

2. O AMBIENTE DE PROGRAMAÇÃO EM C

Este capítulo trata de colocar a linguagem C para funcionar em um ambiente de programação, concentrando-se no compilador *GNU Compiler Collection* (gcc). Mas qualquer outro compilador aderente ao padrão da linguagem pode ser usado.

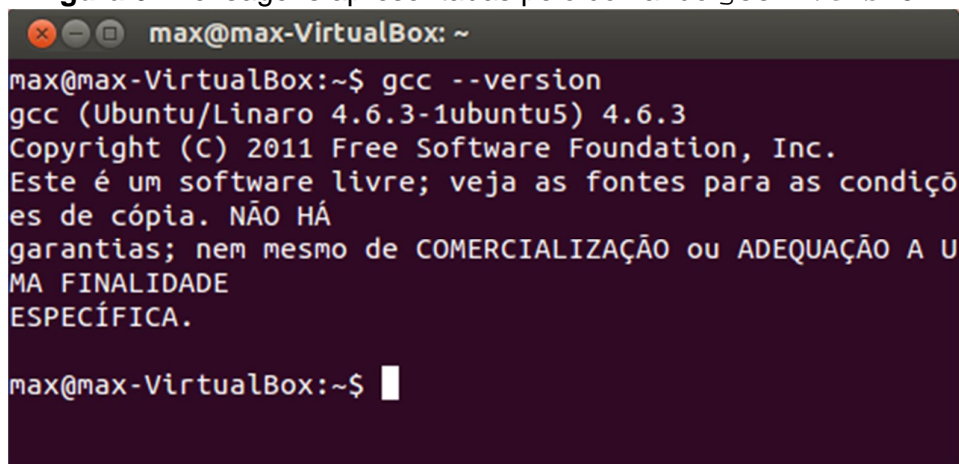
2.1. Instalação do Compilador gcc no Linux

No Linux o compilador gcc já vem instalado na maioria das distribuições Linux. Mas mesmo assim é recomendado fazer um teste, executando em um terminal o seguinte comando:

```
gcc --version
```

Se o compilador estiver presente, serão mostradas mensagens indicando o registo e a versão instalada, como mostra a **Figura 6**. Se o comando não for reconhecido, instale o gcc usando o instalador de pacotes da sua distribuição Linux.

Figura 6: Mensagens apresentadas pelo comando `gcc --version`



```
max@max-VirtualBox: ~  
max@max-VirtualBox:~$ gcc --version  
gcc (Ubuntu/Linaro 4.6.3-1ubuntu5) 4.6.3  
Copyright (C) 2011 Free Software Foundation, Inc.  
Este é um software livre; veja as fontes para as condições de cópia. NÃO HÁ  
garantias; nem mesmo de COMERCIALIZAÇÃO ou ADEQUAÇÃO A UMA FINALIDADE  
ESPECÍFICA.  
max@max-VirtualBox:~$
```

Recomendação



No site do autor (www.univasf.edu.br/~max.santana/fundamentosC) contém uma máquina virtual Linux (Ubuntu 14.04) com o mesmo ambiente de programação utilizado para compilar todos os programas deste livro.

2.2. Instalação do Compilador gcc no Windows

Existem alguns pacotes Windows que implementam um subconjunto do gcc, e os mais conhecidos são: Minimalist GNU for Windows (MinGW) e GNU + Cygnus + Windows (CygWin).



Curiosidade

Alguns editores de programas ou interface de desenvolvimento (IDE) já incorporam o compilador. Por exemplo, o Codeblocks DevC++ e o Microsoft Visual C++ (MSVC++).

2.2.1. MinGW

O MingGW é um ambiente de programação que implementa uma versão mínima do gcc, que foi adaptado para o desenvolvimento de programas nativos para o Windows. O MingGW fornece para o programador um conjunto de ferramentas de programação (*Open Source*), como o *Make* que é um arquivo com uma sequência de comandos para compilação de um programa.



Curiosidade

O MingGW não fornece uma *Portable Operating System Interface* (POSIX). E se você que desenvolver um programa que garanta a portabilidade do código-fonte a partir de um sistema operacional que atenda as normas POSIX é recomendado o Cygwin.

2.3. Editores de Texto

Se você já tem alguma experiência em programação, então provavelmente irá querer digitar o código-fonte em um ambiente integrado de desenvolvimento, do inglês *Integrated Development Environment* (IDE), mas geralmente os ambientes de desenvolvimento criam *makefiles* e regras próprias de compilação que acabam dificultando a aprendizagem de programação.

Principalmente para quem esta aprendendo a programar em C, é fundamental entender os comandos e os parâmetros de compilação, além de entender como se compila um projeto separado em módulos.

E o objetivo desse livro é ensinar programação em C em qualquer ambiente de programação. Dessa forma o evita que o programador iniciante fique dependente de uma IDE específica.



Dica

Aprenda a programar em uma IDE depois de entender todas as etapas de compilação de um programa.

2.3.1. Editores de Texto para Linux

No Linux há excelentes opções de editores de texto que podem ser utilizados para programação. Veja abaixo alguns exemplos de editores de texto Linux:

2.3.1.1. gedit

O gedit é um editor de texto padrão GUI do sistema operacional Linux e já vem com a maior parte das distribuições Linux. Ele é compatível com a codificação UTF-8 e tem suporte a maioria dos recursos de um editor de texto padrão, assim como muitas características avançadas, como: multilinguagem, verificação ortográfica, suporte extensivo ao destaque de sintaxe e um grande número de *plug-ins*.

2.3.1.2. kate

O editor de texto kate é um editor um pouco mais completo que o gedit e possui a opção de abrir vários arquivos simultaneamente e de acoplar um terminal virtual na parte inferior da tela. Dessa forma ele pode funcionar como um ambiente de desenvolvimento leve.

2.3.1.3. geany

O editor de texto geany tem características básicas de um ambiente de desenvolvimento integrado. Ele suporta vários tipos de arquivos (C, Java, PHP, HTML, Python, Perl, Pascal, XML e outros) e tem algumas características interessantes como: destaque de sintaxe, auto-completar, auto fechamento de *tags*, etc.

2.3.1.4. vi

O vi é um dos editores de textos mais utilizados no mundo Linux, e apresenta as seguintes características: permite abrir vários arquivos ao mesmo tempo,

possui sistema de autocorreção, auto-identação, seleção visual, macros, seleção vertical de texto, uso de expressões regulares, sintaxe colorida, etc.



Curiosidade

O editor de texto vi é um editor muito poderoso e o preferido de 9 entre 10 usuários Linux.

2.3.1.5. emacs

O emacs é muito mais do que um simples editor de texto, ele é um ambiente de trabalho, que permite fazer praticamente qualquer tarefa da computação: escrever documentos, mandar e receber e-mails, navegar na web, escutar música, bate papo no IRC, agendar compromissos, anotar endereços e telefones, etc.

2.3.2. Editores de Texto para Windows

Para usuários Windows não há muita opção como no caso do Linux. Uma boa opção é utilizar o notepad++, melhor que o bloco de notas.

O notepad++ é um editor de texto que oferece para os programadores uma interface agradável, com destaque de sintaxe e outras coisas mais.

2.4. Laboratório (Opcional): Instalando o MinGW

Este laboratório tem como finalidade mostrar o processo de instalação do ambiente de programação MinGW em uma máquina Windows. Estas instruções aplicam-se ao Windows 7.

2.4.1. Download do MinGW

Acesse o site <http://sourceforge.net/projects/mingw/files/> e obtenha a última versão do instalador *mingw-get-setup.exe*.

2.4.2. Instalação do MingGW

Execute o arquivo baixado e uma interface gráfica será iniciada, como mostra a **Figura 7**, e em seguida clique no botão *Install*, e uma nova interface gráfica

será aberta, solicitando as informações da instalação, como mostra a **Figura 8**, e aceite a opções sugeridas clicando no botão *Continue*.

Figura 7: Interface gráfica de instalação do MinGW

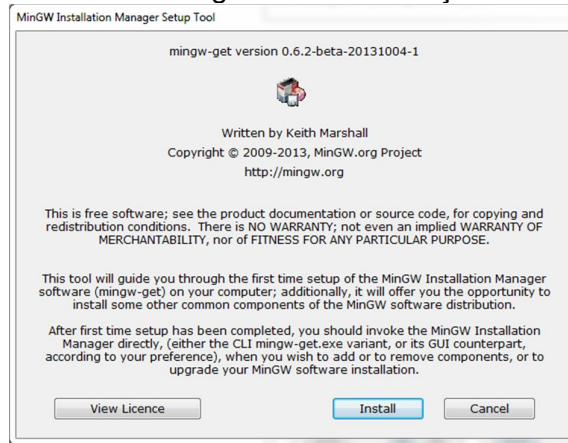
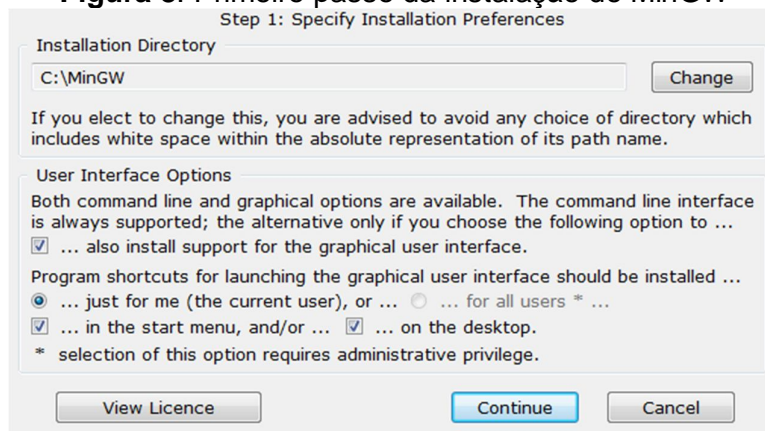


Figura 8: Primeiro passo da instalação do MinGW

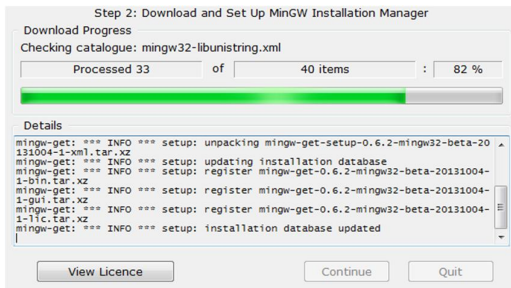


Dica

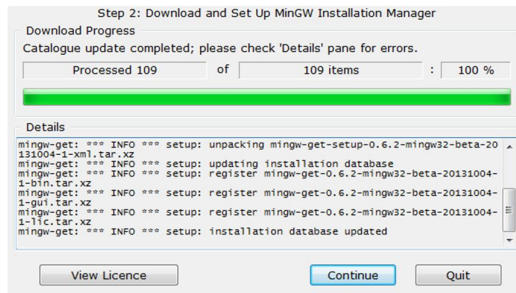
Aceite o diretório de instalação e as opções de interface. Caso queira modificar o diretório de instalação, o MinGW recomenda fortemente que o caminho indicado para instalação não contenha espaços.

Na interface gráfica seguinte o instalador irá baixar os pacotes necessários para a instalação, feito o download de todos os pacotes necessário clique no botão *Continue*, como pode ser visto nas **Figura 9 (a)** e **Figura 9 (b)**.

Figura 9: Baixando os pacotes necessários para instalação do MinGW



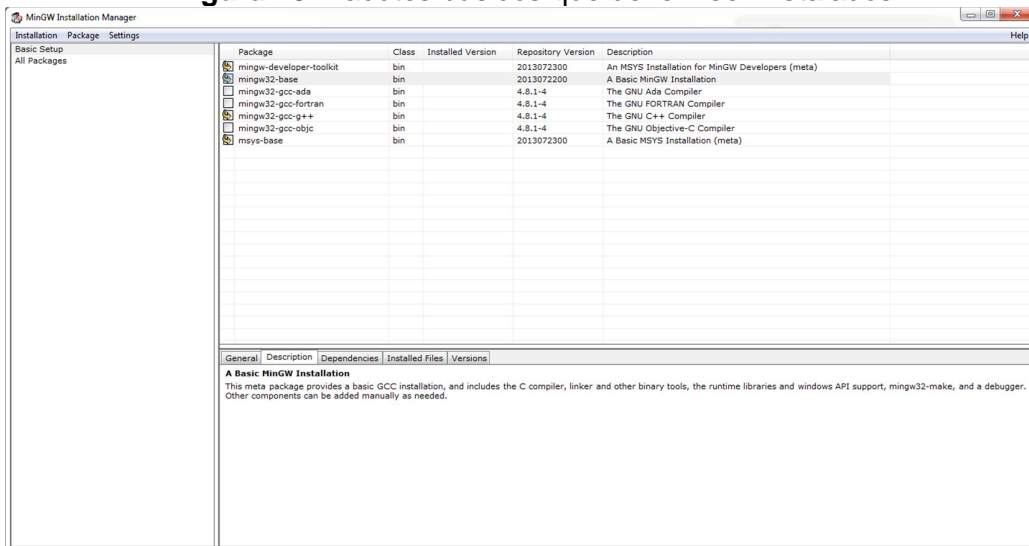
(a)



(b)

Em seguida devem ser selecionados os componentes (pacotes) que deseja instalar. É recomendável instalar *mingw-developer-toolkit*, *mingw32-base*, *mingw32-gcc-g++* e *msys-base*, como mostra a **Figura 10**, e clique em *Installation >> Apply changes*.

Figura 10: Pacotes básicos que devem ser instalados



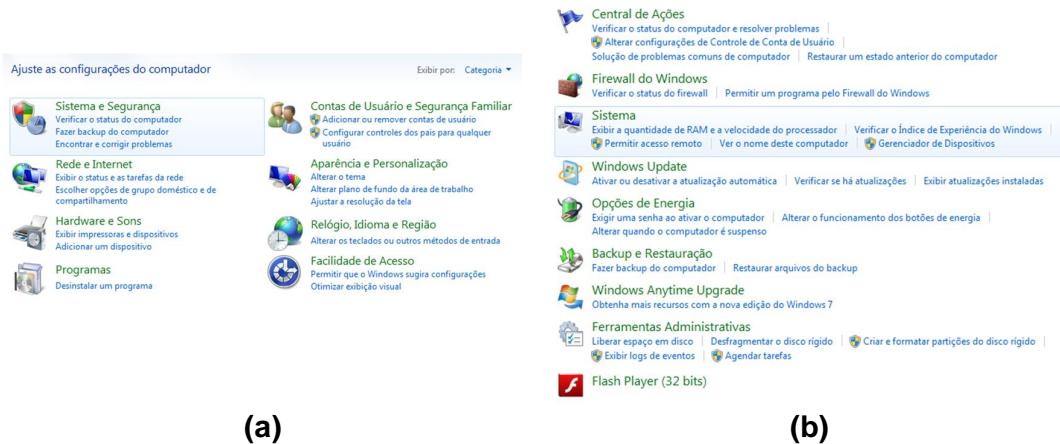
Após a instalação dos pacotes devemos adicionar à variável de ambiente **Path**, o caminho onde o MinGW foi instalado.

2.4.3. Configuração do MinGW

Como já foi mencionado, é necessário configurar a variável de ambiente **Path** do Windows. Essa configuração evita digitar o caminho inteiro do compilador gcc na hora de compilar o programa. Para configurar a variável ambiente, realize os seguintes procedimentos:

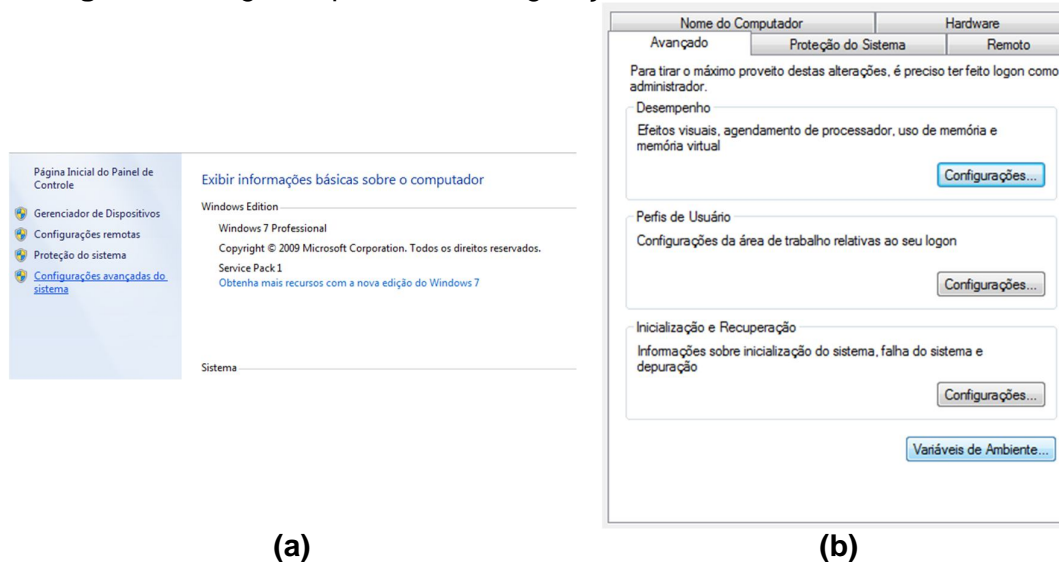
Abra o **Painel de controle** e escolha a opção **Sistema e Segurança** e depois **Sistema**, esse primeiro passo é mostrado nas **Figura 11 (a)** e **Figura 11 (b)**.

Figura 11: Primeiro passo da configuração da variável de ambiente Path



Na tela **Sistema**, clique na opção **Configurações avançadas do sistema**, e depois clique na aba **Avançados** e depois em **Variáveis de Ambiente**. Esse passo é mostrado nas **Figura 12 (a)** e **Figura 12 (a)**

Figura 12: Segundo passo da configuração da variável de ambiente Path



Na caixa **Variáveis do sistema**, selecione a variável **Path** e clique no botão **Editar**, como mostra a **Figura 13 (a)**. Quando aparece a janela de diálogo com duas caixas de texto, como mostra a **Figura 13 (b)**. No campo **Valor da variável**, adicione no final do campo o caminho onde foi instalado o MinGW, inserindo um ponto e vírgula (;), sem espaços, entre o último caminho existente na variável **Path** e o caminho do MinGW. As **Figura 13 (a)** e **Figura 13 (b)** mostram como são feitas essas configurações. Adicione também na variável Path o caminho onde foi instalado o processador de comandos MSYS, como mostra a **Figura 14**.

Figura 13: Último passo da configuração da variável de ambiente Path

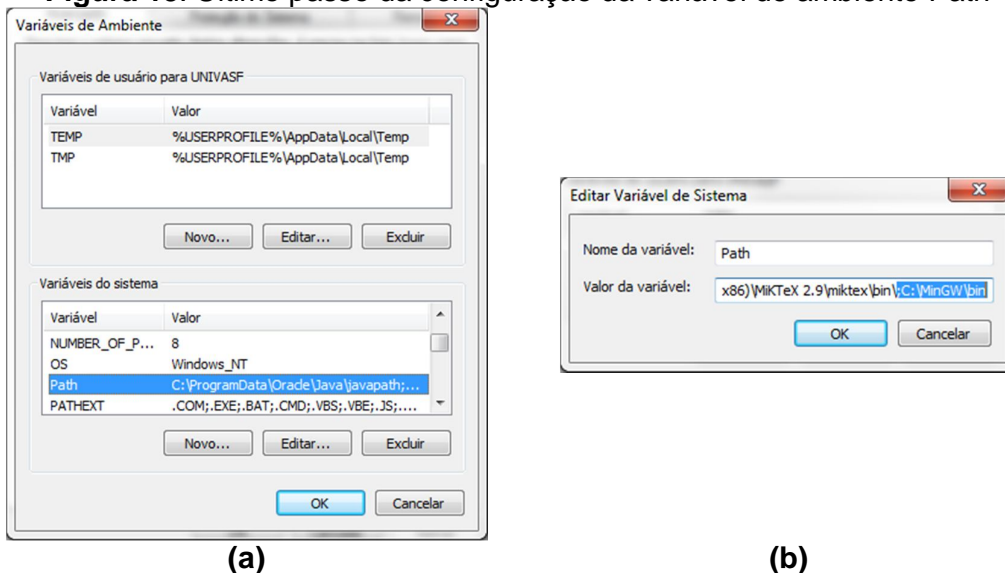
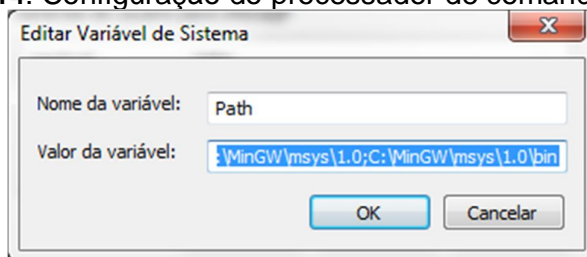


Figura 14: Configuração do processador de comando MSYS



Dica

Se você adotou a sugestão do instalador do MinGW, o caminho do MinGW será: `c:\MinGW\bin\` e os dois caminhos do processador de comando MSYS será: `c:\MinGW\msys\1.0` e `c:\MinGW\msys\1.0\bin\`



Recomendação

Não apague nenhum valor já existente no campo valor da variável, ou seja, só adicione os novos valores.

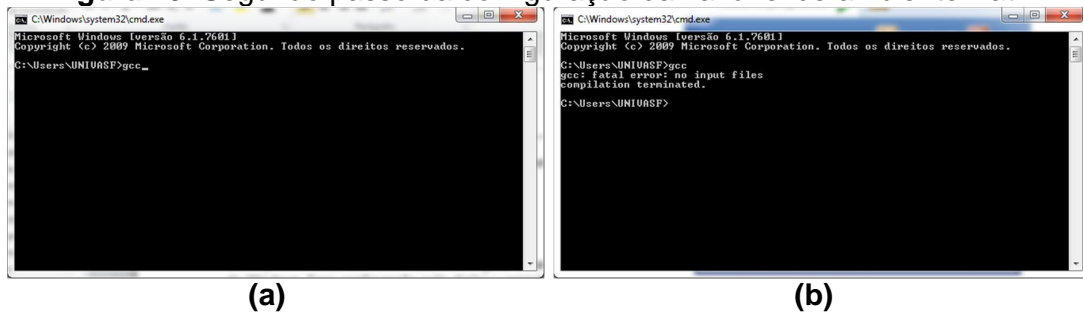
Para finalizar a configuração basta clicar no botão **Ok** de todas as janelas abertas.

2.4.4. Verificar a Instalação

Abra uma janela de comando do Windows, executando o aplicativo **cmd**. Na janela de comando, digite o comando **gcc**, conforme é mostrado na **Figura 15 (a)**.

Se aparecer a mensagem: `gcc: fatal error: no input files`, a instalação e a configuração foram feitas com sucesso, como mostra a **Figura 15 (b)**.

Figura 15: Segundo passo da configuração da variável de ambiente Path



Dica



Caso não apareça, essa mensagem (`gcc: fatal error: no input files`), digite **c:\minggw\bin\gcc.exe** (se adotou a sugestão do instalador) e se a mensagem for exibida, refaça a configuração da variável de ambiente **Path**.

2.5. Exercícios