C++ adalah bahasa yang berbasis C tetapi lebih kompleks dan modern yang bisa OOP.

**UNSUR UTAMA C++ (HELLO WORLD)**

#include <iostream>

int main (){

    std::cout <<"hello world"<< std::endl;

    return 0;

}

**Iostream** itu adalah kode/ file yang sudah ada di windows kita (filesystem), digunakan untuk membaca dan mengluarkan hasil dari suatu program.

**int main** ini adalah fungsi untama, dikarenakan cpp harus menggunakan setidaknya satu fungsi.

**std** adalah standar namespase yang di gunakan untuk menampilkan sesuatu di consol.

**endL** berguna untuk mengakhiri suatu baris kode.

Untuk menjalan c++ bisa menggunakan compiler g++

A diagram of a process

Description automatically generated

**Preprocessing, Compiling, dan Linking**

Pengertian

1. Preprocessing

Sederhananya, preprocessing itu pegabungan antara file c++ dengan file iostream yang mengahsilan file prepocessing

1. Compiling

Sederhananya, compiling itu pengubahan dari tugas preprocessing menjadi bahasa mesin yang di sebut obj file

1. Linking

Sederhanya, linking adalah pegabungan dari beberapa obj obj file menjadi satu yang membuat file exe

**Printing C++**

**UNSUR UTAMA C++ (HELLO WORLD)**

#include <iostream>

int main (){

    std::cout <<"hello world"<< std::endl;

    return 0;

}

**Iostream** itu adalah kode/ file yang sudah ada di windows kita (filesystem), digunakan untuk membaca dan mengluarkan hasil dari suatu program.

**int main** ini adalah fungsi untama, dikarenakan cpp harus menggunakan setidaknya satu fungsi.

**std** adalah standar namespase yang di gunakan untuk menampilkan sesuatu di consol. Std menggunakan :: digunakan untuk menyambungkan std dengan cout itu sendiri.

**endL** berguna untuk mengenter kata di consol, bisa juga menggunakan \n.

    cout <<"hello world\n";

**return 0** digunakan untuk menghasilkan “tidak ada gagal” ke terminal

**Tips :**

using namespace std;

gunukan ini untuk tidak menulis std manual lagi..

#include <iostream>

using namespace std;

int main (){

    cout <<"hello world"<< endl;

    return 0;

}

**Cin.get**

    cin.get();

cin.get digunakan untuk fungsi enter di keyboard

**Variabel dan tipe data di c++**

1. Int = bilangan bulat

#include <iostream>

using namespace std;

int main (){

    int a ;

    a = 10;

    cout << a << endl;

    cin.get();

    return 0;

}

Maka yang ditampilkan adalah 10

**CIN (MENGAHSILKAN INPUT)**

#include <iostream>

using namespace std;

int main (){

    int b;

    cout << "masukan nilai" << endl;

    cin >> b;

    cout << "nilai yang anda masukan ";

    cout << b << endl;

}

Untuk manghasilkan input gunakan **cin >> variabel\_yang\_diinputkan ;**

**TIPE DATA DASAR**

Penulisan tipe data di c++ dapat berupa :

tipedata nama\_variabel = nilai\_variabel

    // bilangan bulat

    int a = 5;

    long b = 6;

    short c = 7;

    // Bilangan desimal (punya koma)

    float d = 1.0;

    double e = 2.5;

    // character

    char f = 'a';

    // boolean (nilai benar atau salah)

    bool g = true;

**int , long , short** itu adalah **bilangan bulat**. **Float, doble** adalah bilangan desimal (punya koma). Char menyimpan satu karakter. **Bool** itu boolean yang menyimpan nilai benar(1) dan salah (0)

    // String

    string h = "dah";

    cout << h ;

string digunakan untuk menyimpan data.

**ARITMATIKA DASAR**

    int a = 6;

    int b = 4;

    int hasil;

Pertama kita deklarasikan a dan juga b

**Pertambahan**

    // penjumlahan

    hasil = a + b;

    cout << a << " + " << b << " = " << hasil << endl;

**Pengurangan**

    // pengurangan

    hasil = a - b;

    cout << a << " - " << b << " = " << hasil << endl;

**Perkalian**

    // perkalian

    hasil = a \* b;

    cout << a << " x " << b << " = " << hasil << endl;

**Pembagian**

Dikarenakan pembagian 6 : 4 itu menghasilkan koma, maka di sini kita mending kasih saja float. Ketika kalian mau melakukan apa pun yang mempunyai koma, tambahkan float!!!!

    // pembagian

    float a\_bagi = 6;

    float b\_bagi = 4;

    float hasil\_bagi = a\_bagi / b\_bagi;

    cout << a\_bagi << " / " << b\_bagi << " = " << hasil\_bagi << endl;

**Modulus (sisa bagi)**

Modulus adalah sisa bagi, contoh 6 bagi 4 itu 2 sisa **2.**

**KOMPARASI**

Sebelum membandingkan suatunilai kita harus mendeklarasikan terlebih dahlulu nilai nya

    int a = 2;

    int b = 2;

    bool hasil1, hasil2;

**1. Komparasi Sebanding**

    // sebanding

    cout << (a == b) << endl;

**“==”** berarti sebanding, alias nilainya sama. Jika nilainya sama akan menghasilkan 1 (TRUE) dan jika tidak sama akan menghasilkan 0 (FALSE)

**2. Komparasi Tidak sebanding**

Komperasi tidak sebanding di tandai dengan “!” yang memiliki logika seperti NOT.

    // tidak sebanding

    hasil2 = (a != b);

    cout << hasil2 << endl;

karena “2 tidak sama dengan 2” Itu salah , maka akan menghasilkan nilai FALSE (0).

**3. Komparasi lebih dari dan kurang dari**

Komperasi lebih dari ditandai dengan “>” , dan kurang dari ditandai dengan “<”. Agar lebih paham, kita ubah nilainya terlebih dahulu

    int a = 2;

    int b = 3;

    // Kurang dari dan Lebih dari.

    hasil1 = (a < b);

    hasil2 = (a > b);

    cout << "Apakah a < b?" << " " << hasil1 <<endl;

    cout << "Apakah a > b?" << " " << hasil2;

Dari operasi terebut yang pertama, apakah a(2) < b(3) , jawaban TRUE(1). Kedua, apakah a(2) > b(3), jawaban FALSE(0)

**4. Komparasi lebih dari dan kurang dari yang disi sama dengan**

Komperasi lebih dari sama dengan itu di tandai dengan = “>=”, kurang dari sama dengan ditandai dengna “<=”

    hasil1 = (a <= b);

    hasil2 = (a >= b);

    cout << "Apakah a <= b?" << " " << hasil1 <<endl;

    cout << "Apakah a >= b?" << " " << hasil2;

Ini adalah kombinasi dari kurang dari ataupun lebih dari dengan sama dengan ini bersifat salah satu atau keduanya benar tetap benar . Contoh a(2) >= b(2) = TRUE (1). a(2) <= b(2) = TRUE (1).

**Dasar Operator Logika**

Operasi logika yang akan di sini di bahas adalah OR, AND dan NOT, Yukk carii tauu!!. Sebelum itu kita harus tau logo logo dari operasi logika, OR **= ||** , AND = **&&** , NOT = **!**.

**Inisalisasikan Var**

    int a = 3;

    int b = 2;

    bool hasil;

Dinisini kita memakai bool agar nilainya 1 (BENAR / TRUE ) atau 0 (FALSE)

**NOT**

    // NOT

    cout << "Operasi NOT \n";

    hasil = !(a == 3);

    cout << "Operator NOT: " << hasil << endl;

Nilai not adalah kebalikan. **Dari seharusnya benar (1) menjadi salah (0)** .

**AND**

    // AND

    cout << "Operasi AND \n";

    hasil = (a == 3 && b == 2);

    cout << "Operator AND, TRUE & TRUE : " << hasil << endl;

    hasil = (a == 2 && b == 2);

    cout << "Operator AND, FALSE & TRUE : " << hasil << endl;

    hasil = (a == 10 && b == 10);

    cout << "Operator AND, FALSE & FALSE : " << hasil << endl;

Nilai AND akan benar (1) jika kedua nilainya benar, Misal nilai a = 3 dan nilai b = 2 , dan memang benar hasilnya bedasarkan deklarasi, maka akan mengerluarkan nilai benar (1). AND bisa di ibaratkan dengan “Saya ingin makan nasi dan lauk” yang dimana kita mau keduanya.

**OR**

    // OR

    cout << "Operasi OR \n";

    hasil = (a == 3 || b == 2);

    cout << "Operator OR, TRUE & TRUE : " << hasil << endl;

    hasil = (a == 2 || b == 2);

    cout << "Operator OR, FALSE & TRUE : " << hasil << endl;

    hasil = (a == 10 || b == 10);

    cout << "Operator OR, FALSE & FALSE : " << hasil << endl;

Nilai OR akan benar (1) jika salah satu pernyataan sudah benar, ataupun kedua nya **yang penting tidak keduanya false.** Or di ibaratkan dengan “ Saya ingin belanja sesuatu ke indomaret ataupun ke alfamart” yang dimana kita memiliki 2 opsi yang bisa salah satu, asal kita belanja ke minimarket .