

MODUL 1: TIPE DATA DASAR, OPERATOR, DAN POINTER

LAB 1.1: TUJUAN PRAKTIKUM

Learning outcome untuk praktikum pemrograman dasar pada modul 1 ini adalah:

LO 1 Mahasiswa mampu mengimplementasikan dan menganalisis kegunaan berbagai macam tipe data dan operator dalam C++.

LAB 1.2: DASAR TEORI

Unsur--unsur sebuah program komputer yang akan dibahas pada modul ini meliputi, unsur--unsur aritmatika, penggunaan tipe data dan operator, penulisan sintaks, dan proses compiling source code dengan compiler GCC menggunakan bahasa C++.

Struktur program dalam bahasa C++

Dalam struktur penulisannya, program dalam bahasa C++ terdiri dari kumpulan fungsi--fungsi, dan paling sedikit harus ada sebuah fungsi wajib yaitu fungsi main. Bahasa C++ merupakan bahasa pemrograman yang mendukung konsep pemrograman berorientasi objek, namun pada pertemuan ini kita akan menggunakan paradigma struktural,

TIPE-TIPE DATA DAN OPERATOR DALAM BAHASA C++

Tipe data dalam bahasa C++ tidak jauh berbeda dengan tipe data dalam bahasa pemrograman lainnya, seperti integer, floating point, character, dan seterusnya. Secara umum, tipe data dalam Bahasa C++ dapat digolongkan ke dalam 3 jenis, yaitu tipe data primitif/dasar, tipe data bentukan, dan tipe data abstrak atau yang ditentukan oleh programmer. Pada modul ini, kita akan belajar mengimplementasikan dan menganalisis kegunaan berbagai macam tipe data dasar pada Bahasa C++.

Secara umum operator semua bahasa pemrograman adalah sama, yaitu menurut operand yang dibutuhkan terdiri dari 2 macam operator, binary dan unary. Yang dimaksud dengan operator binary adalah operator yang memerlukan 2 operand untuk berfungsi dengan baik (tidak ada syntax error) sebagai contoh, operator aritmatika (perkalian, penambahan, pengurangan, pembagian), operator logika sama dengan (==), tidak sama dengan (!=), lebih besar (>), lebih kecil (<), dan lain-lain. Sedangkan operator unary adalah operator yang hanya memerlukan sebuah operand saja untuk melengkapinya menjadi syntax yang benar. Sebagai contoh, operator increment (++ / --), operator NOT (!), operator tanda bilangan (+ / -), dan lain sebagainya.

Operator--operator tersebut mempunyai prioritas yang berbeda--beda. Untuk operator aritmatika urutan prioritas tersebut adalah persis sama dengan urutan pengerjaan suatu operasi hitung matematika. Operasi perkalian dan pembagian akan dikerjakan lebih dulu dari pada operasi penambahan dan pengurangan. Berarti, operator perkalian dan pembagian mempunyai urutan prioritas yang lebih tinggi dari pada operator penjumlahan dan pengurangan.

POINTER

Pointer adalah suatu variabel khusus yang berisikan alamat memori dari suatu data, tidak berisikan data itu sendiri atau dapat juga dipahami bahwa pointer adalah variabel yang berisikan alamat dari suatu data. Pointer sangat sering digunakan dalam program-program bahasa C++, karena dapat memberikan efisiensi dalam program. Seperti variabel lainnya dalam C++, pointer juga harus dideklarasikan dahulu sebelum dapat digunakan dalam program. Untuk mendeklarasikan sebuah pointer, sintaksnya adalah sebagai berikut:

[tipe data] *[nama pointer]

Jadi untuk mendeklarasikan sebuah pointer bernama xp yang menunjuk ke alamat data integer, digunakan perintah seperti berikut:

Int *xp

HINT

Pointer adalah salah satu keunggulan bahasa pemrograman C++ dibanding dengan bahasa lainnya.

Terdapat dua buah operator unary yang digunakan untuk mengakses sebuah pointer, yaitu operator & dan operator *. Operator & digunakan untuk mengakses alamat data yang ditunjuk oleh pointer, sementara operator * digunakan untuk mengakses data yang alamatnya ditunjuk oleh pointer.

```
a = &x;      //a berisikan alamat yang ditunjuk oleh pointer x
b = *x;      //b berisikan data yang alamatnya ditunjuk x
```

LAB 1.3: LATIHAN

Percobaan 1 : Tipe data dasar

Perhatikan program di bawah ini, kemudian salin (ketik ulang, jangan copy-paste), *compile*, dan eksekusi program tersebut.

```
#include<iostream.h>
#include<conio.h>

void main()
{
    clrscr();

    cout<<"Here are the actual size of these data types:\n\n";
    cout<<"char: "<<sizeof(char)<<"\n";
    cout<<"int: "<<sizeof(int)<<"\n";
    cout<<"short int: "<<sizeof(short int)<<"\n";
    cout<<"long int: "<<sizeof(long int)<<"\n";
    cout<<"float: "<<sizeof(float)<<"\n";
    cout<<"double: "<<sizeof(double)<<"\n";
```

```
    getch();  
}
```

1. Apakah satuan dari ukuran tipe data yang menjadi keluaran dari program di atas?
2. Pada situasi seperti apa anda perlu menggunakan tipe data-tipe data yang muncul pada program di atas?

Percobaan 2 : Operator Binary

Berikut ini adalah sebuah program dalam bahasa C++. Salinlah program berikut menggunakan text editor dan simpan dengan nama `binary.cpp`. Kemudian *compile* dan *run* (eksekusi) program tersebut. Contoh di bawah ini hanya terdiri dari sebuah fungsi, yaitu fungsi `main`.

```
#include <iostream>  
  
int main()  
{  
    int alas, tinggi, luas;  
    cout << "Masukkan alas segitiga : ";  
    cin >> alas;  
    cout << "Masukkan tinggi segitiga : ";  
    cin >> tinggi;  
    luas = (alas * tinggi) / 2;  
    cout << "Luas segitiga = " << luas << endl;  
    return 0;  
}
```

3. Apakah terdapat kesalahan pada program tersebut? Bila terdapat kesalahan, cobalah perbaiki program tersebut hingga dapat di-*compile* dan memberikan hasil.
4. Apakah hasil pembagian sesuai ? Bagaimanakah cara menampilkan hasil pembagian yang berupa bilangan pecahan? Jelaskan!

Percobaan 2 : Operator Binary Percobaan 3 : Unary

Pada percobaan ini, salinlah program berikut menggunakan *text editor* dan simpan masing-masing dengan nama `unary.cpp`. Kemudian *compile* dan eksekusi program tersebut.

```
#include <iostream>  
  
int main()  
{  
    int first_number, second_number, result;  
    cout << "Masukkan angka pertama : ";  
    cin >> first_number;  
    cout << "Masukkan angka kedua : ";  
    cin >> second_number;  
    result = first_number++ * second_number++;  
    cout << "Hasil = " << result << endl;  
    return 0;  
}
```

1. Apakah terdapat kesalahan pada program tersebut? Bila terdapat kesalahan, cobalah perbaiki

- program tersebut hingga dapat di-*compile* dan memberikan hasil.
2. Apakah fungsi dari increment ?
 3. Apakah perbedaan antara letak operator increment '++' ? Jelaskan!

Percobaan 4 : Pointer

Program di bawah ini adalah program yang memanfaatkan pointer. Salin, *compile*, dan eksekusi program di berikut ini.

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

int main ()
{
    int Firstvalue = 5, Secondvalue = 15;
    int * p1, * p2;

    p1 = &Firstvalue;      // p1 = address of firstvalue
    p2 = &Secondvalue;     // p2 = address of secondvalue
    *p1 = 10;              // value pointed to by p1 = 10
    *p2 = *p1;             // value pointed to by p2 = value pointed to by p1
    p1 = p2;               // p1 = p2 (value of pointer is copied)
    *p1 = 20;              // value pointed to by p1 = 20

    cout << "firstvalue is " << firstvalue << '\n';
    cout << "secondvalue is " << secondvalue << '\n';
    return 0;
}
```

1. Apakah terdapat kesalahan pada program tersebut? Bila terdapat kesalahan, cobalah perbaiki program tersebut hingga dapat di-*compile* dan memberikan hasil yang benar!
2. Apakah tujuan dari program di atas?

LAB 1.4: CEK POIN 1

Perhatikan program di bawah ini, kemudian salin (ketik ulang, jangan copy-paste), *compile*, dan eksekusi program tersebut. Jika terdapat syntax error atau error karena kesalahan dalam hal penggunaan tipe data, operator, ataupun kesalahan logika, perbaiki hingga dapat dikompilasi dan dijalankan sehingga menghasilkan luaran seperti yang diharapkan.

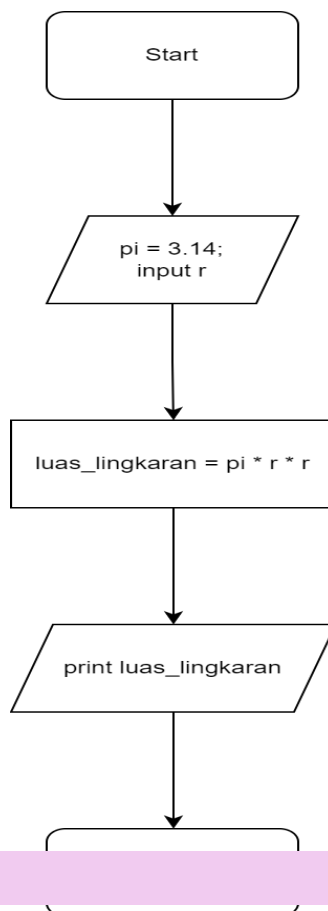
```
#include<iostream>
using namespace std;
int main()
{
    float *pcel , *pfah, fah, cel;
    pcel=cel;
    pfah=fah;

    cout<<"Masukkan suhu dalam Fahrenheit : ";
    cin>>*fah;
    *cel=(*pfah - 32) * 5/9;
```

```
cout<<"Suhu dalam celcius adalah : " <<cel<<endl;  
return 0;  
}
```

LAB 1.5: CEK POIN 2

Berikut adalah *flowchart* dari program penghitung luas lingkaran. Buatlah kode program berdasarkan flowchart berikut! Kemudian *compile* dan eksekusi program anda. Apabila terdapat *syntax error* (kesalahan), perbaiki sampai program anda dapat sukses terkompilasi dan dapat dieksekusi.



LAB 1.6: CEK POIN 3

Berdasarkan *flowchart* untuk menghitung luas lingkaran pada cek poin 2, buatlah sebuah program yang dapat digunakan untuk menghitung volume tabung dan kerucut! Kemudian *compile* dan eksekusi program anda,

LAB 1.7: REFERENSI

Di bawah ini adalah referensi online dari Modul 1. Apabila mengalami kesulitan dalam memahami materi, silahkan buka referensi di bawah ini :

1. https://www.tutorialspoint.com/cplusplus/cpp_operators.html
2. <https://www.geeksforgeeks.org/c-data-types/>
3. <https://www.duniailkom.com/tutorial-belajar-c-plus-plus-jenis-jenis-operator-dalam-bahasa-c-plus-plus/>

LAB 1.8: SEKILAS MATERI UNIT BERIKUTNYA

Materi praktikum pemrograman dasar unit selanjutnya akan lebih banyak membahas beberapa hal, terkait dengan sebuah kontrol program, yaitu seleksi (pemilihan kondisi). Beberapa hal yang patut Anda ketahui, antara lain

1. Pengertian dasar kendali program seleksi.
2. Perbedaan dasar antara flowchart dan pseudocode.
3. Operasi-operasi dengan algoritma seleksi (IF, IF-ELSE, SWITCH, CASE)