Pengertian Function

Function (Fungsi, arti dalam bahasa Indonesia). Function adalah sebuah struktur, Pengelompokan yang mengandung sekelompok pernyataan yang akan dilaksanakan oleh compiler jika nama function tersebut dipanggil untuk dieksekusi, kecuali untuk function utama “int main()” yang akan dieksekusi secara otomatis.

Sebelumnya kita sering meilihat tanpa mengerti “int main()” pada contoh program yang pernah penulis berikan. Pada sebuah program memiliki satu function utama dan merupakan awal program berjalan atau compiler melakukan eksekusi dari sekelompok pernyataan yang diberikan, semua itu akan di mulai dari function “int main()”, function tersebut merupakan function utama yang akan menjadi kepala sebuah program dimana semua eksekusi pernyataan berawal.

Bisakah anda bayangkan melakukan pemrograman sebuah program yang sangat-sangat besar dan memerlukan pernyataan yang sangat panjang, apa lagi jika sebuah proses mempunyai banyak pernyataan dan dibutuhkan tidak hanya sekali, itu pasti akan membuat anda pusing jika anda menaruh semua pernyataan itu pada satu function. Maka, disini Function sangat membantu kita untuk membuat pengelompokan pernyataan berdasarkan tugas, proses atau katagori.

Bentuk Umum Penulisan

tipe nama (parameter1, parameter2, …){

Pernyataan

}

Keterangan :

Tipe = adalah sebuah tipe data dari Function, beberapa fungsi dapat berisikan nilai hasil operasi yang dilakukan di dalam function tersebut. tapi juga ada function yang tidak akan mengembalikan nilai atau tidak berisikan nilai dari operasi di dalam function tersebut, untuk membuat function tanpa melakukan pengembalian nilai kita bisa menggunakan tipe data “void”.

Nama = nama atau identifier, adalah sebuah identitas atau nama dari function tersebut.

Parameter = kita dapat mengkosongi jika tidak membutuhkan. Parameter adalah pemesanan memori untuk memberikan argumen yang berisi nilai disaat pemanggilan function, seperti deklarasi variabel (Memang seperti variabel tapi parameter kita tidak bisa menyebutnya sebagai variabel ) di dalam function yang akan diberikan argument di saat pemanggilan function tersebut. parameter tersebut akan bersifat lokal bagi function tersebut, tidak bisa digunakan diluar function. Kita bisa membuat parameter lebih dari satu, dan masing-masing parameter akan dipisah dengan tanda koma.

Pernyataan = adalah tempat dimana pernyataan-pernyataan di tulis. Semua itu Berada di tengah-tengah { }.

Deklarasi Function

Untuk bentuk penulisan sama seperti apa yang pernah kita tulis untuk membuat function utama yaitu “int main()”, tapi yang berbeda adalah kita bebas memberi nama function dan memberi return type pada function buatan kita, Kita bisa menambahkan paramenter untuk nilai awal atau arguments pada parameter yang akan diproses saat function di jalankan. Dan terakhir adalah function ini tidak akan dibaca oleh compiler jika nama function buatan tidak dipanggil di dalam function utama.

Contoh Penulisan

//Tanpa Parameter

void cetakPesan (){

cout<<”Belajar Function C++”;

}

//Dengan Parameter

Int tambah (int a, int b){

result = a+b;

return a+b;

}

Pada contoh pertama merupakan contoh yang menunjukan bentuk function tanpa menggunakan parameter. Dan pada contoh kedua merupakan contoh menggunakan dua parameter yang dipisahkan dengan tanda koma (,). pada baris ke satu d function dengan parameter “result = a+b” akan memberi hasil penjumlahan dari dua parameter ‘a’ dan ‘b’ kepada variabel global “result”. Untuk baris kedua “return a+b” akan memberikan hasil penjumlahan dari dua parameter ‘a’ dan ‘b’ kepada function bertipe data int. sehingga ketika kita memanggil function itu akan memberikan nilai hasil proses penjumlahan tersebut.

Cara Mengakses Fuction

Compiler Hanya akan melaksanakan pernyataan-pernyataan yang berada di dalam function utama “int main”. Saat mendirikan sebuah function sendiri, pernyataan-pernyataan yang ada di dalam function buatan kita, tidak akan dibaca oleh compiler jika nama atau identifier pada function yang kita buat tidak dipanggil di dalam fungsi utama.

Jalur pembacaan compiler, compiler akan melaksanakan semua pernyataan yang dilewati. Ketika compiler itu melewati sebuah pernyataan nama dari function maka compiler akan melaksanakan semua pernyataan di dalam function itu hingga selesai. jika sudah, compiler akan melanjutkan pembacaan pada function sebelumnya. Kita juga bisa membuat function buatan memanggil function buatan lainya.

Untuk memanggil function, kita cukup menuliskan nama dari function tersebut dan memberi sepasang tanda “()”, itu merupakan tempat dimana nilai arguments untuk parameter function tersebut, kita bisa mengkosonginya jika memang function tersebut tidak mengandung parameter.

Contoh Penulisan

//Function Tanpa Parameter

cetakPesan ();

//Function Dengan Parameter

tambah (argument1, argument2);

Contoh Program

//Deklarasi dan Pemanggilan Function

#include <iostream>

using namespace std;

int result;

void cetakPesan (){

cout<<"Belajar C++"<<endl<<endl;

}

int tambah (int a, int b){

result = a+b;

//atau kita bisa menggunakan return

return a+b;

}

int main () {

cetakPesan();

cout<<"result = "<<result<<endl;

cout<<"tambah(5,6) = "<<tambah(5,6)<<endl;

return 0;

}

Bukan hanya itu saja, Function tersebut juga bisa memanggil function itu sendiri. Itu sangat membantu untuk melakukan rumus matematika, salah satunnya seperti rumus factorial.

Funtion Bawaan

Pada bahasa pemrograman sepeti C++ biasanya sudah disediakan beberapa function bawaan yang bisa membantu kita, seperti beberapa pernyataan yang penulis berikan pada contoh-contoh sebelumnya. seperti strcpy(), printf() dan lain-lain. Keyword-keyword tersebut juga bisa disebut sebagai fungsi. Karena memang bentuk dan cara penggunaan mirip seperti function yang telah dijelaskan di atas. function-function tersebut sudah disediakan dari awal oleh pembuat bahasa pemrograman untuk mempermudah kita dalam membuat program.