

# Tutorial de configuração do GLUT utilizando a IDE Code Blocks - Windows 7/ 8 (64 bits)

(Por Tais Rocha Silva)

## PASSO 1: Baixando o freeglut.

- Acesse o endereço <http://freeglut.sourceforge.net> .
- No tópico "Downloads..." e no subtópico "Prepackaged Releases ", clicar no link "[Martin Payne's Windows binaries \(MSVC and MinGW\)](#)":

### Downloads...

Below are file links for the FreeGLUT project. README files are included. Have fun!

#### Testing Releases

The ports to Android and BlackBerry 10 as well as other API and implementation enhancements (e.g., move to CMake build system, VBO and shader support for geometry) are currently only available in trunk. Feel free to test by downloading a [tarball of current trunk](#), or [grabbing a copy from svn](#), and give us feedback on how it worked for you. All this will eventually become a FreeGLUT 3.0 release.

#### Stable Releases

[Freeglut 2.8.1 \[Released: 5 April 2013\]](#)  
[Freeglut 2.8.0 \[Released: 2 January 2012\]](#)  
[Freeglut 2.6.0 \[Released: 27 November 2009\]](#)  
[Freeglut 2.4.0 \[Released: 9 June 2005\]](#)  
[Freeglut 2.2.0 \[Released: 12 December 2003\]](#)  
[Freeglut 2.0.1 \[Released: 23 October 2003\]](#)

#### Prepackaged Releases

The FreeGLUT project does not support packaged versions of FreeGLUT excepting, of course, the tarballs distributed here. However, various members of the community have put time and effort into providing source or binary rollups, and we thank them for their efforts. Here's a list which is likely incomplete:

[Martin Payne's Windows binaries \(MSVC and MinGW\)](#)   
[Florian Echtler's MPX Patch](#)

If you have problems with these packages, please contact their maintainers - we of the FreeGLUT team probably can't help.

#### Development Releases

[SVN trunk tarball](#)  
[Anonymous SVN Instructions](#)

- Como o Code Blocks utiliza-se do MinGW como compilador, devemos escolher o pacote do freeglut específico para esse software. Assim, deve-se clicar no link "Download freeglut 2.8.1-1 for MinGW":

### freeglut 2.8.1 MSVC Package

This package contains 32 and 64 bit Windows DLLs, import libraries, and header files, allowing freeglut applications to be written using Microsoft's Visual C++. The package was built from source code using Visual Studio 2012. The DLL is binary compatible with both my MinGW DLL, and the GLUT for Win32 DLL provided by Nate Robins. The DLL has been tested on Windows 98 SE, Windows ME, Windows 2000, Windows XP, Windows Vista, Windows 7 (64 bit), and Windows 8 (64 bit).

[Download freeglut 2.8.1-1 for MSVC](#)

### freeglut 2.8.1 MinGW Package

This package contains 32 and 64 bit Windows DLLs, import libraries, static libraries, and header files, allowing freeglut applications to be compiled using the MinGW compiler. The package was built from source code using MinGW. The DLL is binary compatible with both my MSVC DLL, and the GLUT for Win32 DLL provided by Nate Robins. The DLL has been tested on Windows 98 SE, Windows ME, Windows 2000, Windows XP, Windows Vista, Windows 7 (64 bit), and Windows 8 (64 bit).

The package should also work with the various IDEs which offer frontends to MinGW / gcc, for example Eclipse, Dev-C++, and Code::Blocks. However, you'll need to consult the manual for your IDE for instructions on how to do this.

[Download freeglut 2.8.1-1 for MinGW](#) 

For further details on using freeglut with MinGW, see my [GLUT MinGW article](#).

### Older freeglut Packages

If you would prefer to download the older 2.4.0 or 2.6.0 packages, they can be found in the [older GLUT packages directory](#). If you need to support Windows NT 4.0, you will need to download one of these packages as freeglut ≥2.8.0 is not compatible with this platform.

- O link acima salva um arquivo compactado de extensão ".mp" em uma pasta de sua preferência. Descompacte o arquivo para seguirmos ao próximo passo.

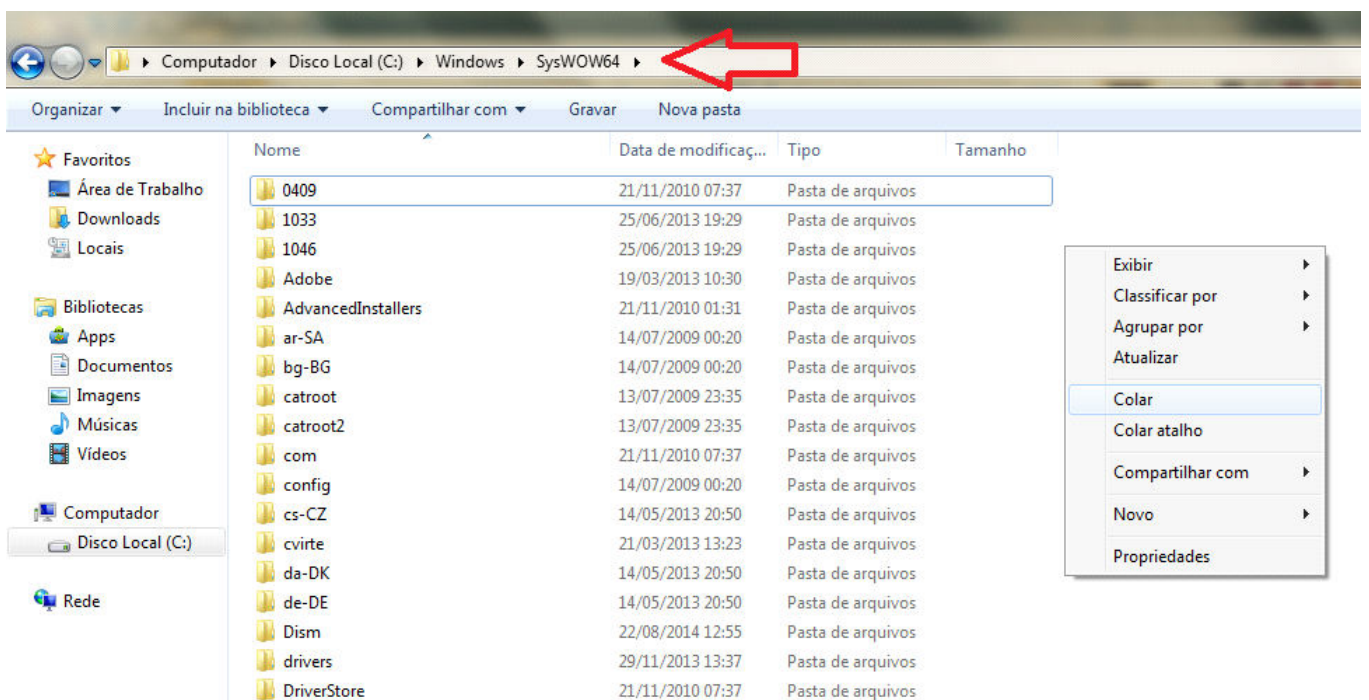
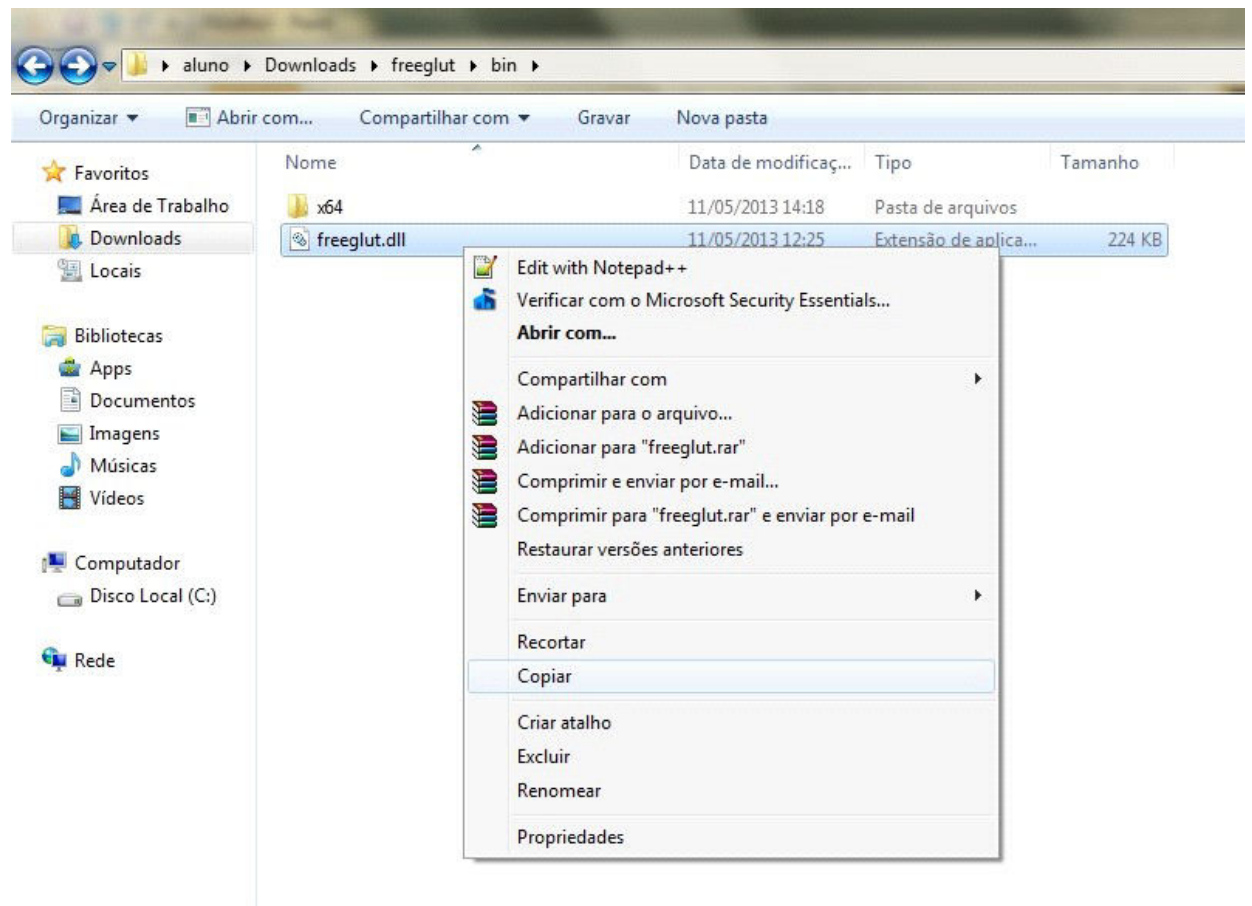
## **PASSO 2: Configurando seu PC para a utilização do GLUT.**

- Dentro do arquivo descompactado há uma pasta de nome fre glut. Por sua vez, no interior dessa pasta existem três outras, extremamente importantes: **bin**, **include** e **lib**. Devemos direcionar o conteúdo dessas pastas às respectivas pastas no Code Blocks e no sistema, para que a existência do GLUT seja reconhecida nos projetos a serem criados.
  - Bin: Aqui existe um arquivo de extensão ".dll", referente a um arquivo do sistema do tipo 32 bits. Existe também uma pasta de nome "x64" que contém um arquivo idêntico ao citado acima, mas do tipo 64bits.

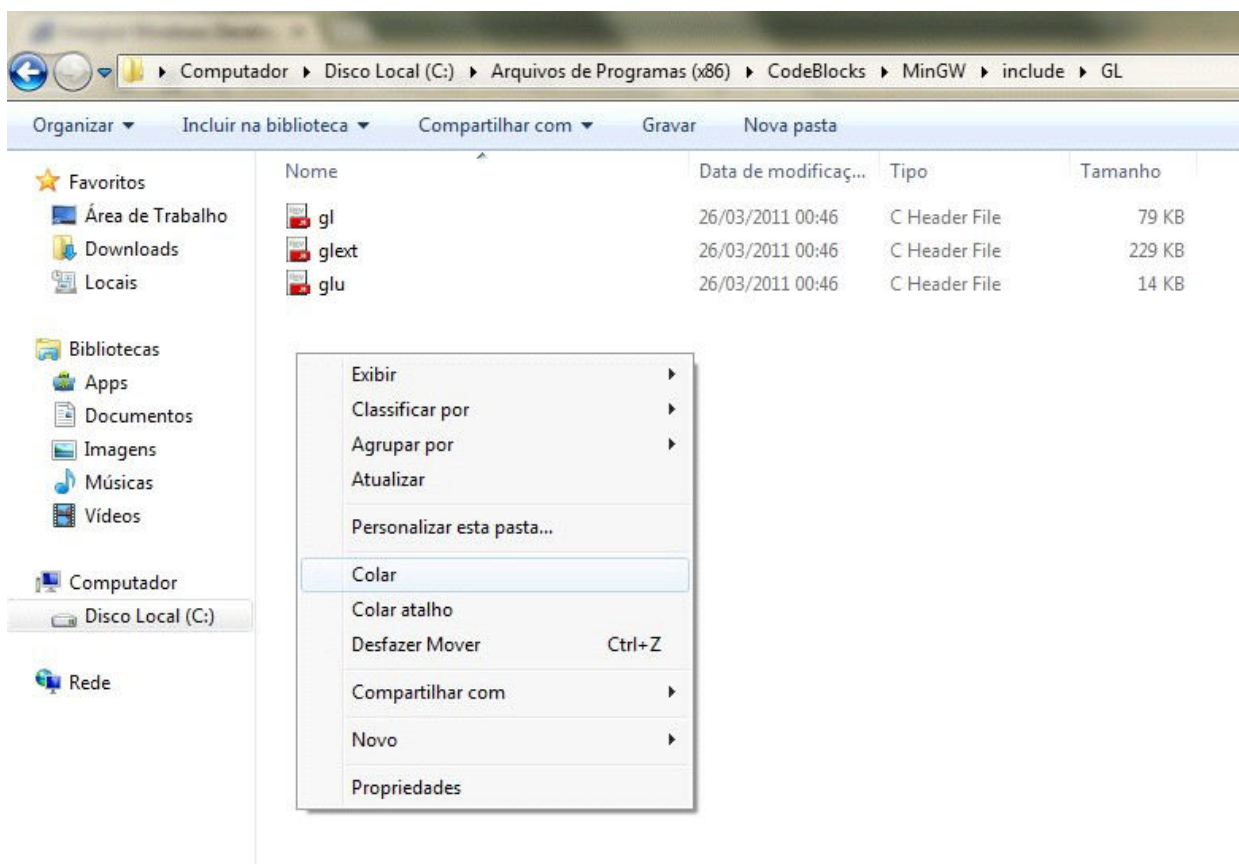
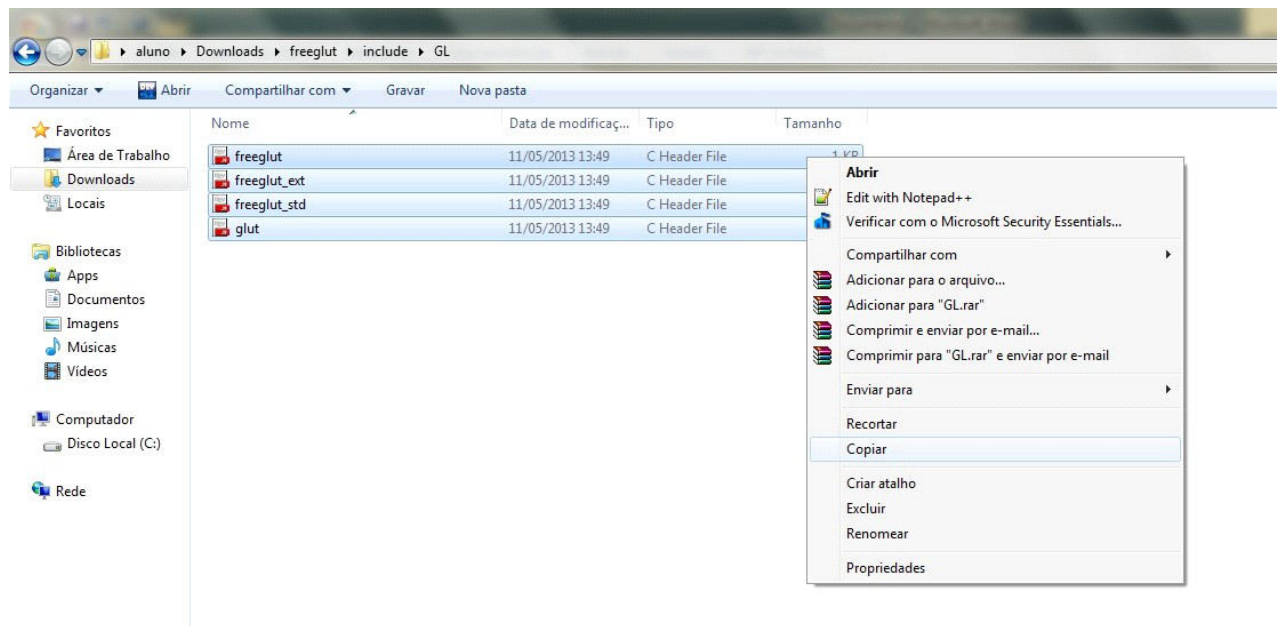
### **ATENÇÃO!**

Este tutorial contém os passos de configuração do GLUT no Code Blocks para um sistema operacional Windows de 64 bits, o que possivelmente nos induz a pensar que, neste momento, o arquivo ".dll" a ser copiado para o sistema seria o que está dentro da pasta "x64", certo?

ERRADO!!! Os usuários curiosos certamente já perceberam que ao ser instalado o Code Blocks no PC é criada uma pasta, com o mesmo nome do programa, no diretório "Disco local (C:)", pasta "Arquivos de Programas (x86)". Isso quer dizer que esse software "pertence a um grupo de programas de 32 bits", apesar do sistema ser de 64. Portanto, para que o sistema associe o GLUT ao CG, devemos copiar o arquivo "fre glut.dll" que está na raiz da pasta bin e, de forma aparentemente contraditória, levar este arquivo para a pasta **SysWOW64**, uma pasta análoga a conhecida System32, porém, para SO's de 64 bits. Ambas as pastas podem ser encontradas na pasta Windows, dentro do diretório "(C:)". Veja as imagens:

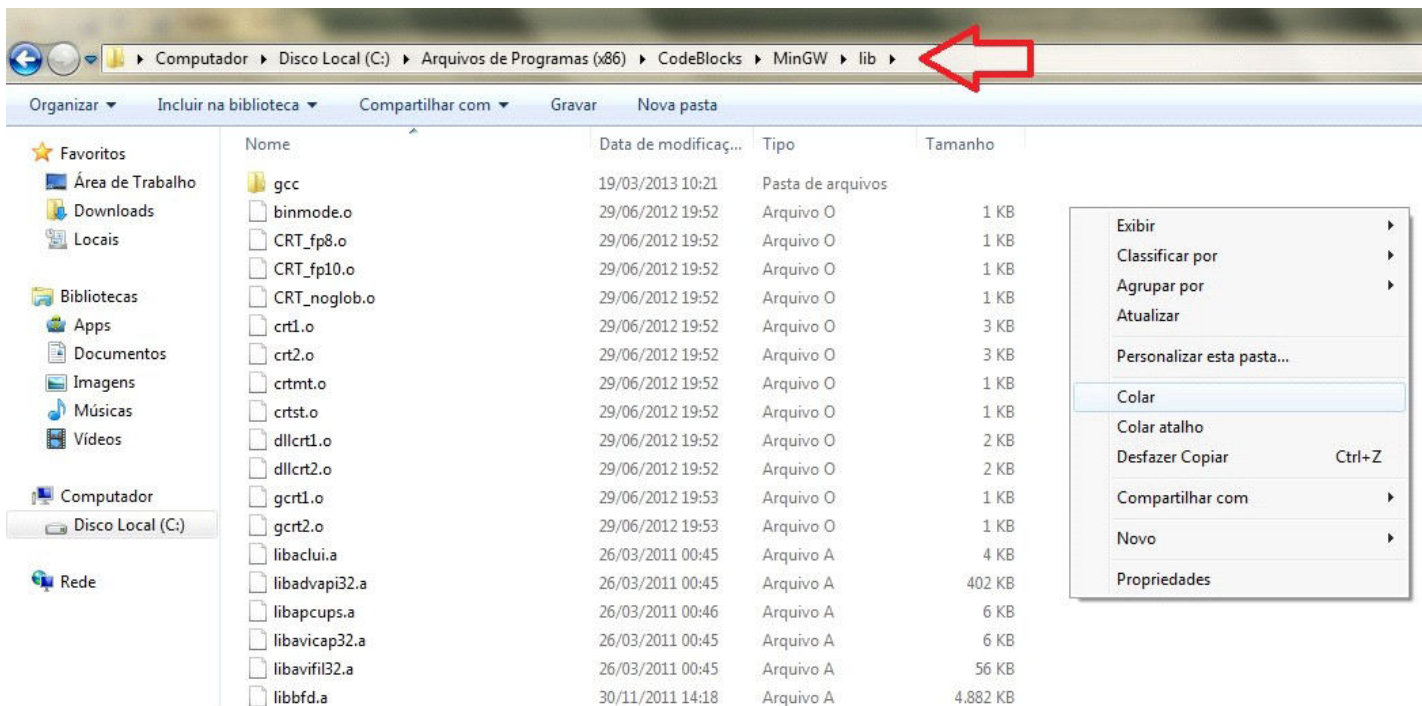
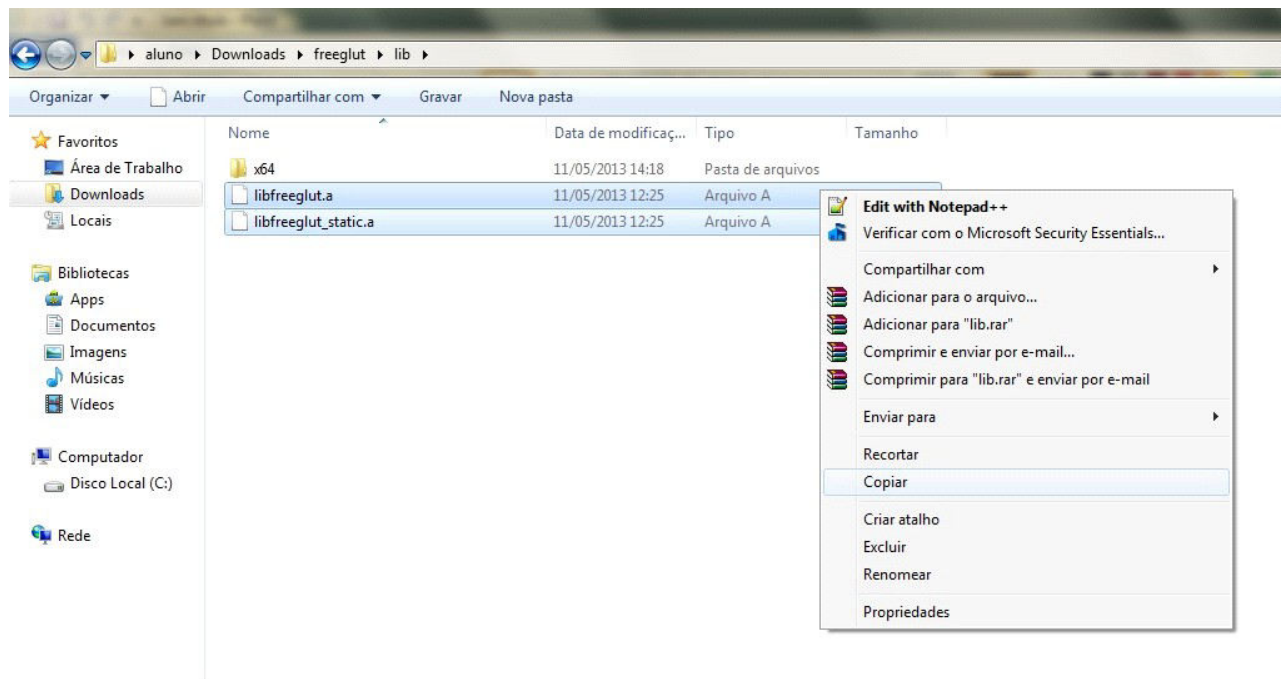


- Include: Contém arquivos de biblioteca com a extensão ".h". Basta copiar o conteúdo desta pasta e colá-lo em uma pasta idêntica a essa, dentro da pasta MinGW, contida na pasta do Code Blocks.



- Lib: ("Library") É a pasta de biblioteca do GLUT, contém dois arquivos de extensão ".a". Analogamente à pasta bin, difere os arquivos, aparentemente idênticos, quanto ao sistema operacional (32 ou 64 bits). Como anteriormente, copiamos os dois arquivos que estão na raiz da pasta lib (32 bits) e levamos a uma pasta de mesmo nome que está na pasta MinGW, dentro da pasta do Code Blocks.

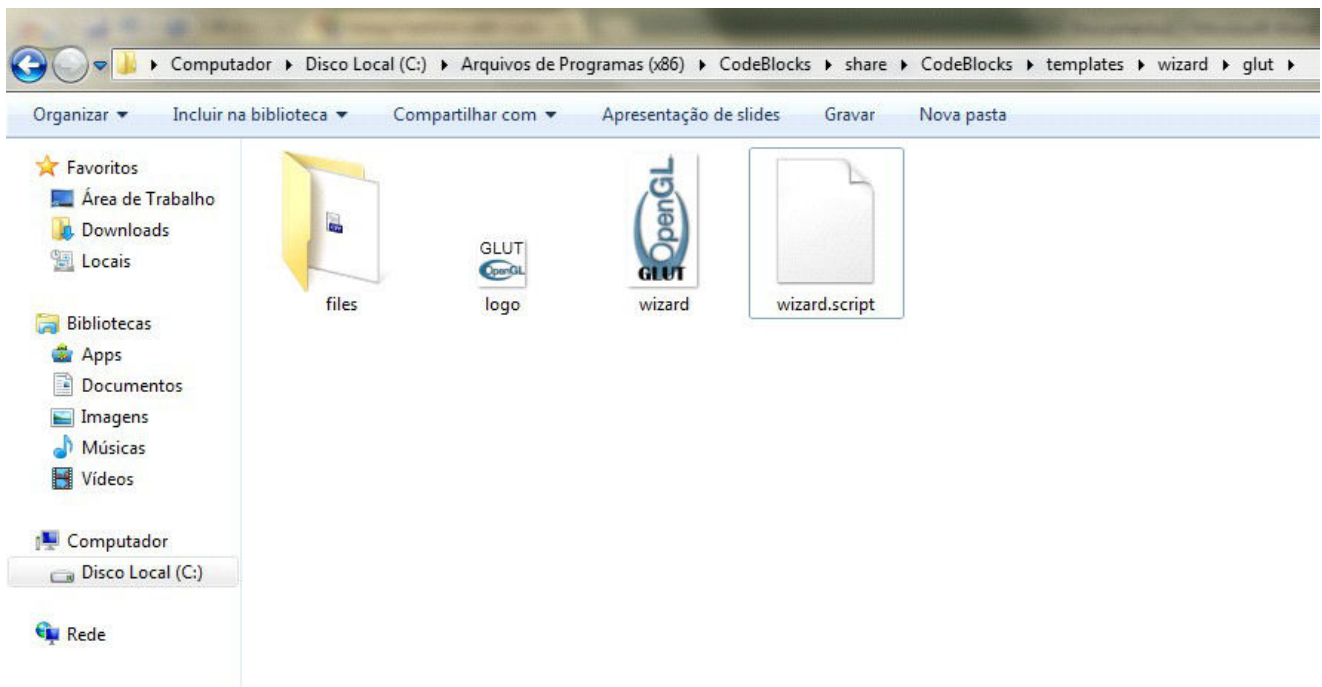




- Depois de tantas transferências de arquivos o processo ainda não chegou ao fim. O Code Blocks está "programado" para reconhecer o arquivo do GLUT denominado glut32, associando essa biblioteca de funcionalidades do OpenGL à nossa IDE (objetivo deste tutorial). Porém, o que temos a oferecer ao mesmo é uma versão mais antiga do GLUT, o freeglut, o que nos leva a ter que alterar essa referência do programa. Para a realização dessa tarefa:

- Devemos ir até a pasta do Code Blocks; entrar na pasta de nome "share"; "Code Blocks" novamente; "templates"; duplo clique na pasta "wizard" e por fim, entrar na pasta "glut". Nesta pasta, clique com o botão direito do mouse sobre o arquivo de nome "wizard.script" e escolha a opção "abrir com...", você deve escolher o bloco de notas ou

o Wordpad para abrir o arquivo.



Aqui devemos fazer duas alterações:

1ª) No trecho do código:

```
if (PLATFORM == PLATFORM_MSW)
{
    if (!VerifyLibFile(dir_nomacro_lib, _T("glut32"), _T("GLUT's"))) return false;
}
```

devemos alterar "glut32" para "freeglut".

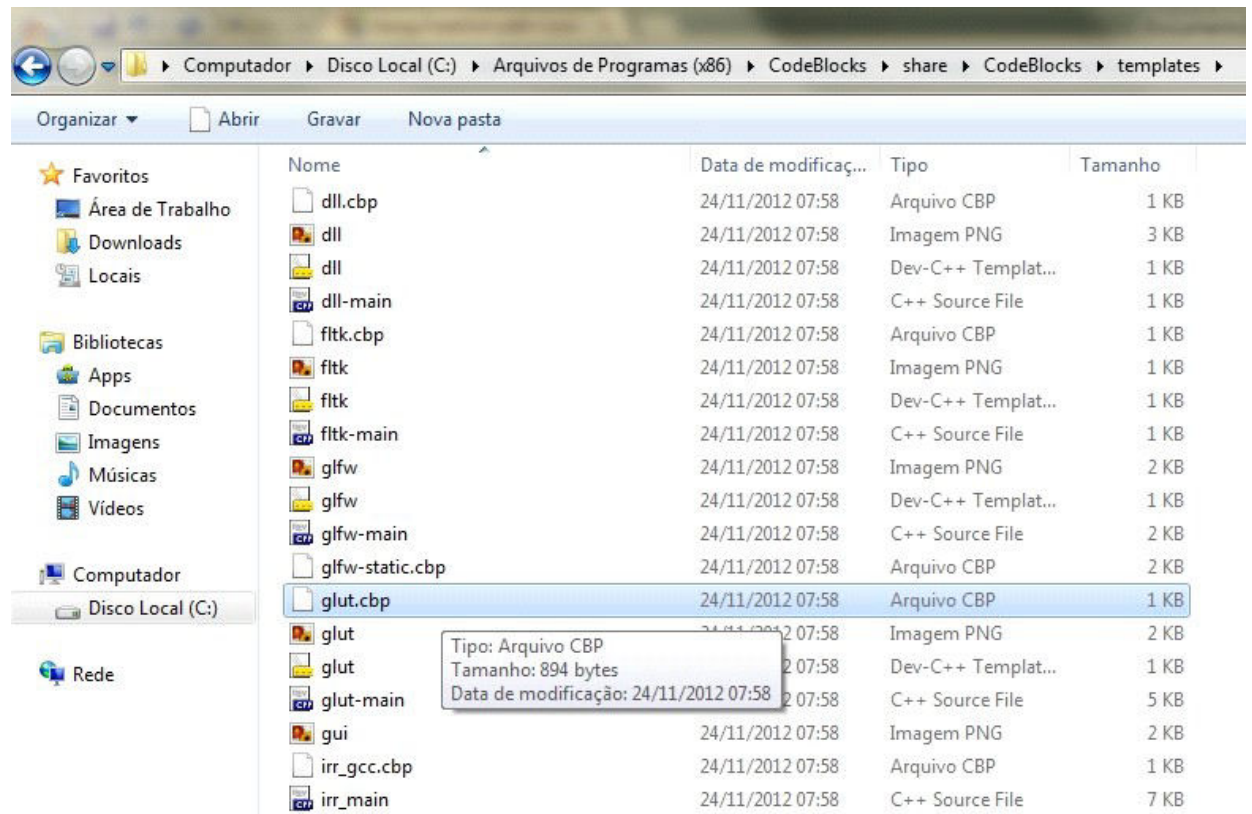
2ª) Assim como no anterior, no trecho:

```
if (PLATFORM == PLATFORM_MSW)
{
    project.AddLinkLib(_T("glut32"));
    project.AddLinkLib(_T("opengl32"));
    project.AddLinkLib(_T("glu32"));
    project.AddLinkLib(_T("winmm"));
    project.AddLinkLib(_T("gdi32"));
}
```

devemos alterar **apenas** onde está "glut32" para "freeglut".

Para ativar as alterações realizadas, salve o arquivo na área de trabalho do computador, pois o sistema não aceita que você salve diretamente na pasta onde o arquivo está (aparecerá uma mensagem caso tente fazer isso), depois arraste o arquivo da área de trabalho para a pasta de origem, substituindo o anterior.

2. Agora devemos realizar o mesmo procedimento em um arquivo de nome "glut.cbp", localizado na pasta "templates" (C:\Program Files (x86)\CodeBlocks\share\CodeBlocks\templates).



Este arquivo pode ser aberto da mesma forma que o anterior, por bloco de notas ou wordpad (ou afins).

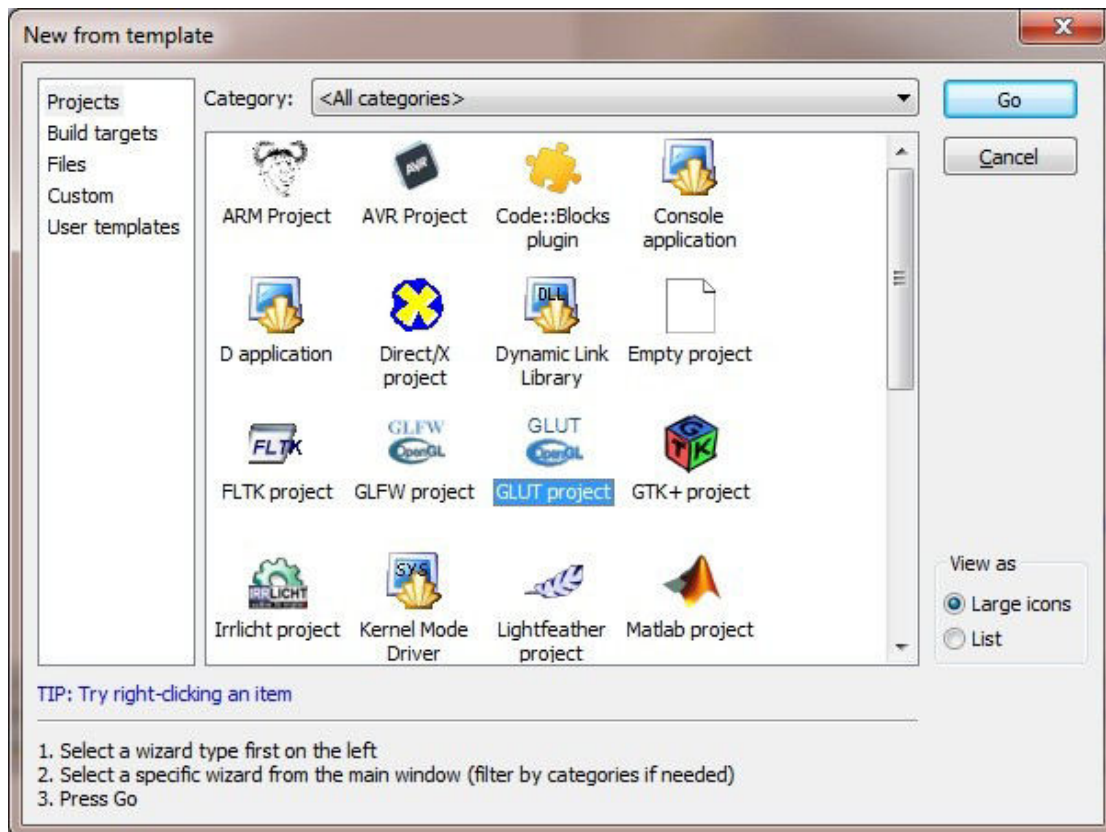
Aqui há apenas uma alteração a ser realizada:

No trecho de código `<Add library="glut32" />`, altere "glut32" para "freeglut". Para salvar as alterações, siga o mesmo procedimento anterior (salve o arquivo na área de trabalho, depois mova-o de volta para a pasta de origem, "templates", substituindo o arquivo anterior).

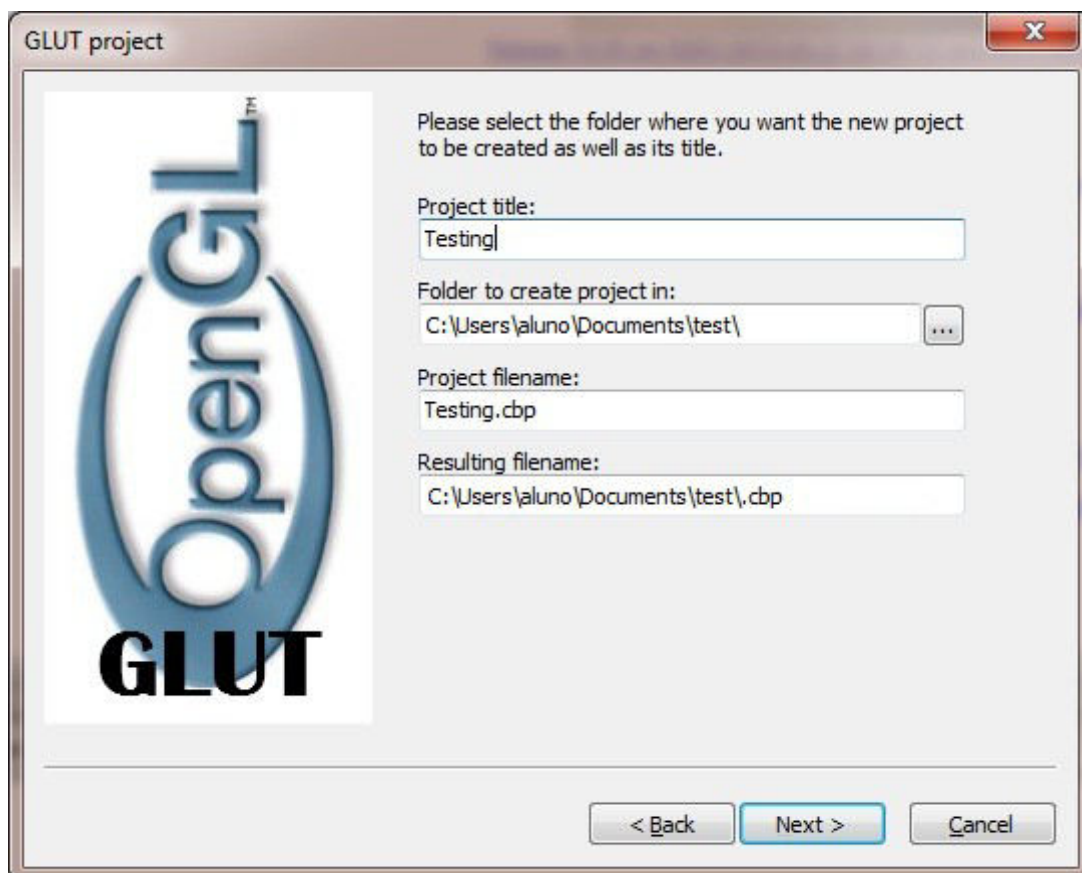
PRONTO! Agora podemos criar o nosso projeto!

### PASSO 3: Criando um projeto GLUT.

- Abra o Code Blocks.
- Na barra superior clique em "File", em seguida clique em "New", e depois em "Project...".
- Aberta a nova janela, busque pelo ícone "GLUT project" e depois clique em "Go".

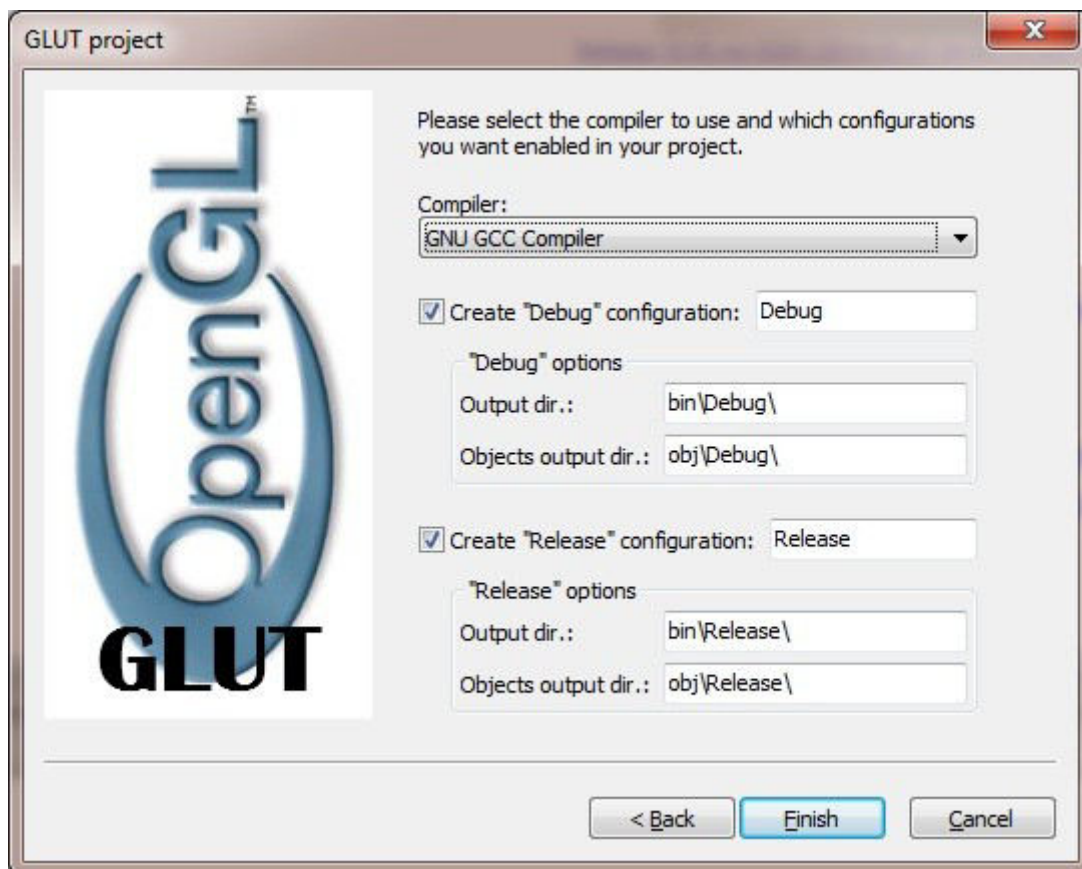


- Em seguida apresente um nome para o projeto e especifique o local onde ele será salvo em seu computador. Clique em "Next".

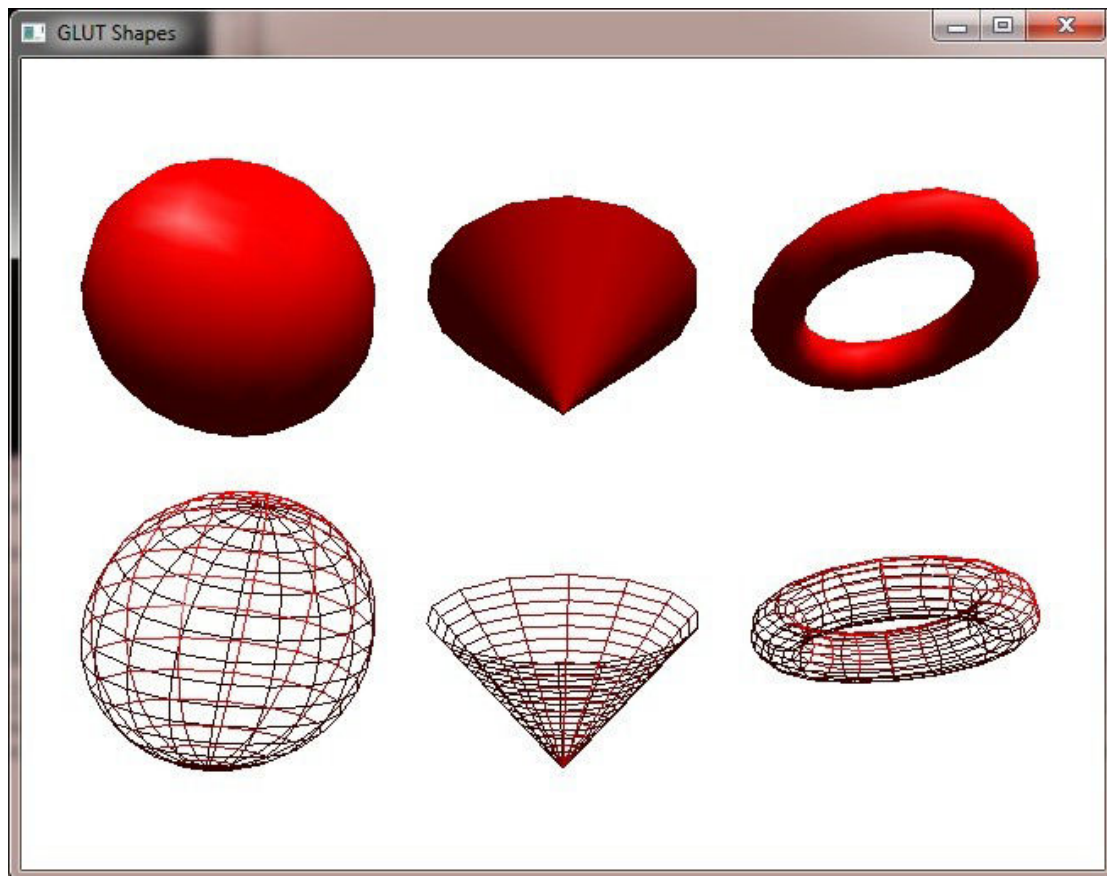


- Busque o caminho onde está a pasta do MinGW. Clique em "Next" e na janela seguinte clique em "Finish".





- Por último, pressione a tecla F9 ou clique em "Build and run" no Code Blocks.



Caso a imagem acima tenha surgido na tela do seu computador, PARABÉNS, a biblioteca GLUT está devidamente configurada na IDE Code Blocks do seu sistema operacional. Bom trabalho!