

## Opções de compilação do compilador “gcc”

1. Gerar o código executável a partir de um código fonte
  - a. Suponha que o arquivo contendo o código fonte seja: “teste.c”
  - b. A linha a ser executada na linha de comando MS-DOS será:  

```
gcc teste.c -o teste
```

Será gerado um arquivo denominado “teste.exe” no diretório atual.

**Nota 1:** se fosse digitada somente a linha “gcc teste.c”, o compilador gcc gerará um executável com o nome padrão “a.out”.

**Nota 2:** no UNIX o comando deve ser “gcc ./teste.c -o teste” e o nome do arquivo gerado será “teste”.

2. Somente compilar um arquivo fonte:
  - a. Suponha que o arquivo contendo o código fonte seja: “teste.c”
  - b. A linha a ser executada na linha de comando MS-DOS será:  

```
gcc teste.c -c
```

Essa linha gerará um arquivo denominado “teste.o” contendo somente o código objeto<sup>1</sup> relativo ao código fonte “teste.c”.

Se forem listados os arquivos no diretório atual poderá ser visto que o tamanho do arquivo “teste.o” é bem menor do que o código executável (arquivo com extensão “.exe”).

**Nota 1:** dê preferência a opção somente de compilação enquanto estiver em fase de depuração do seu programa (encontrar erros no programa). Isso porque essa opção é mais rápida do que se você pedir para gerar o código executável.

**Nota 2:** no UNIX o comando deve ser “gcc ./teste.c -c”

3. Compilar arquivos fontes que incluam outros arquivos cabeçalhos criados por você próprio (extensão “.h”):
  - a. Suponha que tenha criado o seguinte arquivo fonte e salvo ele como “teste.c”:

```
#include “funcoes.h”  
main()  
{  
    funcao1();  
}
```

---

<sup>1</sup> Foge ao escopo deste curso a definição precisa do que é um código objeto. Entenda somente que é um código

- b. Suponha que tenha criado também o arquivo cabeçalho “funcoes.h” seguinte e que ele esteja no diretório “c:\funcoes\minhas”

```
void funcao1(void);
```

**Lembrete:** um arquivo “.h” deve conter somente o cabeçalho das funções, portanto o conteúdo desse arquivo “funcoes.h” está correto!

- c. Se fosse digitado o comando “gcc teste.c -c” ocorreria um erro, pois o gcc não encontraria o arquivo de cabeçalho “funcoes.h” para incluir no arquivo quando da compilação. Para isso, deve ser indicado explicitamente ao compilador onde procurar o arquivo cabeçalho por meio da opção “-I”:

```
gcc teste.c -c -I c:\funcoes\minhas
```

Desse modo, o gcc irá procurar no diretório “c:\funções\minhas” o arquivo cabeçalho “funções.h” na hora de compilar o arquivo “teste.c”.

**Nota 1:** no UNIX seria “gcc ./teste.c -c -I /funcoes/minhas”

4. Gerar executáveis a partir de arquivos fontes que incluam outros arquivos cabeçalhos criados por você próprio (extensão “.h”):

- a. Suponha que tenha criado o seguinte arquivo fonte e salvo ele como “teste.c”:

```
#include “funcoes.h”
main()
{
    funcao1();
}
```

- b. Suponha que tenha criado também o arquivo cabeçalho “funcoes.h” seguinte e que ele esteja no diretório “c:\funcoes\minhas”:

```
void funcao1(void);
```

- c. Suponha que também tenha criado o arquivo “funcoes.c” contendo o seguinte código:

```
#include <stdio.h>
void funcao1(void)
{
    printf(“Estou dentro da funcao 1”);
}
```

- d. Para gerar o código de máquina, a seguinte seqüência de passos deve ser executada:

- Compile o arquivo fonte da biblioteca sem gerar o executável:  
*gcc funcoes.c -c*

**Nota:** é gerado o arquivo “funcoes.o”

- Usar o comando **ar** com as opções **rc** para gerar uma biblioteca a partir do código objeto do arquivo “funcoes.o”

*ar rc libfuncoes.a funcoes.o*

**Nota 1:** é gerado o arquivo de biblioteca “libfuncoes.a”

**Nota 2:** para saber o que significam as opções “rc” digite “ar --help” na linha de comando para obter a ajuda disponível para o programa “ar”.

- Finalmente, para ligar a biblioteca gerada “libfuncoes.a” junto com o código objeto relativo ao programa (arquivo “teste.c”) deve digitar o seguinte comando:

*gcc teste.c -o teste -L. -lfuncoes*

**Nota 1:** esta linha vai gerar o arquivo executável “teste.exe”

**Nota 2:** a opção “-L” serve para indicar em qual diretório deve ser procurado o arquivo de biblioteca que será ligado. No caso, como foi especificado o caminho “.”, então irá procurar no diretório atual.

**Nota 3:** a opção “-l” (L minúsculo) serve para indicar o nome da biblioteca que deve ser ligada ao código do programa “teste.c” para poder ser gerado o código executável (arquivo “teste.exe”). Observe que deve ser digitado somente o nome “funcoes”.

A partir desse nome o gcc acrescenta **automaticamente** a extensão “.a” a esse nome e também acrescenta o termo “lib” na frente desse nome, ou seja, o gcc **irá procurar na prática pela biblioteca denominada “libfuncoes.a”**.

Foi por isso que foi dado esse nome para a biblioteca (nome “libfuncoes.a”) quando da execução do programa **ar** (ver **acima**).

**Nota 4:** no UNIX seria “*gcc ./teste.c -o teste -L. -lfuncoes*”

- e. Agora é só digitar “teste” na linha de comando, que o programa executará normalmente.