

**UNIVERSIDADE FEDERAL DE UBERLÂNDIA**

**FACULDADE DE ENGENHARIA MECÂNICA**

GRADUAÇÃO EM ENGENHARIA MECATRÔNICA

**Semana 02 – Sistemas Embarcados II**

**Ambiente de Programação Linux**

Fernando Rabelo Fernandes Junior - 11611EMT020

Uberlândia, dezembro de 2021

1. De modo geral, a compilação funciona lendo o código fonte inteiro, realizando a coleta e a organização das informações contidas e fazendo a tradução completa, podendo ser um executável. Isso é feito através da tradução das linhas do código para outra linguagem, como por exemplo, C.
2. // C program showing the header file including

// standard input-output header file

#include <stdio.h>

int main()

{

// "printf()" belongs to stdio.h

printf("hello world");

return 0;

}

// It is not recommended to put function definitions

// in a header file. Ideally there should be only

// function declarations. Purpose of this code is

// to only demonstrate working of header files.

void add(int a, int b)

{

printf("Added value=%d\n", a + b);

}

void multiply(int a, int b)

{

printf("Multiplied value=%d\n", a \* b);

}

// C program to illustrate file inclusion

// <> used to import system header file

#include <stdio.h>

// " " used to import user-defined file

#include "process.h"

// main function

int main()

{

// add function defined in process.h

add(10, 20);

// mult function defined in process.h

multiply(10, 20);

// printf defined in stdio.h

printf("Process completed");

return 0;

}

Incluindo o arquivo “process.h” em outro programa. Agora, como precisamos incluir stdio.h como #include para usar a função printf () de maneira semelhante, também precisamos incluir o arquivo de cabeçalho process.h como #include “process.h”. O ”” instrui o pré-processador a examinar a pasta atual ou a pasta padrão de todos os arquivos de cabeçalho, se não for encontrado na pasta atual. Se colchetes angulares forem usados ​​em vez de ””, será necessário salvá-lo na pasta padrão de arquivos de cabeçalho. Se estiver usando ””, você precisa garantir que o arquivo de cabeçalho criado seja salvo na mesma pasta em que o arquivo C atual usando este arquivo de cabeçalho é salvo.

1. -static: vincula contra bibliotecas estáticas

-g: incluem informações de depuração padrão no binário.

-pendantic: emite todos os avisos requeridos pela ANSI/ISSO C padrão.

-Wall: emite todos os avisos geralmente úteis que o gcc pode fornecer.

-Os: instrui o gcc a “otimizar ao tamanho”. Ativa todas as otimizações de -O2 que não aumentam o tamanho do executável e, em seguida, também alterna alguns sinalizadores de otimização para reduzir ainda mais o tamanho do executável.

-O3: tenta otimizar o código fortemente para o desempenho. Inclui todas as otimizações -O2 e mais algumas.