

# Tutorial de setup do OpenGL para Mac

CEFET-MG


Engenharia de Computação

Computação Gráfica

Aluno: Luis Carlos Cardoso Filho

\* Para este tutorial, utilizou-se o OS 10.9 Mavericks com o Xcode 6, mas a única mudança provável em relação a sistemas anteriores é a interface do Xcode.

# Baixe o aplicativo Xcode da App Store



## Xcode

Create great apps  
for Mac, iPhone, and iPad.

Installed ▾

**Xcode** 4+

Essentials

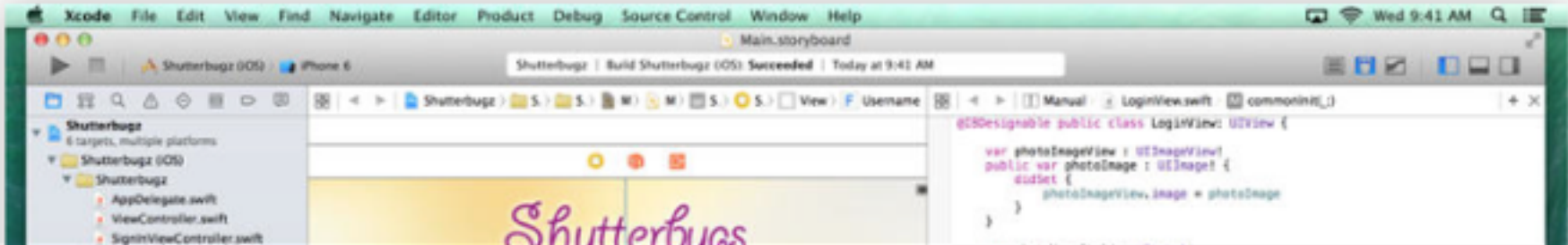
Xcode provides everything developers need to create great applications for Mac, iPhone, and iPad. Xcode brings user interface design, coding, testing, and debugging all into a unified workflow. The Xcode IDE combined with the Cocoa and Cocoa Touch frameworks, and the Swift programming language make developing apps easier and more fun than ever before.

...

**What's New in Version 6.0.1**

Includes SDKs for OS X 10.9 Mavericks and iOS 8.0.

...



Apple Web Site ➤

Xcode Support ➤

App License Agreement ➤

Privacy Policy ➤

...More

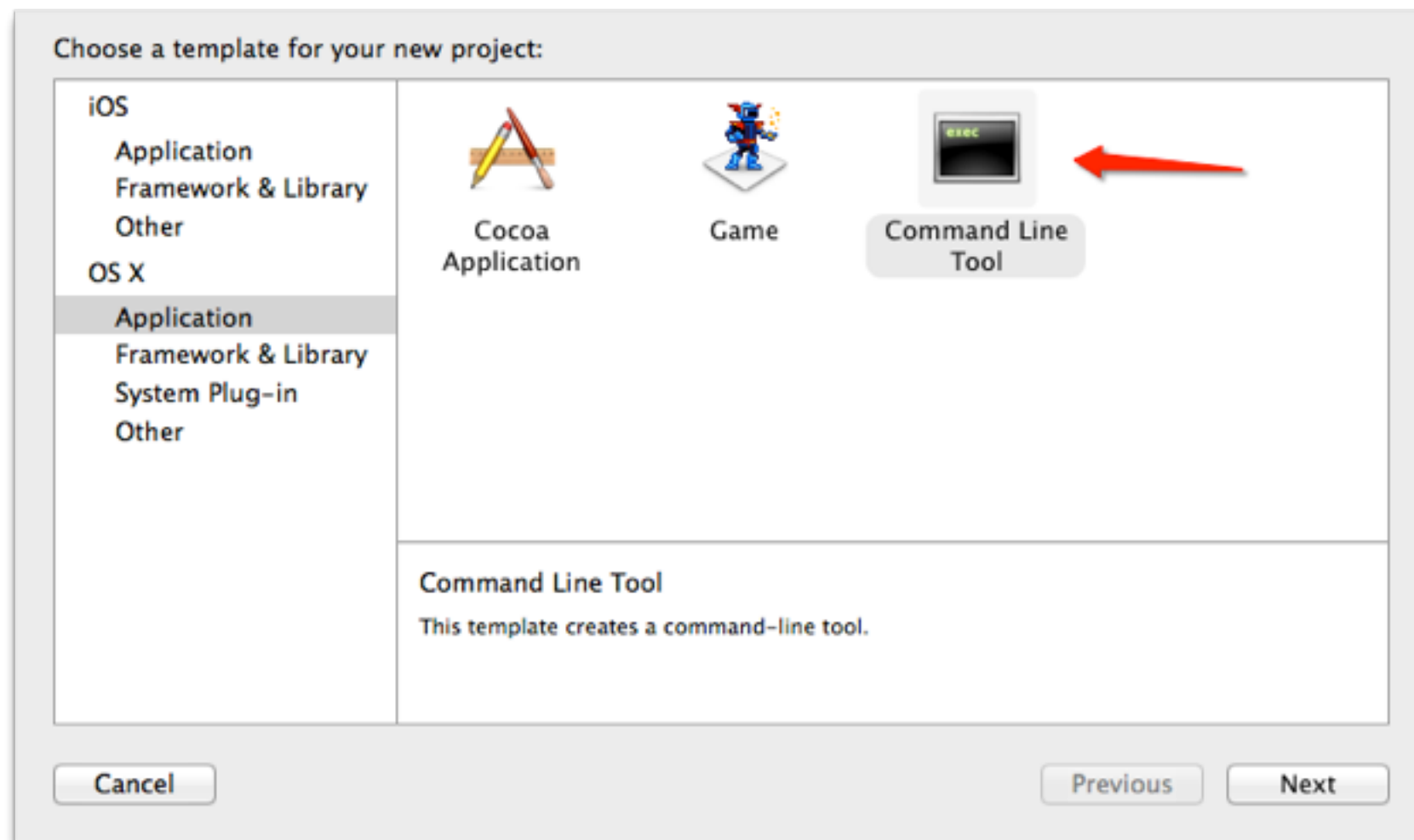
...More

### Information

Category: Developer Tools  
Updated: Sep 18, 2014  
Version: 6.0.1  
Price: Free  
Size: 2.46 GB  
Family Sharing: Yes  
Language: English  
Seller: Apple Inc.  
© 1999–2013 Apple Inc.

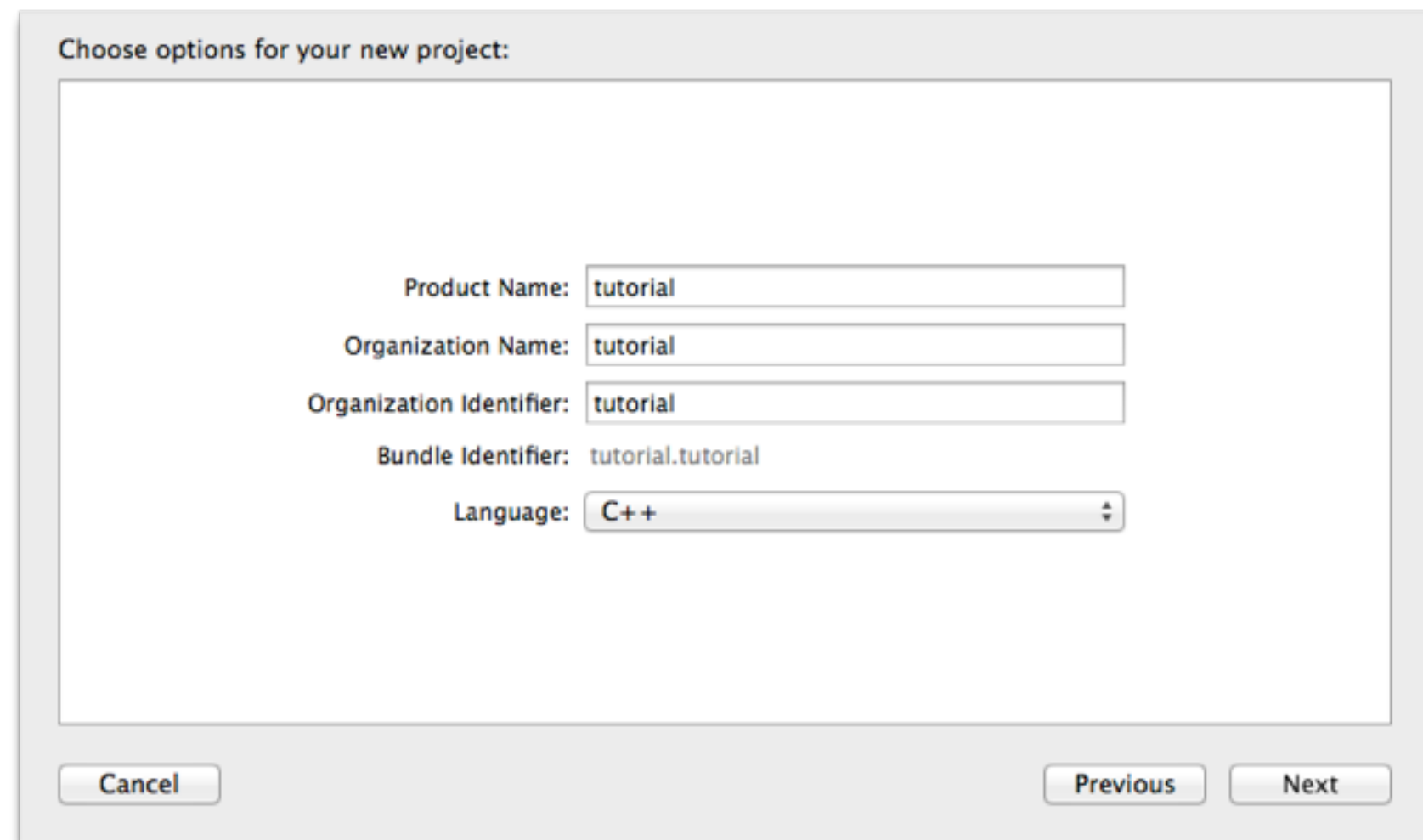
Após a instalação, abra-o e crie um novo projeto.



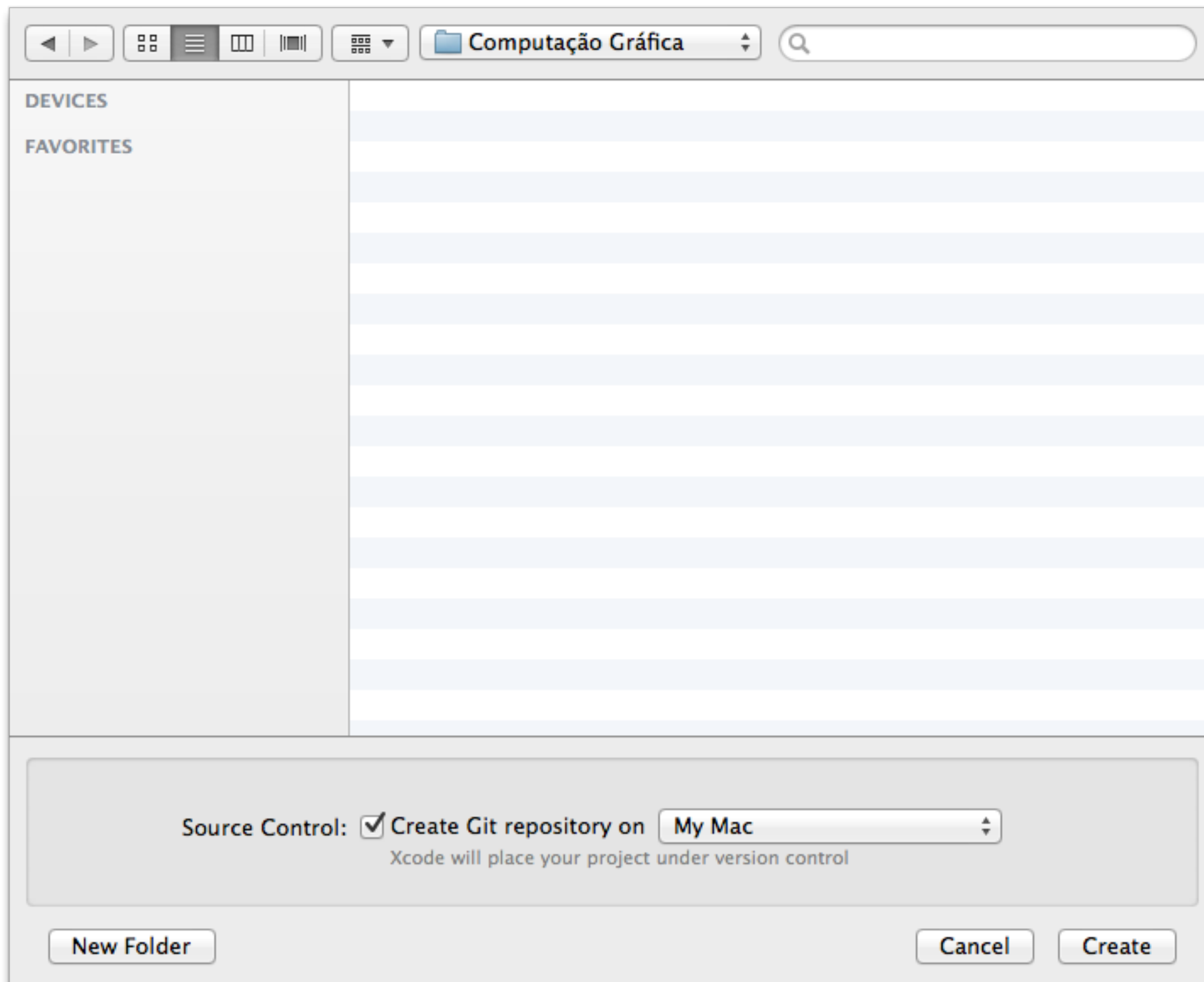


**Selecione “Command Line Tool”.**

**Escolha o nome que desejar.**

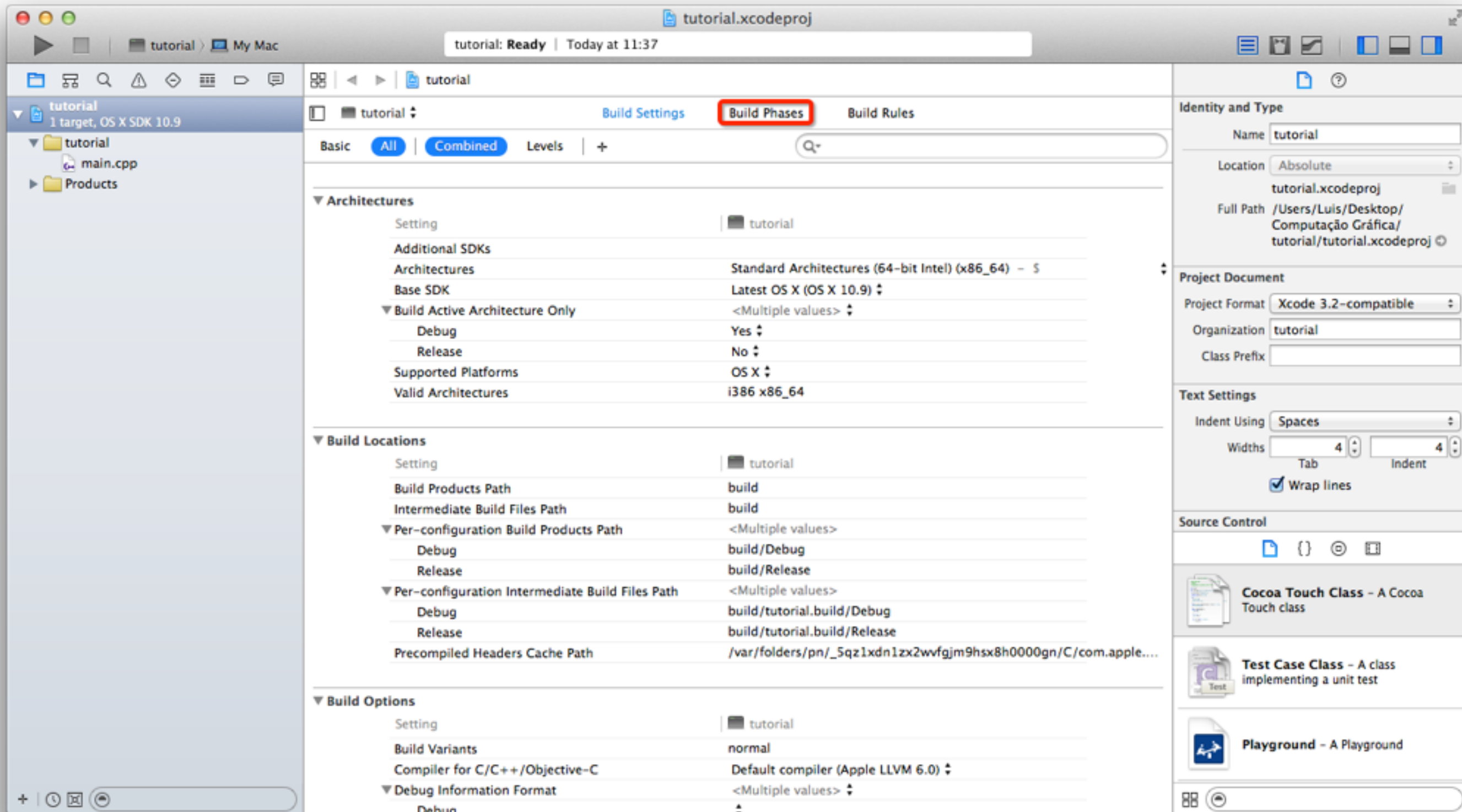


**Escolha o local em que seu projeto será salvo.**

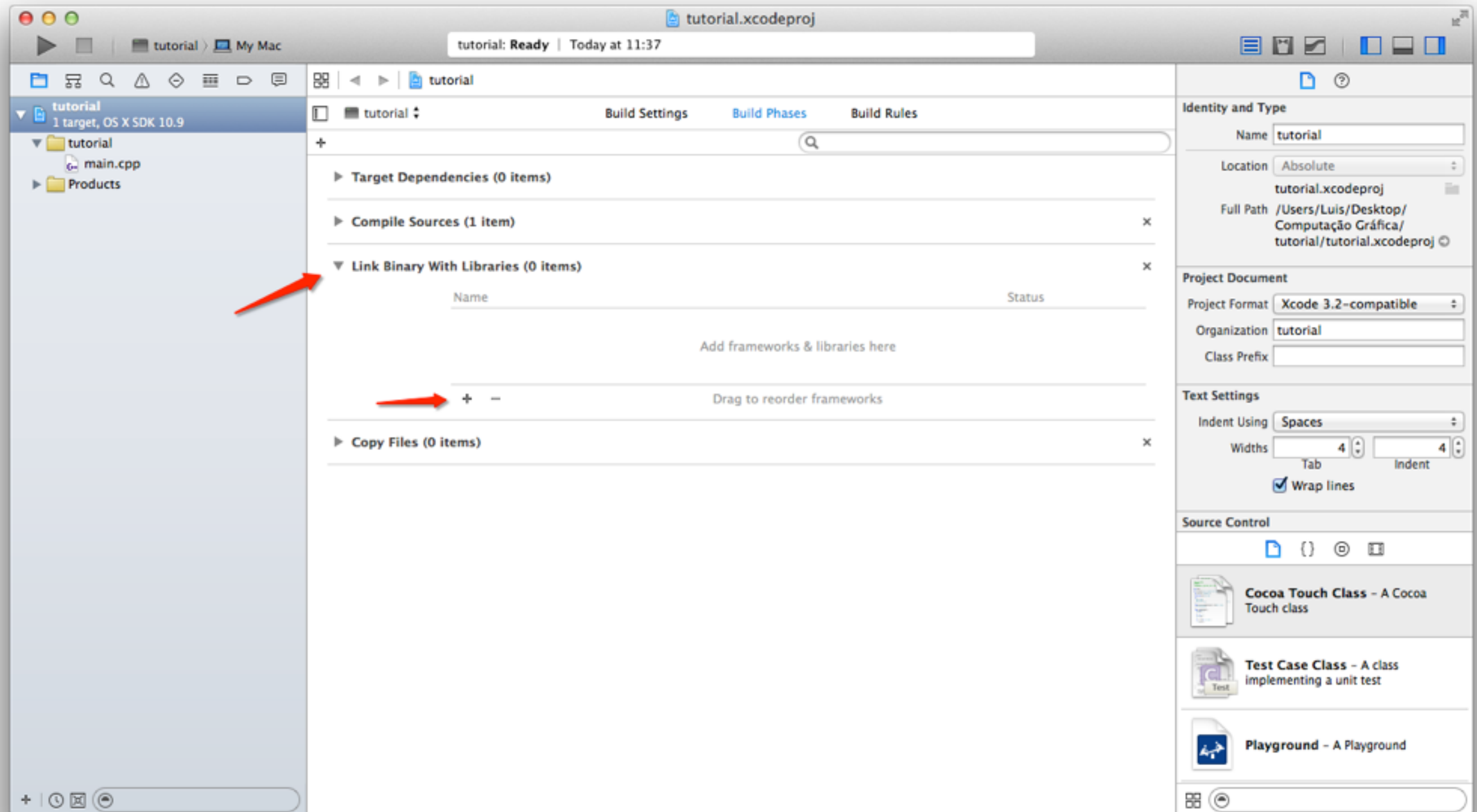


Agora é necessário incluir os frameworks do OpenGL e do GLUT.

Clique em “Build Phases”.

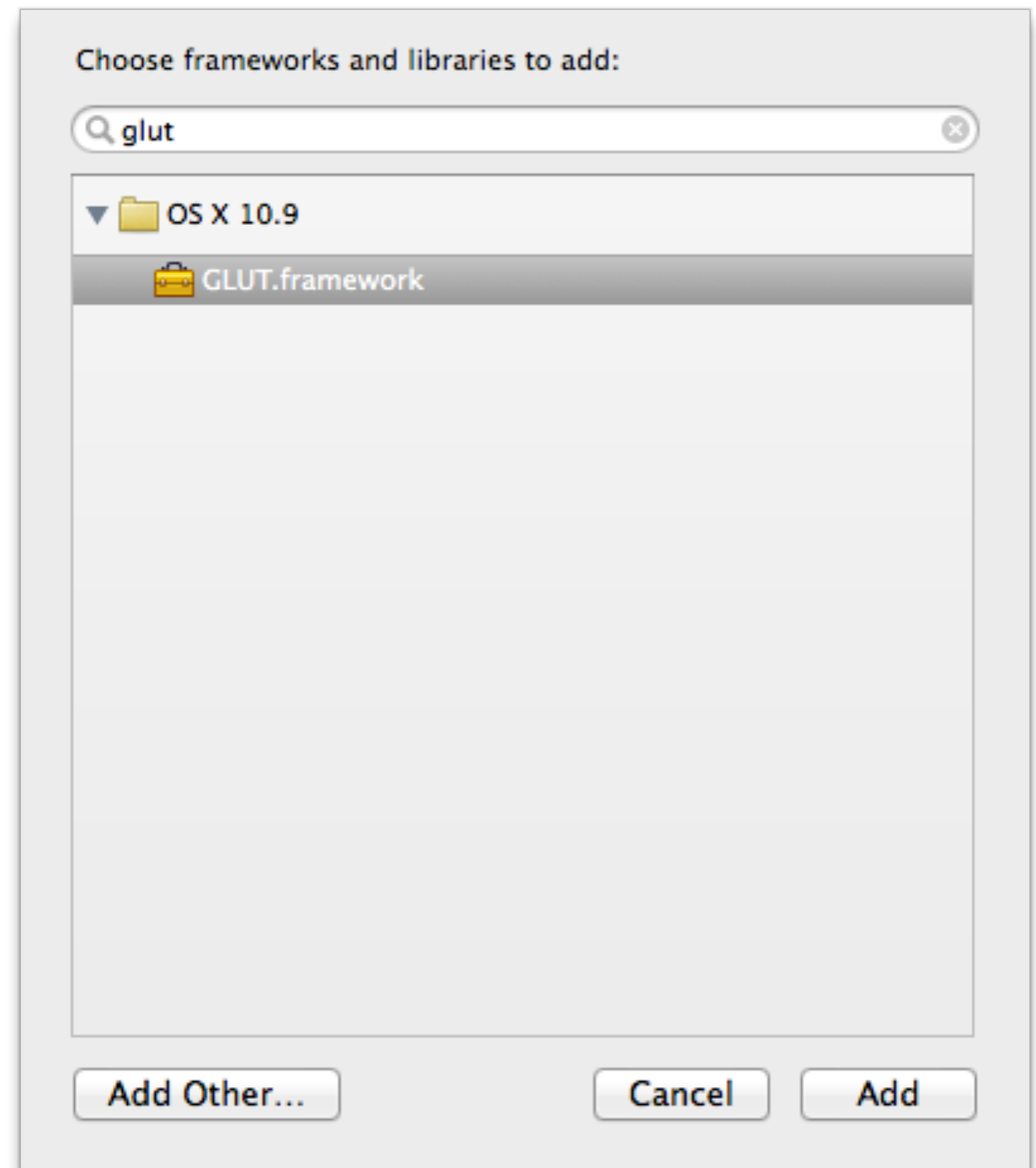
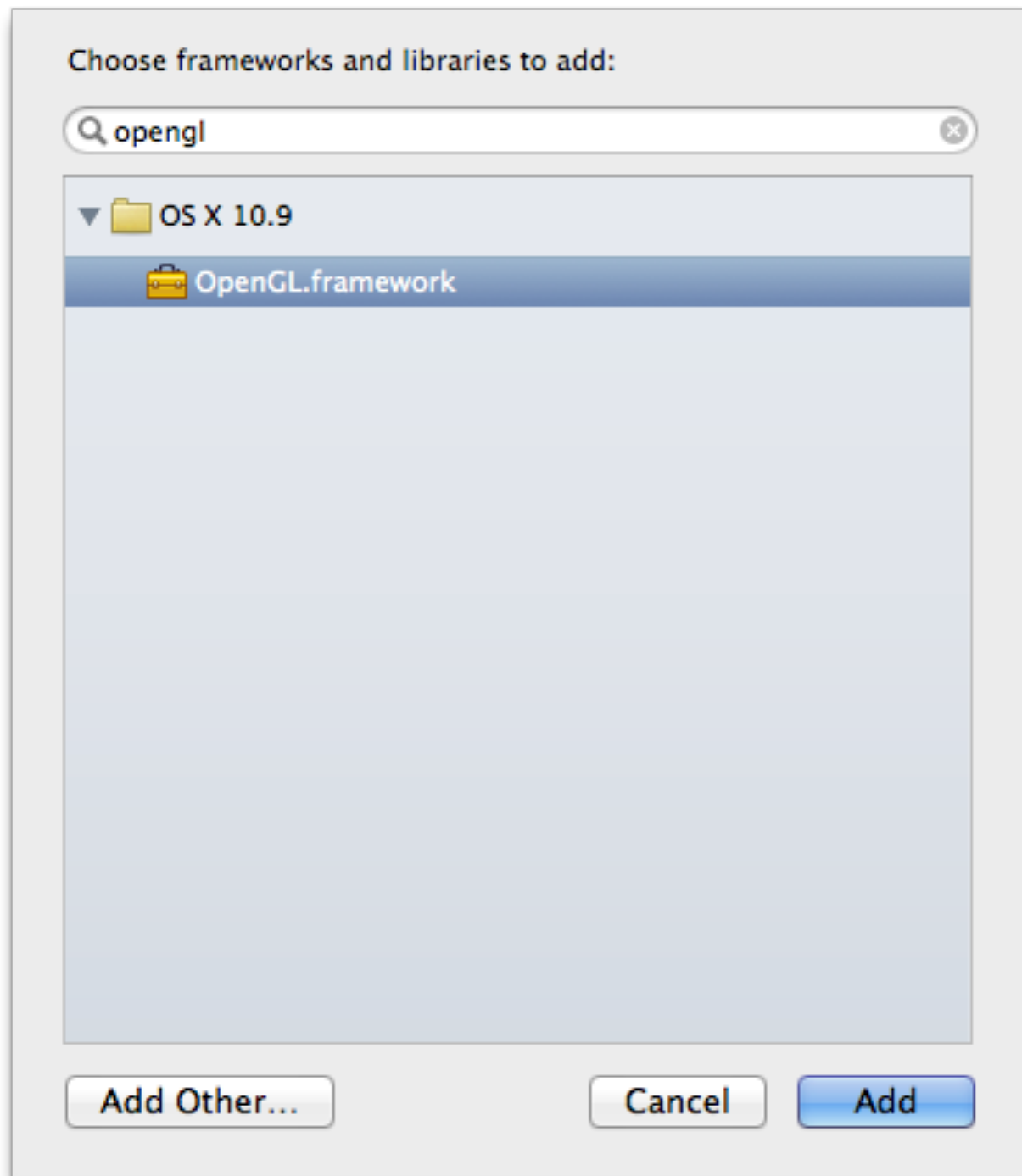


Clique em “Link Binary With Libraries” para abrir o novo menu um clique em “+”.



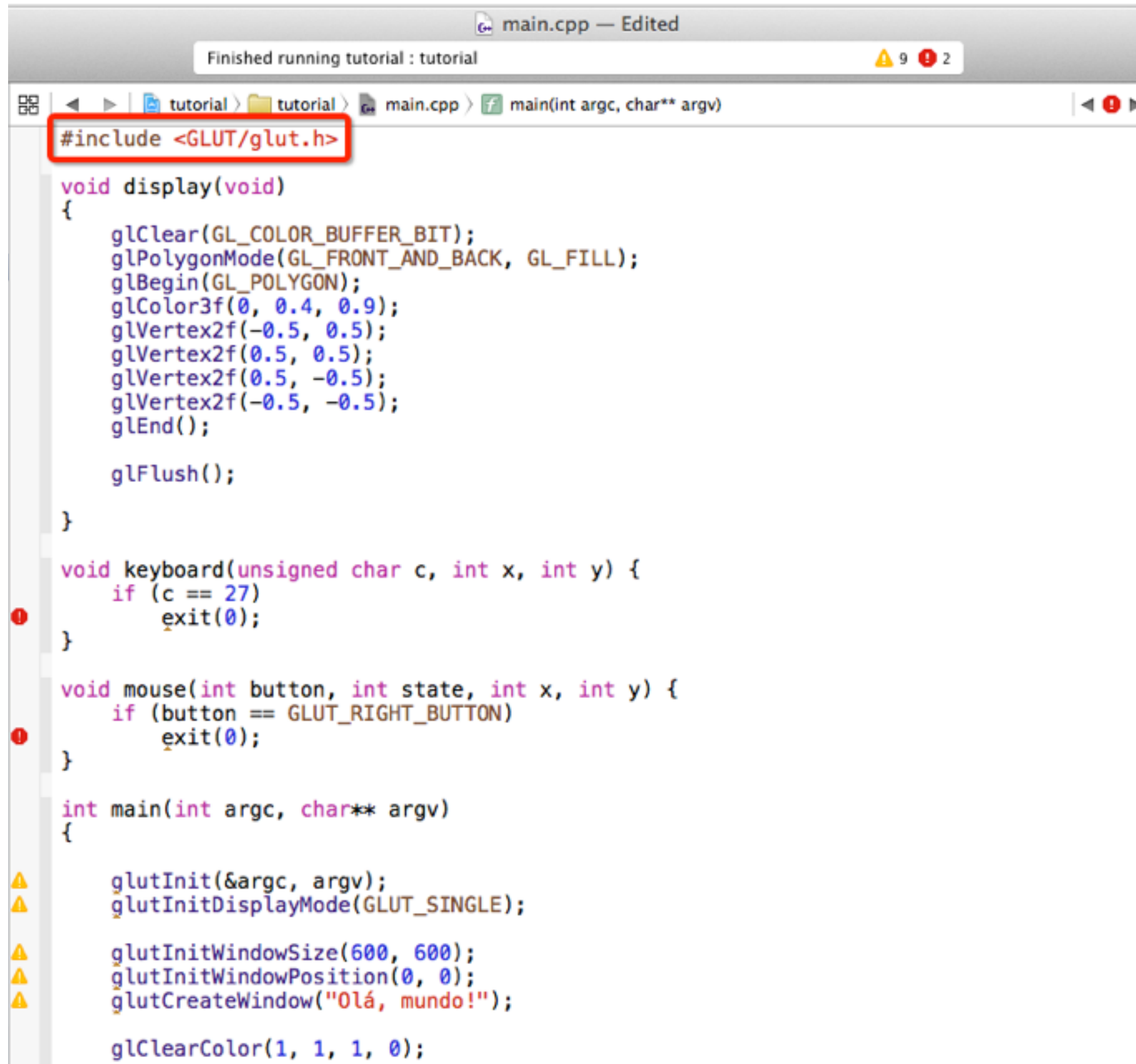


Procure por OpenGL e por GLUT e adicione-os.





A única biblioteca a ser incluída é a <GLUT/glut.h>.



```
main.cpp — Edited
Finished running tutorial : tutorial 9 2
tutorial > tutorial > main.cpp > f main(int argc, char** argv)
#include <GLUT/glut.h>

void display(void)
{
    glClear(GL_COLOR_BUFFER_BIT);
    glPolygonMode(GL_FRONT_AND_BACK, GL_FILL);
    glBegin(GL_POLYGON);
    glColor3f(0, 0.4, 0.9);
    glVertex2f(-0.5, 0.5);
    glVertex2f(0.5, 0.5);
    glVertex2f(0.5, -0.5);
    glVertex2f(-0.5, -0.5);
    glEnd();

    glFlush();
}

void keyboard(unsigned char c, int x, int y) {
    if (c == 27)
        exit(0);
}

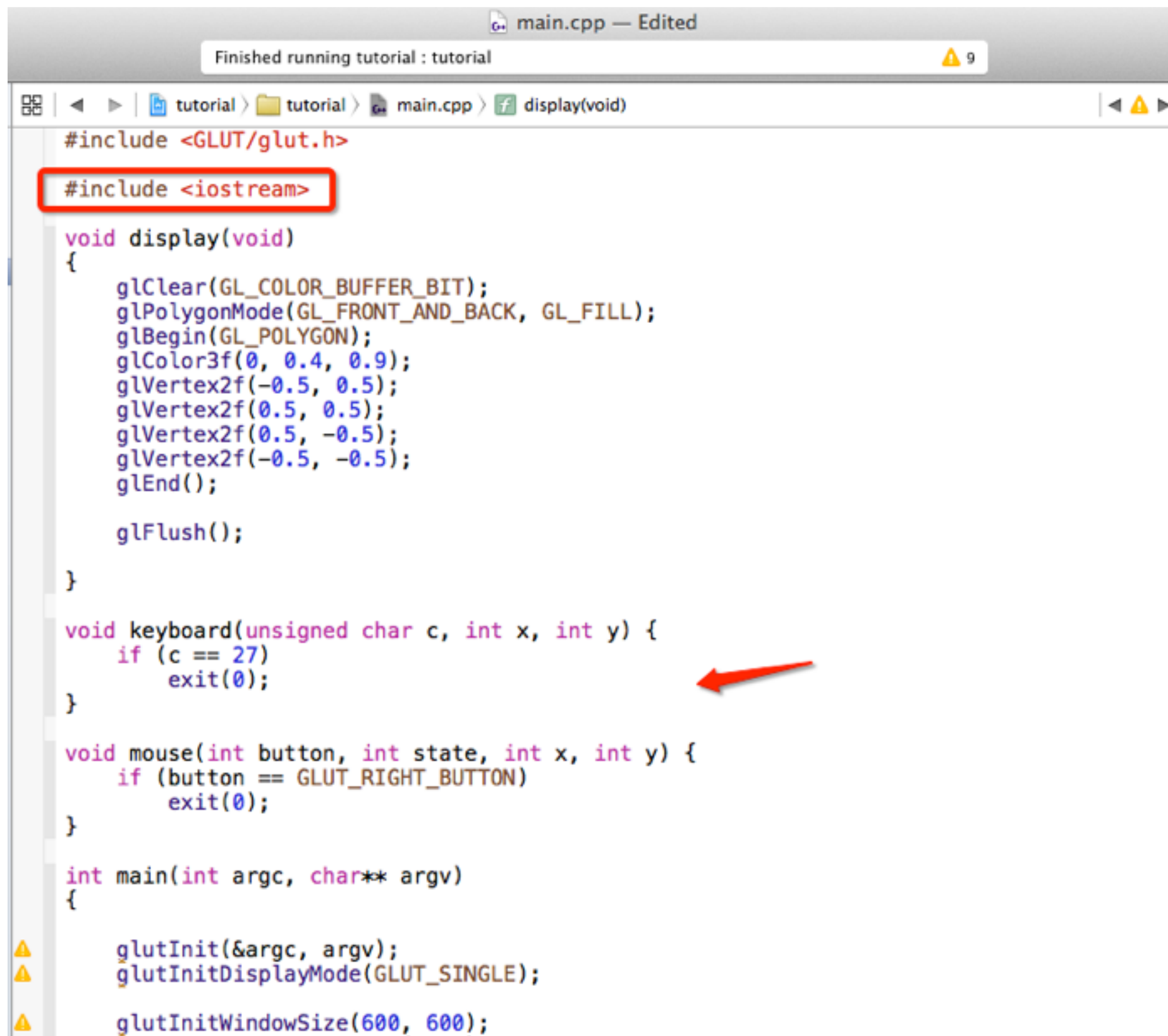
void mouse(int button, int state, int x, int y) {
    if (button == GLUT_RIGHT_BUTTON)
        exit(0);
}

int main(int argc, char** argv)
{
    glutInit(&argc, argv);
    glutInitDisplayMode(GLUT_SINGLE);

    glutInitWindowSize(600, 600);
    glutInitWindowPosition(0, 0);
    glutCreateWindow("Olá, mundo!");

    glClearColor(1, 1, 1, 0);
}
```

Para a comunicação com o teclado e com o mouse, entretanto, é necessário incluir a biblioteca <iostream>.



```
main.cpp — Edited
Finished running tutorial : tutorial
tutorial > tutorial > main.cpp > display(void)
#include <GLUT/glut.h>
#include <iostream>
void display(void)
{
    glClear(GL_COLOR_BUFFER_BIT);
    glPolygonMode(GL_FRONT_AND_BACK, GL_FILL);
    glBegin(GL_POLYGON);
    glColor3f(0, 0.4, 0.9);
    glVertex2f(-0.5, 0.5);
    glVertex2f(0.5, 0.5);
    glVertex2f(0.5, -0.5);
    glVertex2f(-0.5, -0.5);
    glEnd();

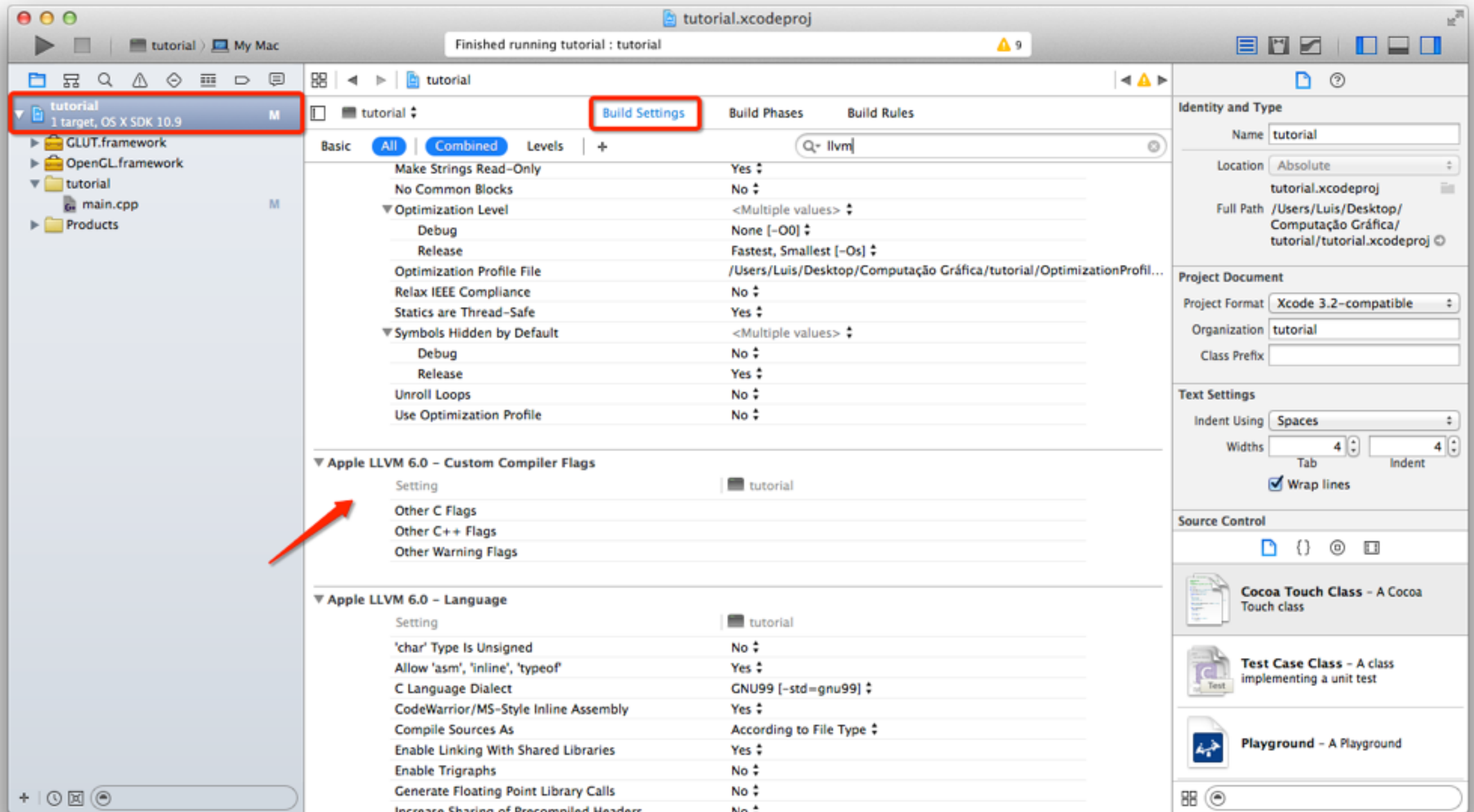
    glFlush();
}

void keyboard(unsigned char c, int x, int y) {
    if (c == 27)
        exit(0);
}

void mouse(int button, int state, int x, int y) {
    if (button == GLUT_RIGHT_BUTTON)
        exit(0);
}

int main(int argc, char** argv)
{
    glutInit(&argc, argv);
    glutInitDisplayMode(GLUT_SINGLE);
    glutInitWindowSize(600, 600);
```

Todas as funções GLUT estarão acompanhadas de um aviso. Isso não atrapalhará o funcionamento do programa, mas, para retirá-los, selecione o nome do seu projeto, clique em “Build Settings” e procure por “LLVM” até encontrar “Custom Compiler Flags”.



**Selezione “Other C Flags”, aperte ENTER e digite “-Wno-deprecated”.**

▼ Apple LLVM 6.0 – Custom Compiler Flags

Setting

tutorial

► Other C Flags

-Wno-deprecated

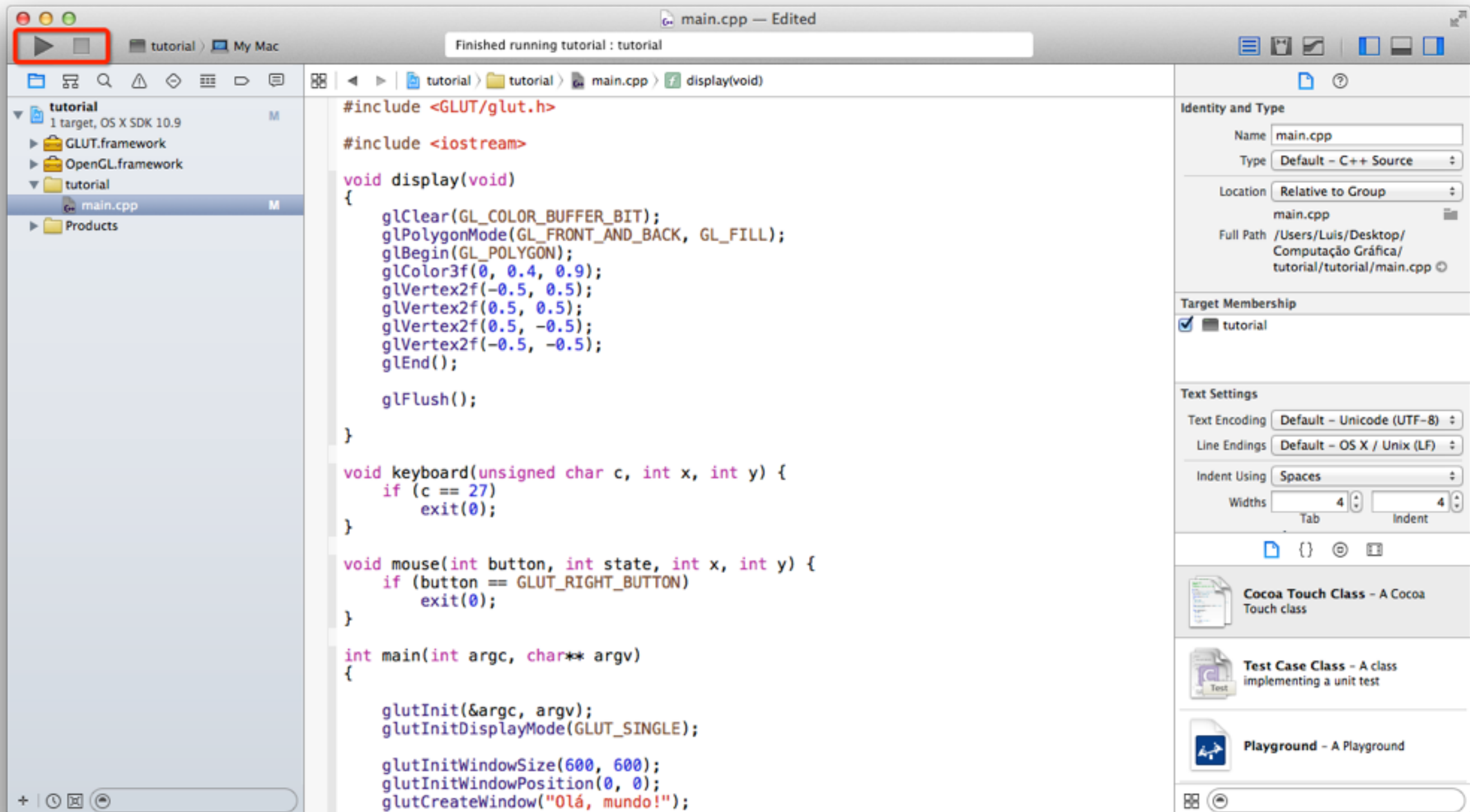
Other C++ Flags

-Wno-deprecated

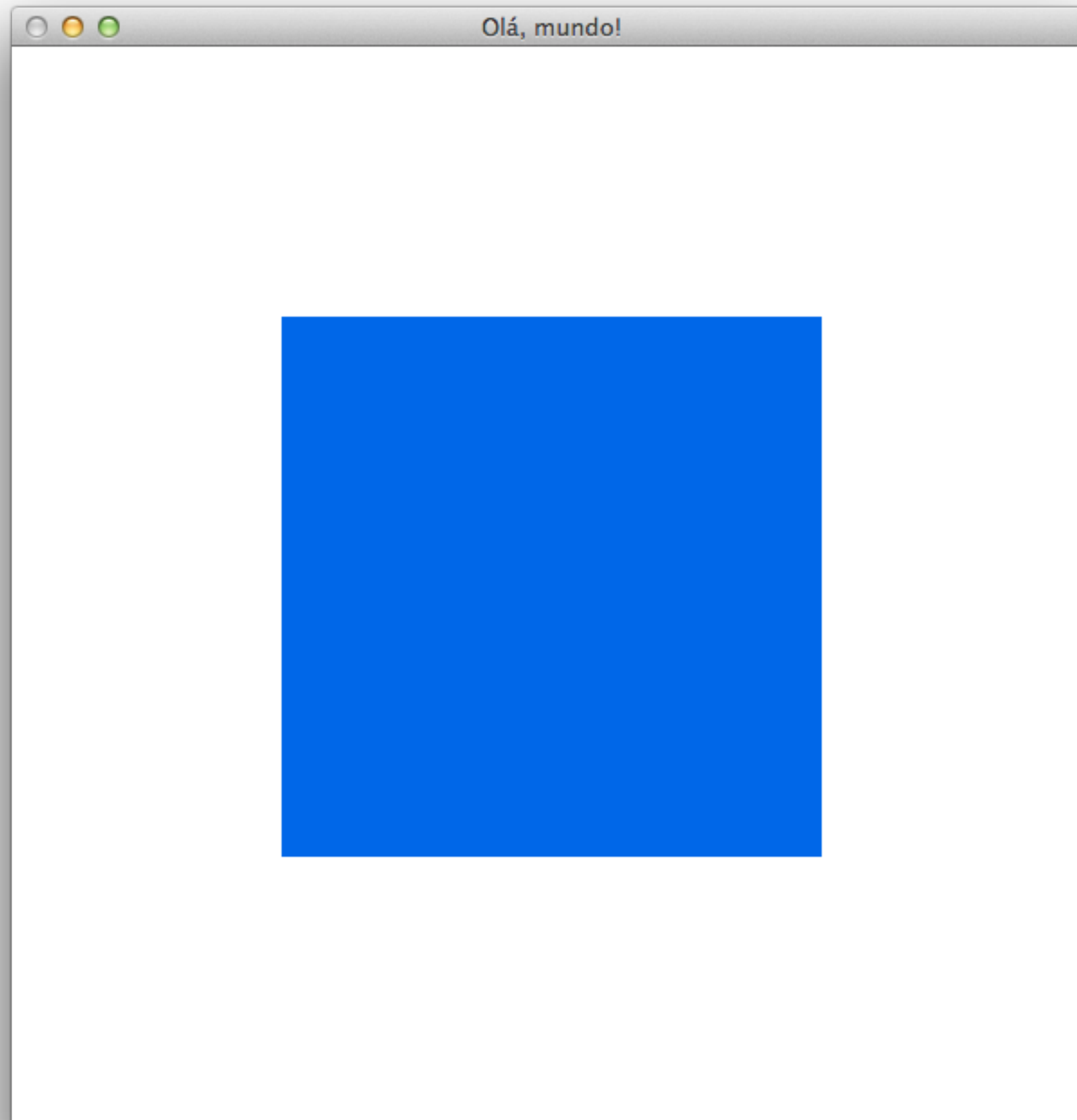
Other Warning Flags

Dessa forma, os avisos desaparecerão.

Execute o programa clicando no símbolo destacado abaixo.



**Pronto! Para fechar a janela, clique no botão de parar ao lado do botão de executar ou aperte Command+Q ou, no caso do programa do tutorial, aperte ESC.**





# Código utilizado no tutorial.

```
#include <GLUT/glut.h>

#include <iostream>

void display(void)
{
    glClear(GL_COLOR_BUFFER_BIT);
    glPolygonMode(GL_FRONT_AND_BACK, GL_FILL);
    glBegin(GL_POLYGON);
    glColor3f(0, 0.4, 0.9);
    glVertex2f(-0.5, 0.5);
    glVertex2f(0.5, 0.5);
    glVertex2f(0.5, -0.5);
    glVertex2f(-0.5, -0.5);
    glEnd();

    glFlush();
}

void keyboard(unsigned char c, int x, int y) {
    if (c == 27)
        exit(0);
}

void mouse(int button, int state, int x, int y) {
    if (button == GLUT_RIGHT_BUTTON)
        exit(0);
}

int main(int argc, char** argv)
{
    glutInit(&argc, argv);
    glutInitDisplayMode(GLUT_SINGLE);

    glutInitWindowSize(600, 600);
    glutInitWindowPosition(0, 0);
    glutCreateWindow("Olá, mundo!");

    glClearColor(1, 1, 1, 0);

    glutDisplayFunc(display);
    glutKeyboardFunc(keyboard);
    glutMouseFunc(mouse);

    glutMainLoop();
    return 0;
}
```