Function overloaded adalah salah satu fitur C/C++ yang memungkinkan kita untuk membuat atau mendirikan dua atau lebih function dengan nama yang sama di dalam satu scope. Penulis pernah menulis di salah satu artikel sebelumnya bahwa dalam mendirikan variabel dalam satu scope kita tidak diperbolehkan untuk membuat lebih deklarasi dengan nama yang sama, hal itu akan membuat pesan error dari compiler.

Tapi itu berbeda dengan metode overloading, function pada pemrograman C/C++ kita diperbolehkan untuk mendirikan dua function dengan nama atau identifier yang sama, lebih dari satu dalam satu scope. Overloading dapat terjadi jika nama atau identifier itu sama dengan function lainya dan masing-masing function mempunyai patameter yang berbeda.

Meskipun di pengertian di atas kita memperbolehkan membuat function yang sama bukan berarti function itu kembar identik, compiler akan memberikan pesan error jika deklarasi function sangat sama dan tidak ada perbedaan. Perbedaan yang penulis maksud adalah patameter, bukan return type pada function.

Compiler akan mebaca deklarasi function buatan kita (selain main function) sebelum masuk ke function utama, dan mengakui bahwa function yang kita buat itu ada dan compiler sudah mengetahuinya, sehingga compiler akan melakukan pembacaan pada isi function buatan kita, ketika kita memanggil nama function buatan kita itu.

Dengan adanya function overloading, dan ketika kita memanggil function overloadin compiler akan menentukan sendiri function mana yang harus dimasuki berdasarkan argument yang diberikan. Maka dari itu kita harus melakukan perbedaan pada function overloading. Overloading tidak akan terjadi jika kita hanya melakukan perbedaan pada return type. Overloading akan sangat membantu , salah satunya adalah ketika kita membuat function dengan tugas di dalamnya, tetapi function tersebut harus bisa menerima perbedaan argument, meskipun pengguna melakukan input dengan type data yang berbeda-beda program tetap bisa menyelesaikan tugasnya dengan baik dan benar susuai apa yang programmer dan pengguna harapkan.

Contoh program

#include <iostream>

#include <string>

using namespace std;

void myFunction(int a){

cout<<"int myFunction = "<<a<<endl;

}

void myFunction(double a){

cout<<"Double myFunction = "<<a<<endl;

}

void myFunction(char a){

cout<<"Char myFunction = "<<a<<endl;

}

void myFunction(string a){

cout<<"String myFunction = "<<a<<endl;

}

int main() {

myFunction(1.298);

myFunction(20);

myFunction('z');

myFunction("Belajarcpp.com");

return 0;

}

Di atas adalah contoh dari penggunaan function overloaded, di atas memliki 4 function yang menggunakan nama yang sama dan masing-masing mempunyai parameter berbeda. Compiler akan menentuan function mana yang akan dimasuki berdasarkan argument pada permanggilan function tersebut.