

HƯỚNG DẪN CÀI ĐẶT OPENGL, FREEGLUT, GLEW TRÊN MS VISUAL C++

A- Chuẩn bị các file thư viện Glew, FreeGlut:

Lưu ý: Nếu có các file cài đặt cùng gói file hướng dẫn thì không cần thực hiện các bước sau để download.

1. Glew: <http://glew.sourceforge.net/>

Download file đã biên dịch:



2. FreeGlut:

a) <https://www.transmissionzero.co.uk/software/freeglut-devel/>

freeglut 3.0.0 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 2013. 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 3.0.0 for MSVC \(with PGP signature and PGP key\)](#)

Download thêm:

Get the Sample Code

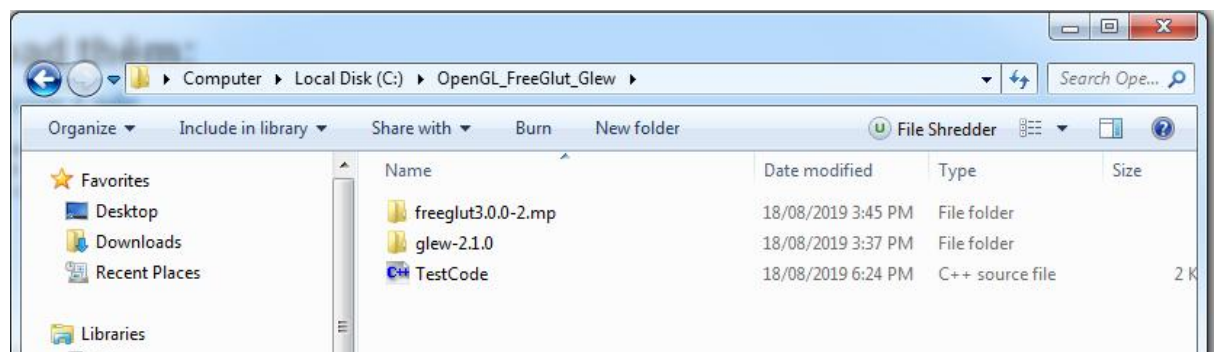
If you're downloading one of the packages on this page, you'll probably want some sample code to try it out. If you have Git installed, you can get the sample code for a simple "Hello World" GLUT application by running "git clone https://github.com/TransmissionZero/Hello-GLUT.git". Alternatively you can [download a Hello GLUT source release](#). You are also encouraged to visit that link to star and watch the repository if you find it useful.

b) Hoặc: <http://freeglut.sourceforge.net/index.php#download>

B- Tạo thư mục chứa các file của FreeGlut và Glew

- Tạo thư mục chứa các file giải nén của FreeGlut và Glew đã download về.

Ví dụ:

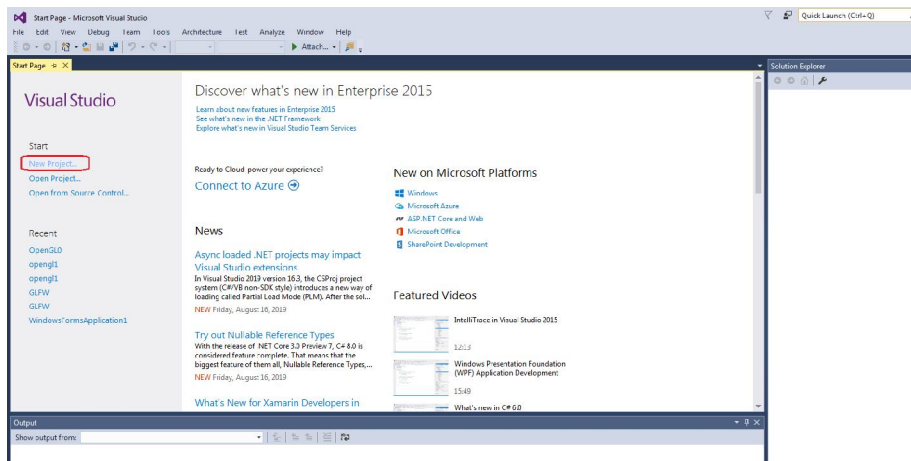


- Copy 2 file `glew32.lib`, `glew32s.lib` tương ứng với máy 32 bit hoặc 64 bit từ thư mục
... <Glew>\Lib\Release ra thư mục ... <Glew>\lib.

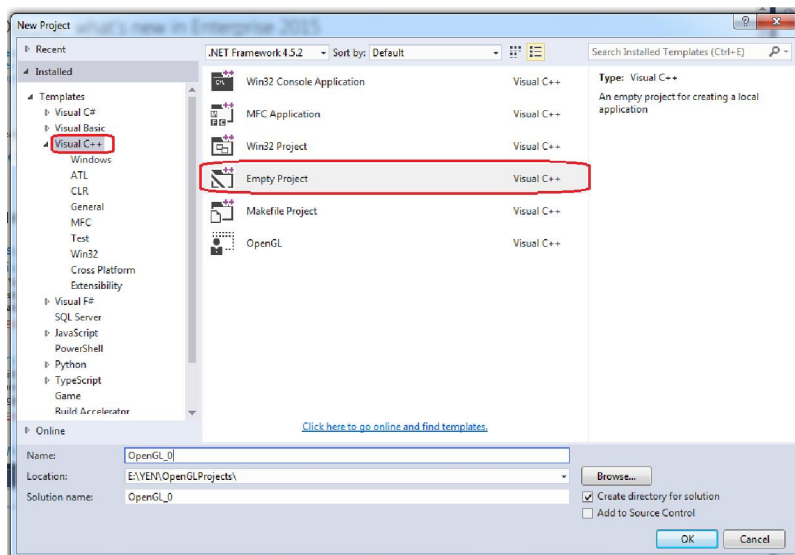
Lưu ý: Nếu là các file thư viện đi kèm với file hướng dẫn này chỉ việc tải về và đặt vào vị trí mong muốn, thông thường đặt tại C:\

C- Thiết lập OpenGL, FreeGlut, Glew trong MS Visual Studio C++

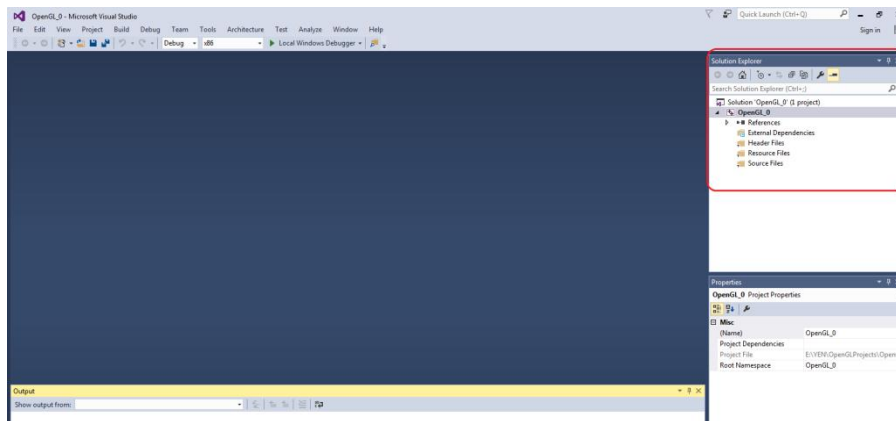
1. Mở một Project mới



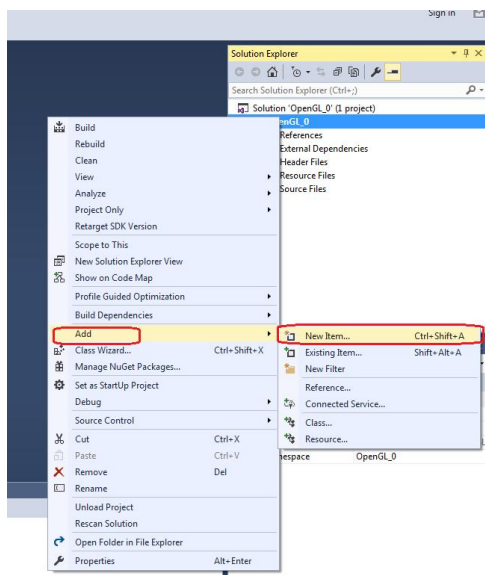
2. Chọn Visual C++ > Empty Project > Đặt tên Project > Chọn OK



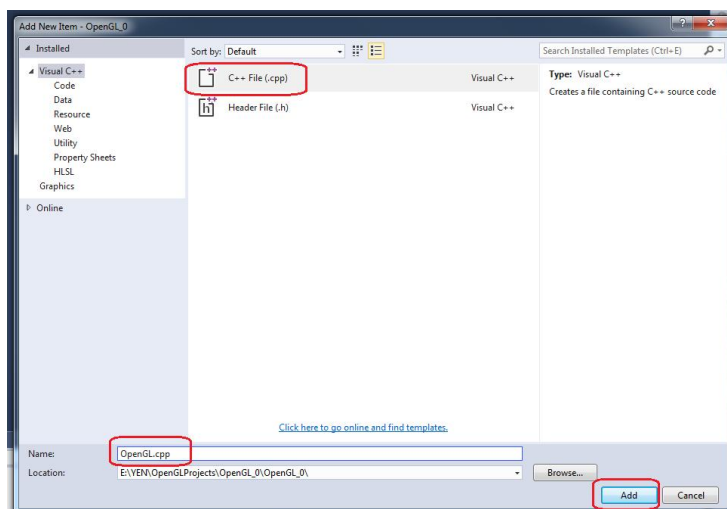
3. Hiện thị Solution Explorer: *View > Solution Explorer*



4. Click chuột phải vào tên của Project (OpenGL_0) trong *Solution Explorer* chọn *Add > New Item*

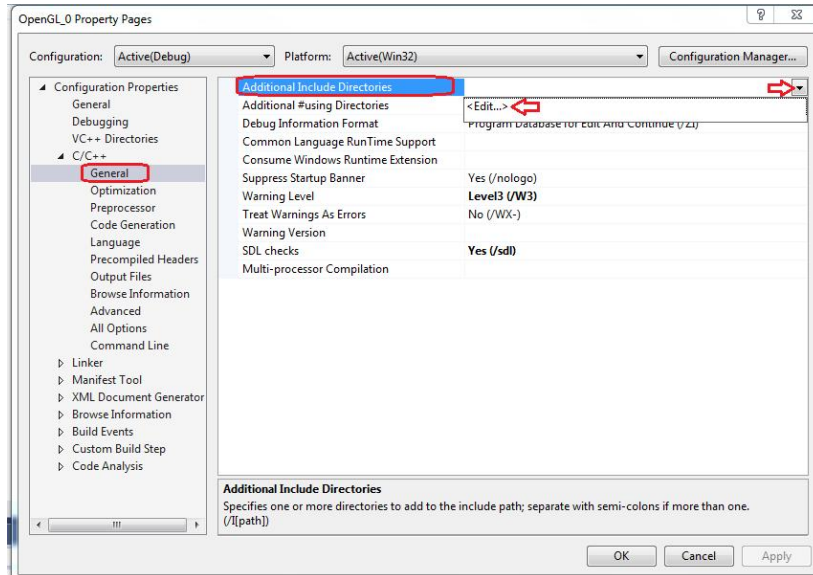


5. Trong cửa sổ *Add New Item*: Chọn *C++ File (.cpp)* > Gõ tên file > Chọn *Add*



6. Thêm các đường dẫn đến các thư mục Include của FreeGlut và Glew

a) Click chuột phải vào tên Project (VD: OpenGL_0) trong *Solution Explorer* > Chọn *Properties* > Xuất hiện cửa sổ *Property Pages* > Chọn *General* > *Additional Include Directories*: click nút đổ xuống combo box > click *Edit*

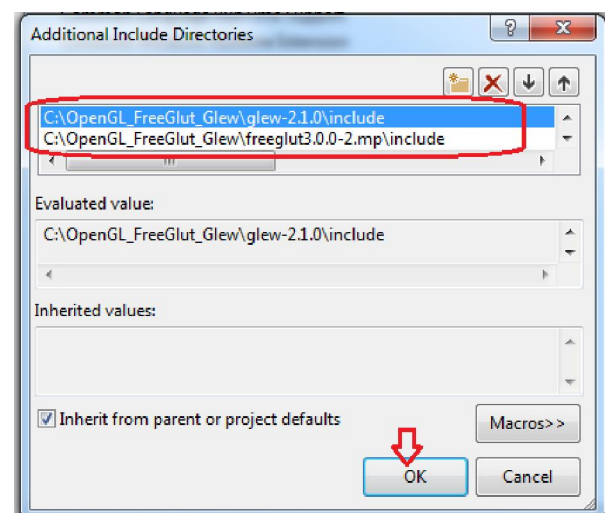
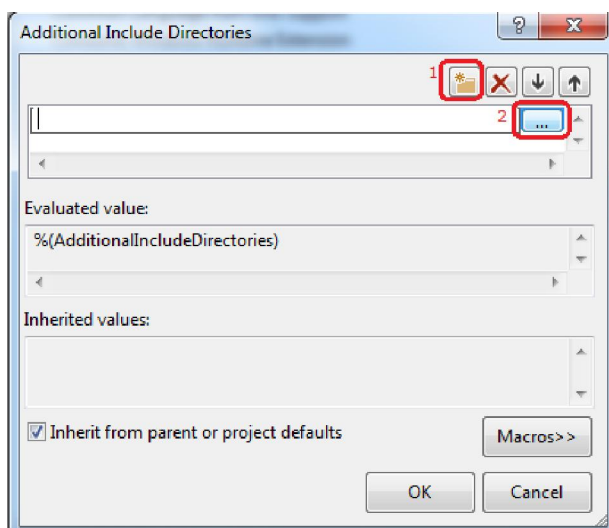


b) Thêm các đường dẫn đến các thư mục Include

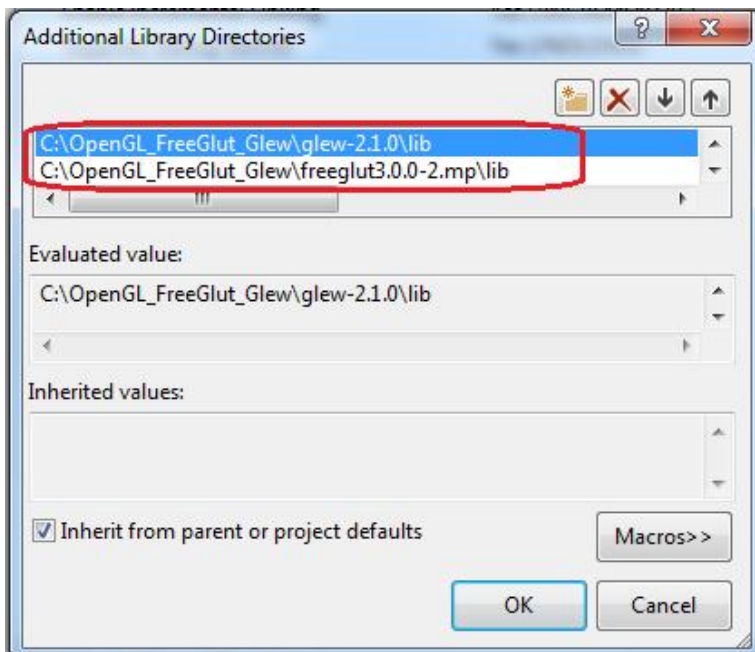
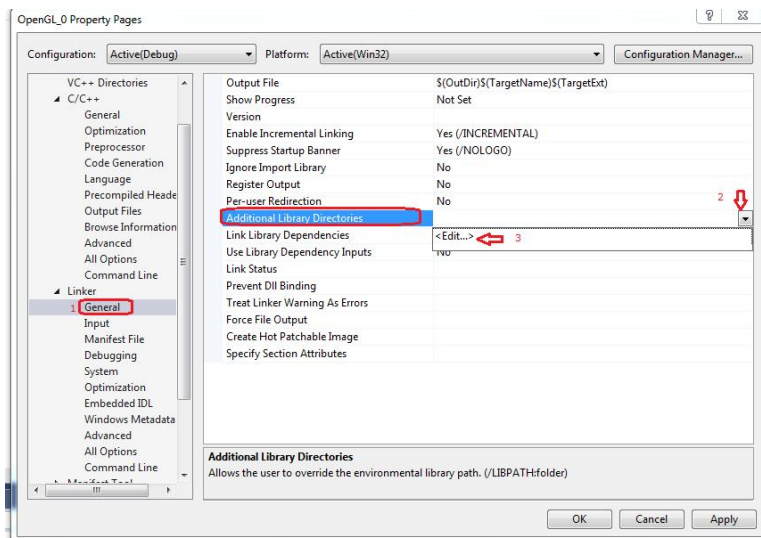
Xuất hiện cửa sổ *Additional Include Directories* thực hiện:

- Click nút *New Line* > Click nút “...” như hình dưới > Chọn đến thư mục Include của *FreeGlut*
- Thực hiện tương tự cho thư mục Include của *Glew*

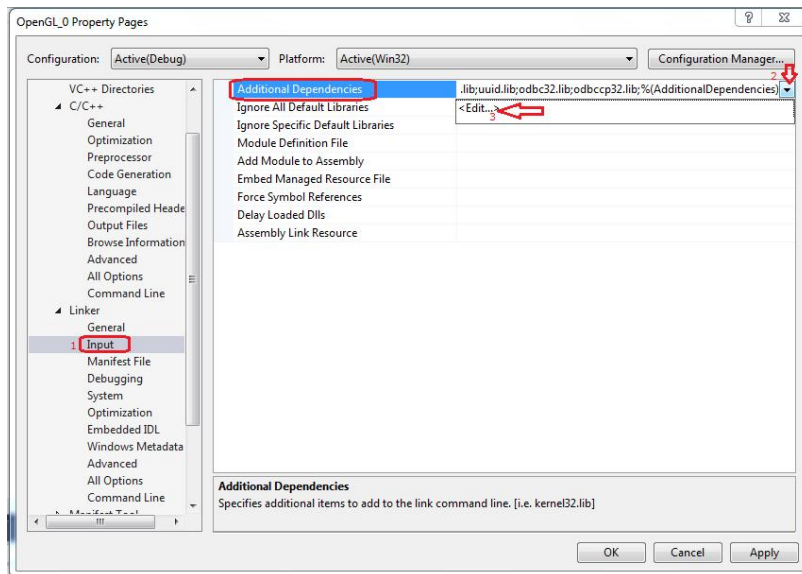
Kết quả như hình:



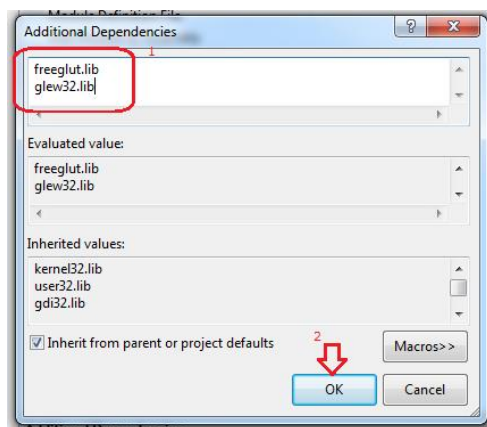
7. Tương tự như các thư mục Include, với các thư mục Lib: Tiếp tục chọn *Linker > General > Additional Library Directories >*



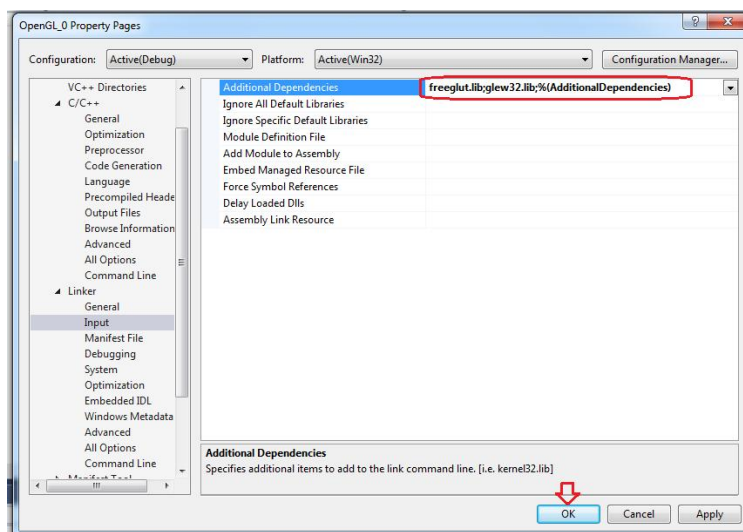
8. Linker > Input > Additional Dependencies > ...



Gõ tên 2 file thư viện: freeglut.lib, glew32.lib > chọn OK



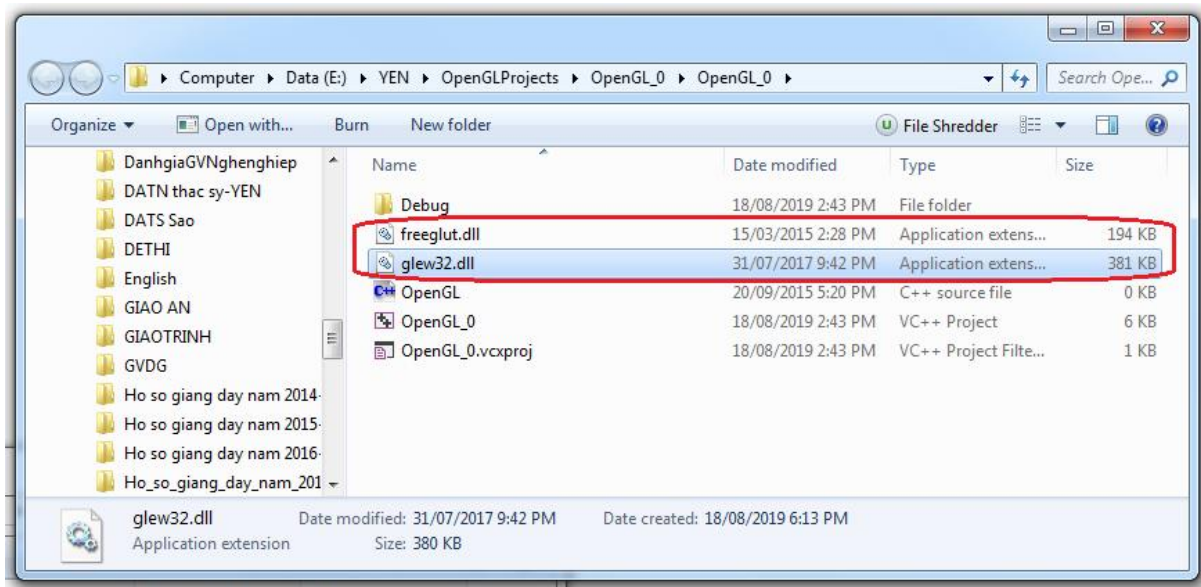
Chọn OK



9. Thêm các file freeglut.dll, glew32.dll vào project

Click chuột phải vào tên project trong Solution Explorer > chọn Open Folder in File Explorer

Mở thư mục lưu trữ các file của FreeGlut và Glew > tìm các file *.dll tương ứng trong thư mục bin > Copy các file freeglut.dll, glew32.dll vào thư mục của Project như hình dưới.



10. Copy mã nguồn Testcode.cpp vào file *.cpp của project (OpenGL.cpp) và chạy chương trình:

```
#include <GL\glew.h>
#include <GL\freeglut.h>
#include <iostream>

using namespace std;

void changeViewPort(int w, int h)
{
    glViewport(0, 0, w, h);
}

void render()
{
    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);

    glBegin (GL_QUADS);
        glVertex2f(-0.5, -0.5);
        glVertex2f(-0.5, 0.5);
        glVertex2f(0.5, 0.5);
        glVertex2f(0.5, -0.5);
    glEnd();
    glutSwapBuffers();
}

int main(int argc, char* argv[]) {
    glutInit(&argc, argv); // Initialize GLUT
    glutInitDisplayMode(GLUT_DOUBLE | GLUT_RGBA | GLUT_DEPTH); // Set up some memory buffers for our display
    glutInitWindowSize(700, 700); // Set the window size
    glutCreateWindow("OpenGL example"); // Create the window with the title "OpenGL example"
```

```

glutReshapeFunc(changeViewport); // Bind the two functions (above) to respond when necessary
glutDisplayFunc(render); // Very important! This initializes the entry points in the OpenGL driver so we
                             can call all the functions in the API.

GLenum err = glewInit();
if (GLEW_OK != err) {
    cout << stderr << " GLEW error";
    return 1;
}

glutMainLoop();
return 0;
}

```

Kết quả:

