

Tutorial

Como usar o Glut em C no CodeBlocks para Windows 7 (64 Bits)

Lauro César Jacques Santos
Computação Gráfica
Engenharia da Computação – CEFET-MG

Baseado na experiência pessoal e no tutorial disponível em
<http://www.sci.brooklyn.cuny.edu/~goetz/codeblocks/glut/>

Tutorial: Como usar o Glut em C no CodeBlocks para Windows 7

1º passo: Instalação do Codeblocks.

Baixe e instale o CodeBlocks (<http://www.codeblocks.org/downloads>).

2º passo: Conteúdo do Glut.

Baixe o GLUT bin file disponibilizado por Nate Robins através do endereço <http://user.xmission.com/~nate/glut.html>. É o primeiro link de download.

Extraia os arquivos com alguma ferramenta de compressão/descompressão de arquivos. Então copie os seguintes arquivos para os respectivos endereços:

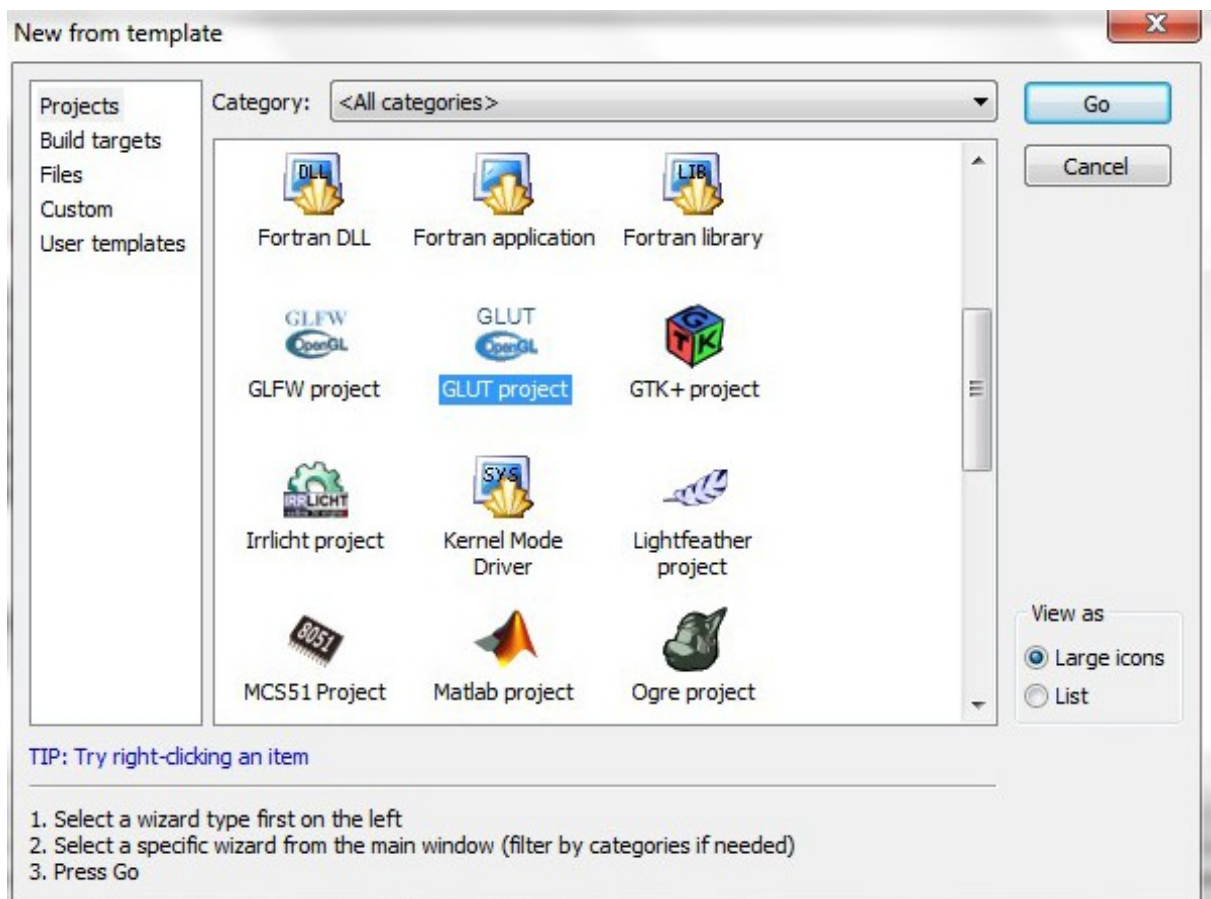
- glut32.lib para C:\Arquivos de Programas(x86)\CodeBlocks\MinGW\lib;
- glut.h para C:\Arquivos de Programas(x86)\CodeBlocks\MinGW\include\GL.
- glut32.dll para C:\Windows\System32;

Os endereços podem variar de acordo com o usuário. Se for caso, é necessário encontrar o diretório correspondente. Pode ser necessário fornecer permissão de administrador.

3º passo: Criando o projeto no CodeBlocks.

Clique em File > New > Project (Arquivo > Novo > Projeto) na barra de ferramentas.

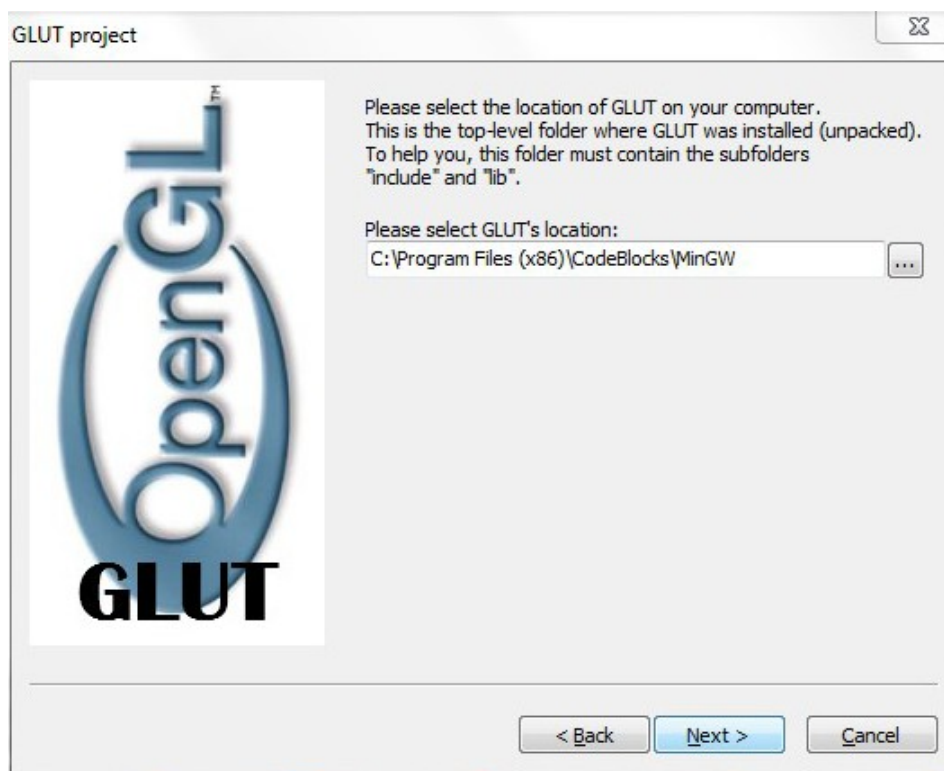
Selecione a opção GLUT Project e avance.



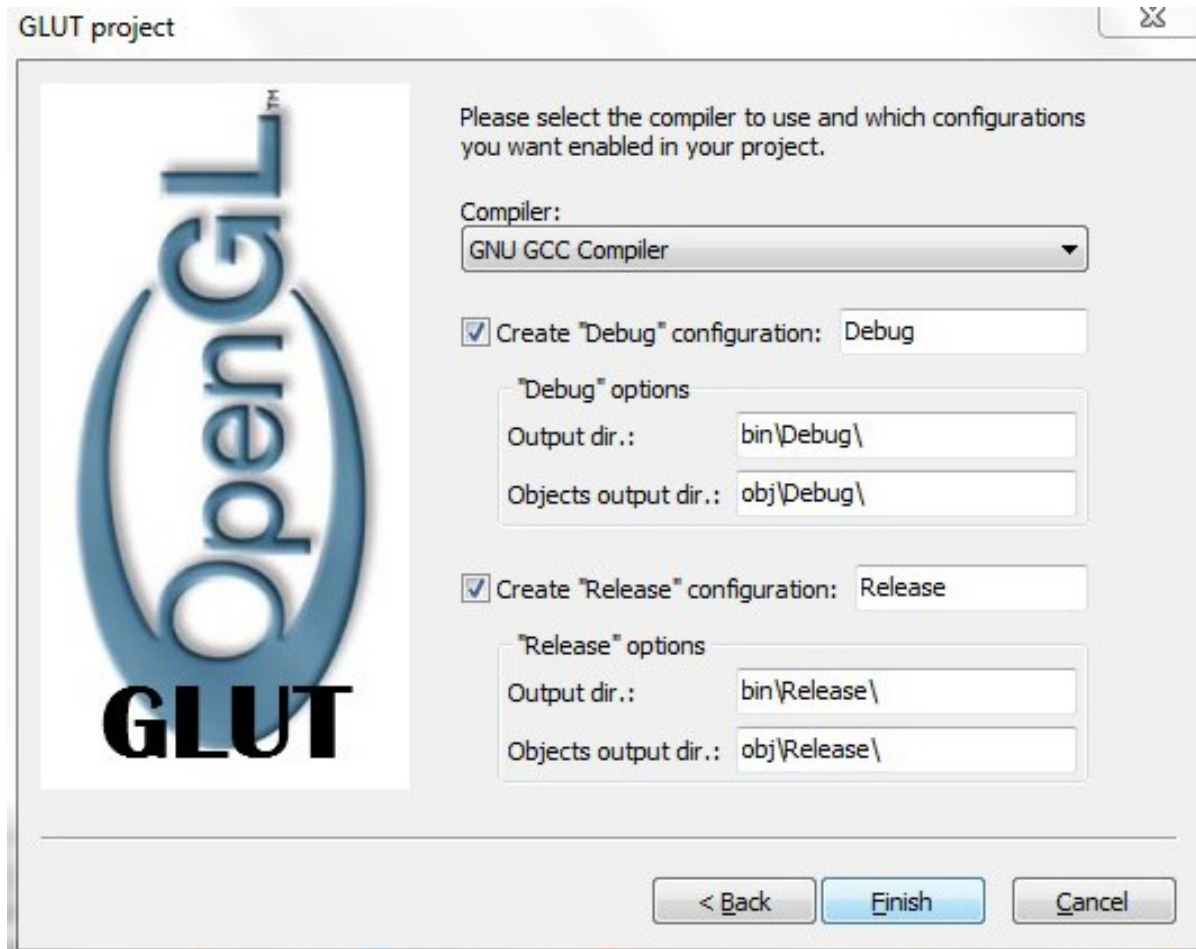
Escolha o nome e o caminho do projeto e clique em Next.



Selecione então a pasta na qual você colocou os arquivos do Glut (C:\Program Files (x86)\CodeBlocks\MinGW por padrão do tutorial). Se o programa abrir outras janelas, feche-as.

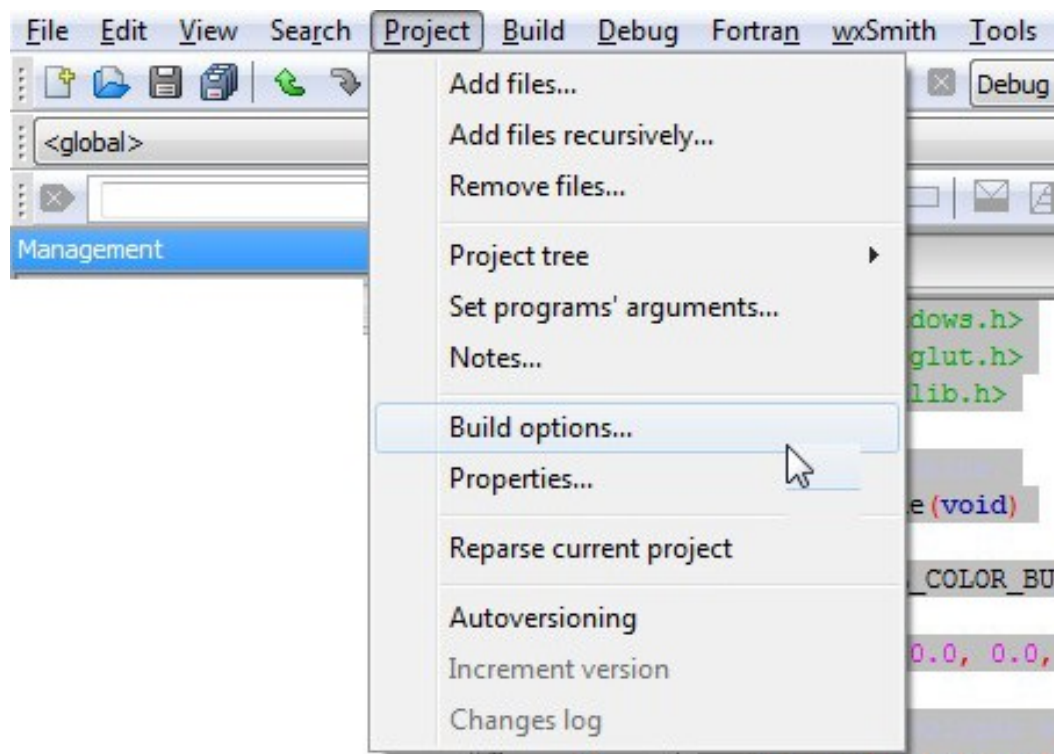


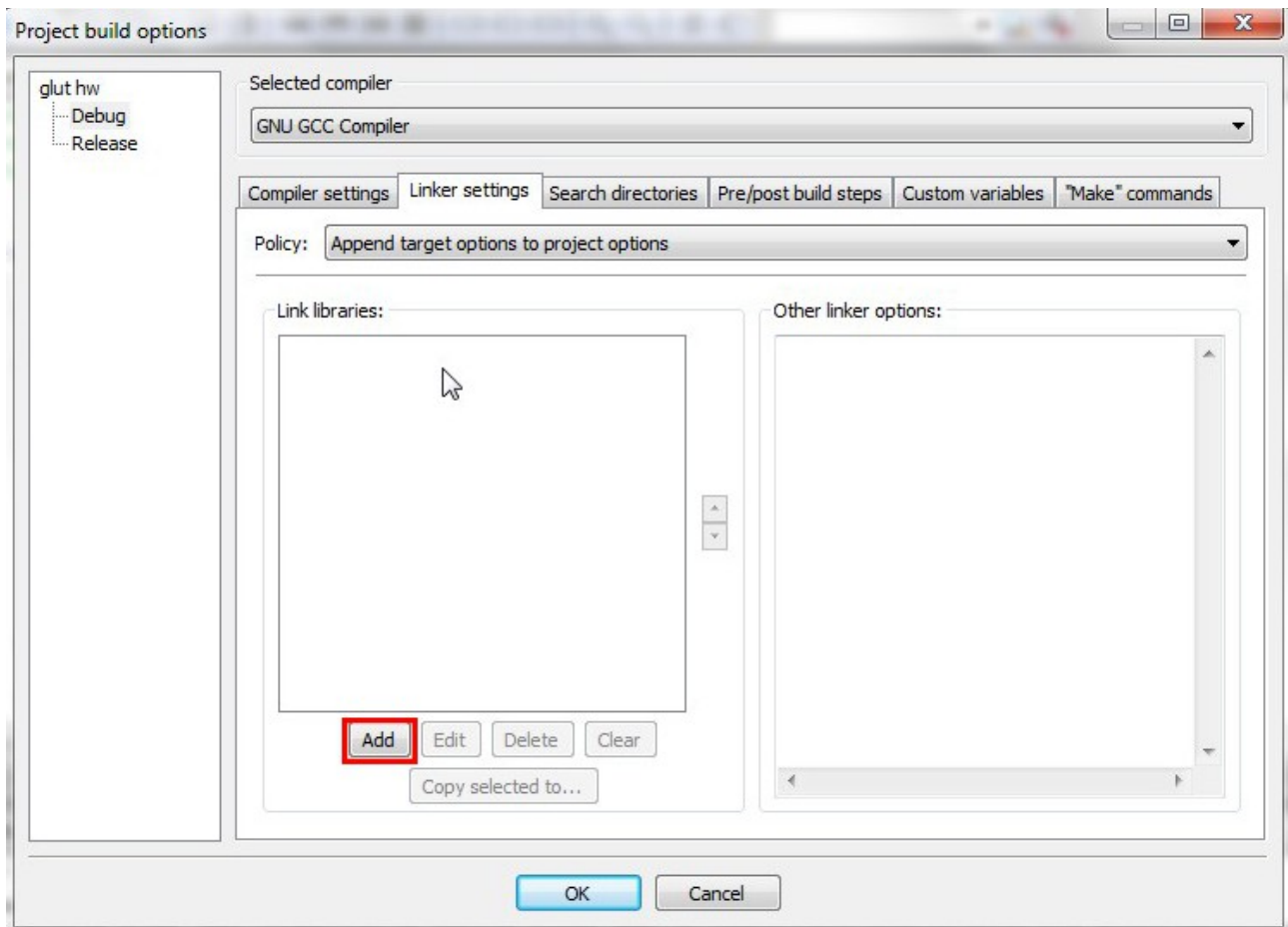
Avance e termine o projeto.



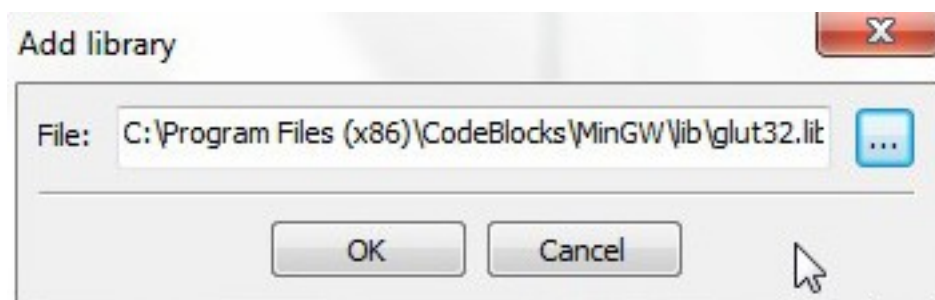
4º passo: Vinculando a biblioteca

Vá em Project > Build Options. Clique na aba Linker Settings e pressione o botão "Add".





Insira o endereço do glut32.lib, que deve ser C:\Arquivos de Programas(x86)\CodeBlocks\MinGW\lib por padrão. Se o programa perguntar se você deseja fazer desse endereço um endereço relativo, clique em não. Dê Ok e dê ok na janela do Project build tools também.



5º passo: Criando o arquivo.

Crie um novo arquivo em branco, você deve dar um nome a ele e salvar com a extensão ".c". Dê ok em qualquer outra janela que abrir.

Remova o arquivo main.cpp do projeto.

Pronto. O ambiente está pronto para desenvolver os seus projetos em OpenGL através do Glut. Não se esqueça de incluir as bibliotecas através do

```
#include <windows.h>
#include <GL/glut.h>
#include <stdlib.h>
```

Passo complementar: código para teste.

```
#include <windows.h>
#include <GL/glut.h>
#include <stdlib.h>

// Drawing routine.
void drawScene(void)
{
    glClear(GL_COLOR_BUFFER_BIT);

    glColor3f(0.0, 0.0, 0.0);

    // Draw a polygon with specified vertices.
    glBegin(GL_POLYGON);
        glVertex3f(20.0, 20.0, 0.0);
        glColor3f(0.7, 0.5, 0.3);
        glVertex3f(80.0, 20.0, 0.0);
        glVertex3f(80.0, 80.0, 0.0);
        glColor3f(0.5, 0.3, 0.2);
        glVertex3f(20.0, 80.0, 0.0);
    glEnd();

    glFlush();
}

// Initialization routine.
void setup(void)
{
    glClearColor(1.0, 1.0, 1.0, 0.0);
    //glPolygonMode( GL_FRONT_AND_BACK, GL_LINE );
}

// OpenGL window reshape routine.
void resize(int w, int h)
{
    glViewport(0, 0, w, h);
    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();
    glOrtho(-100.0, 100.0, -100.0, 100.0, -5.0, 5.0);
    glMatrixMode(GL_MODELVIEW);
    glLoadIdentity();
}

// Keyboard input processing routine.
void keyInput(unsigned char key, int x, int y)
{
    switch(key)
    {
        case 27:
```

```

        exit(0);
        break;
    default:
        break;
    }
}

// Main routine.
int main(int argc, char **argv)
{
    glutInit(&argc, argv);

    glutInitDisplayMode(GLUT_SINGLE | GLUT_RGBA);
    glutInitWindowSize(500, 500);
    glutInitWindowPosition(100, 100);
    glutCreateWindow("square.cpp");
    glutDisplayFunc(drawScene);
    glutReshapeFunc(resize);
    glutKeyboardFunc(keyInput);

    setup();

    glutMainLoop();
    return 0;
}

```

O resultado deve ser semelhante a este:

