

Comandos Linux Importantes

- **sudo:** executa o comando seguinte como administrador da maquina, é necessário senha;
- **cd:** acessar um diretorio, como por exemplo cd diretório, cd .. , cd /;
- **ls:** Lista todos os arquivos do diretório;
- **df:** Mostra a quantidade de espaço usada no disco rígido;
- **top:** Mostra o uso da memória;
- **cmp:** Compara dois arquivos;
- **comm:** Seleciona ou rejeita linhas comuns a dois arquivos selecionados;
- **cp:** Copia arquivos;
- **mkdir:** Cria um diretório;
- **rm:** Apaga arquivos ou diretorios, é semelhante ao comando del no Windows, é preciso ter cuidado com o comando `rm *` pois apaga tudo sem confirmação;
- **rmdir:** Apaga diretório;

Usando o compilador no terminal do linux

Usamos o comando abaixo para instalar o compilador da linguagem C no linux.

```
$ sudo apt-get install gcc
```

Com o gcc já instalado no seu sistema, é muito simples usá-lo para compilar programas em C. Se o programa consistir de um único arquivo, você pode simplesmente executar este comando no terminal:

```
$ gcc prog.c -o prog
```

Para executar o programa, a maneira mais “universal” é digitar o seguinte comando no terminal:

```
$ ./prog
```

onde *prog.c* é o nome do arquivo que contém o código. Os outros dois parâmetros, *-o prog*, indicam o arquivo de saída do compilador — o arquivo executável que conterá o programa. Você não verá nenhuma mensagem na tela se a compilação ocorrer sem problemas; o compilador só diz alguma coisa quando ocorrem erros.

Você precisa especificar o nome do arquivo executável de saída pois o padrão, por razões históricas, é usar o arquivo *a.out*. *Caso não seja especificado o nome do arquivo os comandos para compilar e executar o programa seriam:*

```
$ gcc prog.c
```

```
$ ./a.out
```

Ao fazer seus programas, é muito importante verificar erros de sintaxe ou quaisquer outros problemas no código. Algumas opções podem ajudá-lo a fazer isso:

-Wall

ativa vários avisos que não são ativados normalmente. Pode dar alguns conselhos úteis.

-pedantic

ativa alguns avisos que existem de acordo com o padrão C mas que são desativados normalmente por extensões do gcc.

-ansi

compila seu programa estritamente de acordo com o padrão ANSI, desativando qualquer extensão específica do gcc. (Como os professores costumam adotar o gcc, isso não é sempre necessário. Mas preste atenção às instruções do seu professor quanto à compilação.)

O gcc pode otimizar o seu programa de modo a aumentar seu desempenho e/ou diminuir o tamanho do código de máquina gerado. Por omissão, o gcc não realiza nenhuma otimização. Há três níveis de otimização: 1, 2 e 3. Quanto maior o nível, maior deve ser a melhora no desempenho; mas também deve ser maior o tempo de compilação.

Para ativar a otimização, use as opções -O1, -O2 ou -O3 (a letra O, não o número zero), de acordo com o nível de otimização que você desejar. Também existe a opção -Os, que realiza as mesmas otimizações de -O2, excetuando as que costumam aumentar o tamanho do executável gerado.