision floating-point literal

    // Character literals
    char letter = 'A';     // Character literal
    char newline = '\n';   // Escape sequence for newline

    // String literals
```

```

string greeting = "Hello, World!"; // String literal
// Boolean literals
bool isTrue = true;           // Boolean literal true
bool isFalse = false;        // Boolean literal false

// Output the values
cout << "Int: " << dec << ", " << oct << ", " << hex << ", " << bin << endl;
cout << "Float: " << pi << ", " << e << ", " << golden << endl;
cout << "Char: " << letter << endl;
cout << "String: " << greeting << endl;
cout << "Boolean: " << isTrue << ", " << isFalse << endl;

system("PAUSE");

return 0;
}

```

Jalankan kode di atas dan amati outputnya. Kemudian lakukan beberapa modifikasi:

- Ubahlah nilai variabel **dec** menjadi **5.8**
pi menjadi **3**
Jalankan kembali dan amati apa yang terjadi.
- Ubahlah variabel **greeting** menjadi konstanta dengan menambah keyword **const** di depan **string**. Kemudian di baris sebelum **system("PAUSE")** tambahkan kode:

```

greeting = "Good morning";
cout << "String: " << greeting << endl;

```

Apa yang terjadi?

ArithmeticTest

Tujuan dari project ini adalah untuk melakukan percobaan terhadap beberapa operasi aritmatik. Tipe-tipe data apa saja yang dapat dan tidak dapat digunakan pada operasi aritmatik tertentu.

Buatlah project baru dengan nama **ArithmeticTest** dengan cara yang sama seperti sebelumnya kemudian tuliskan kode di bawah ini.

```

#include <iostream>

using namespace std;

int main()
{
    int a = 10;
    int b = 3;

    cout << "Addition: " << (a + b) << endl;           // Addition
    cout << "Subtraction: " << (a - b) << endl;        // Subtraction
    cout << "Multiplication: " << (a * b) << endl;      // Multiplication
    cout << "Division: " << (a / b) << endl;           // Division
    cout << "Modulus: " << (a % b) << endl;           // Modulus
    cout << "Increment: " << (++a) << endl;            // Increment
    cout << "Decrement: " << (--b) << endl;           // Decrement
    cout << "Compound Addition: " << (a += 5) << endl; // Compound Addition
    cout << "Compound Multiplication: " << (b *= 2) << endl; // Compound Multiplication

    system("PAUSE");

    return 0;
}

```

Jalankan kode di atas kemudian lakukan beberapa modifikasi:

1. Ubah tipe data pada variabel **a** dan **b** menjadi **double**
2. Ubah tipe data pada variabel **a** dan **b** menjadi **char**
3. Sebelum baris **system("PAUSE")** tambahkan kode:

```
double c = 5 / 2;
cout << "nilai c: " << c << endl;
```

Amati nilai **c**, mengapa demikian?

Tentukan tipe data untuk variabel **c** agar nilainya benar.

RelationalTest

Tujuan dari project ini adalah untuk melakukan percobaan operasi relasional dan logika dikombinasikan dengan aritmatik.

Buatlah project baru dengan nama **RelationalTest** kemudian tuliskan kode seperti di bawah ini.

```
#include <iostream>
using namespace std;
int main()
{
    int a = 10;
    int b = 9;

    cout << "Equal to: " << (a == b) << endl;
    cout << "Not equal to: " << (a != b) << endl;
    cout << "Greater than: " << (a > b) << endl;
    cout << "Less than: " << (a < b) << endl;
    cout << "Greater than or equal to: " << (a >= b) << endl;
    cout << "Less than or equal to: " << (a <= b) << endl;

    cout << "Logical AND: " << ((a < 5) && (b++ == 10)) << endl;
    cout << "a: " << a << ", b: " << b << endl;
    cout << "Logical OR: " << ((a > 5) || (b == 10)) << endl;
    cout << "a: " << a << ", b: " << b << endl;

    system("PAUSE");

    return 0;
```

Jalankan kode tersebut, amati setiap perubahan pada variabel **a** dan **b**. Ubahlah operator **AND** dan **OR** dari **&&** dan **||** menjadi **&** dan **|**. Amati lagi apa yang terjadi pada variabel **a** dan **b**.

Latihan

Kerjakan dalam sebuah project dengan nama **Latihan**.

1. Buatlah program untuk menghitung luas lingkaran dengan jari-jari 14.
2. Buatlah program untuk menghitung rata-rata dari barisan bilangan: 4, 1, 9, 6, 2. Tentukan tipe
3. Diberikan sebuah variabel dengan nama **second** bernilai **3781**. Variabel tersebut menunjukkan nilai detik. Lakukan operasi untuk mengkonversi nilai variabel **second** menjadi **jam**, **menit**, dan **detik**.