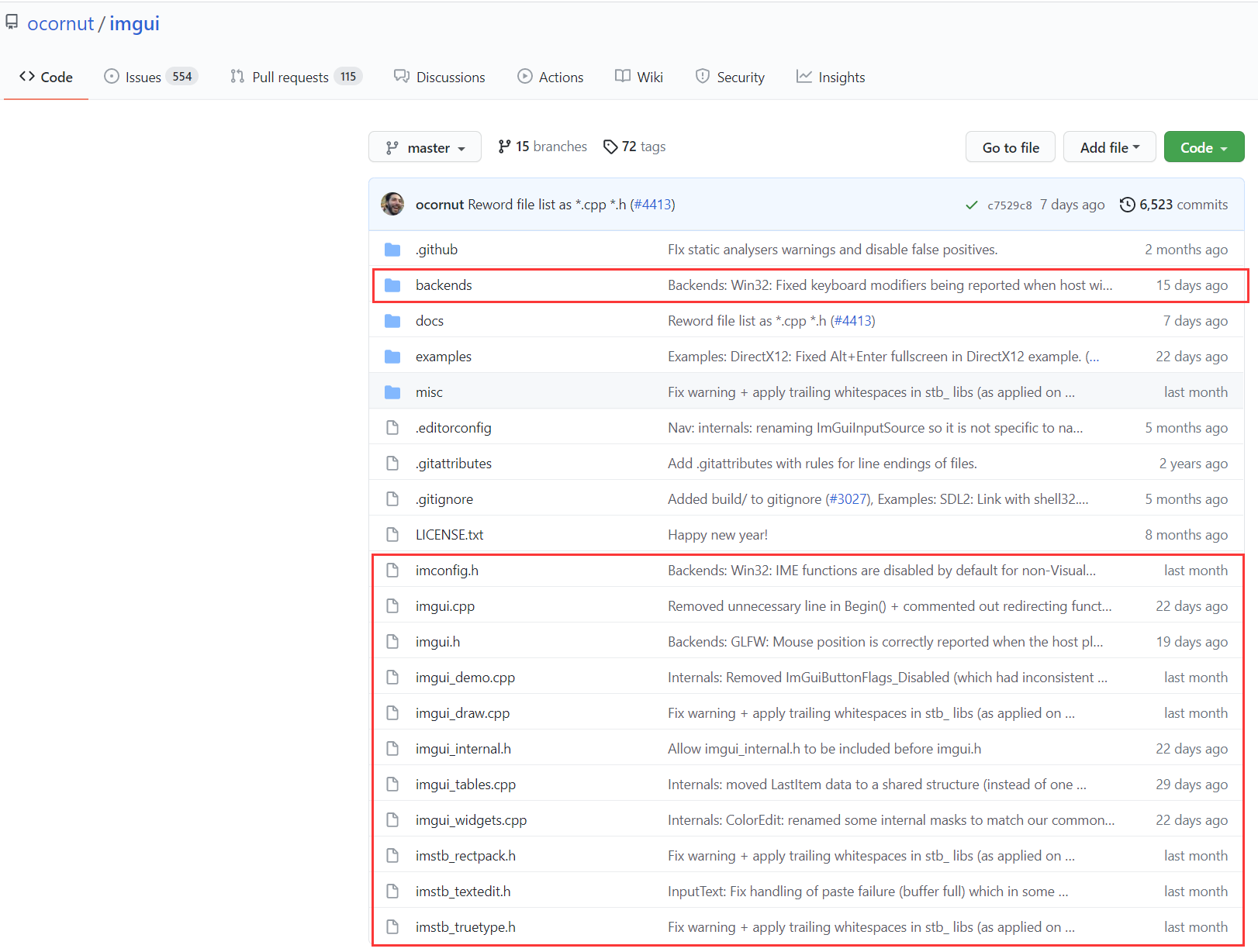
Imgui是一种新的ui实现思路，它将gui变成分为前端和后端分离开来。前端是指呈现给用户的界面，例如文本框，勾选框等；后端是指具体的系统级实现。

Imgui将前后端分离之后，使用者们可以使用不同的后端（例如opengl,vulkan,directX等）来实现前端的界面效果。由于我们的图形学环境配置基于opengl下的glfw和glad，正好与imgui其中的一个后端相合，因而，在完成了1.1环境配置的基础上，我们可以再进一步地配置imgui环境，方便编程使用。

1. Github上的imgui界面如下所示，



红框中的.h和.cpp文件是imgui的前端定义源码，backends文件夹下是不同的后端实现源码。在下载源码之后，首先将前端源码复制到实验1.1代码的同级文件夹下。下载地址：[Release v1.84.2 · ocornut/imgui · GitHub](https://github.com/ocornut/imgui/releases/tag/v1.84.2)

1. 进入到backends文件夹下，将后端源码

imgui\_impl\_glfw.cpp

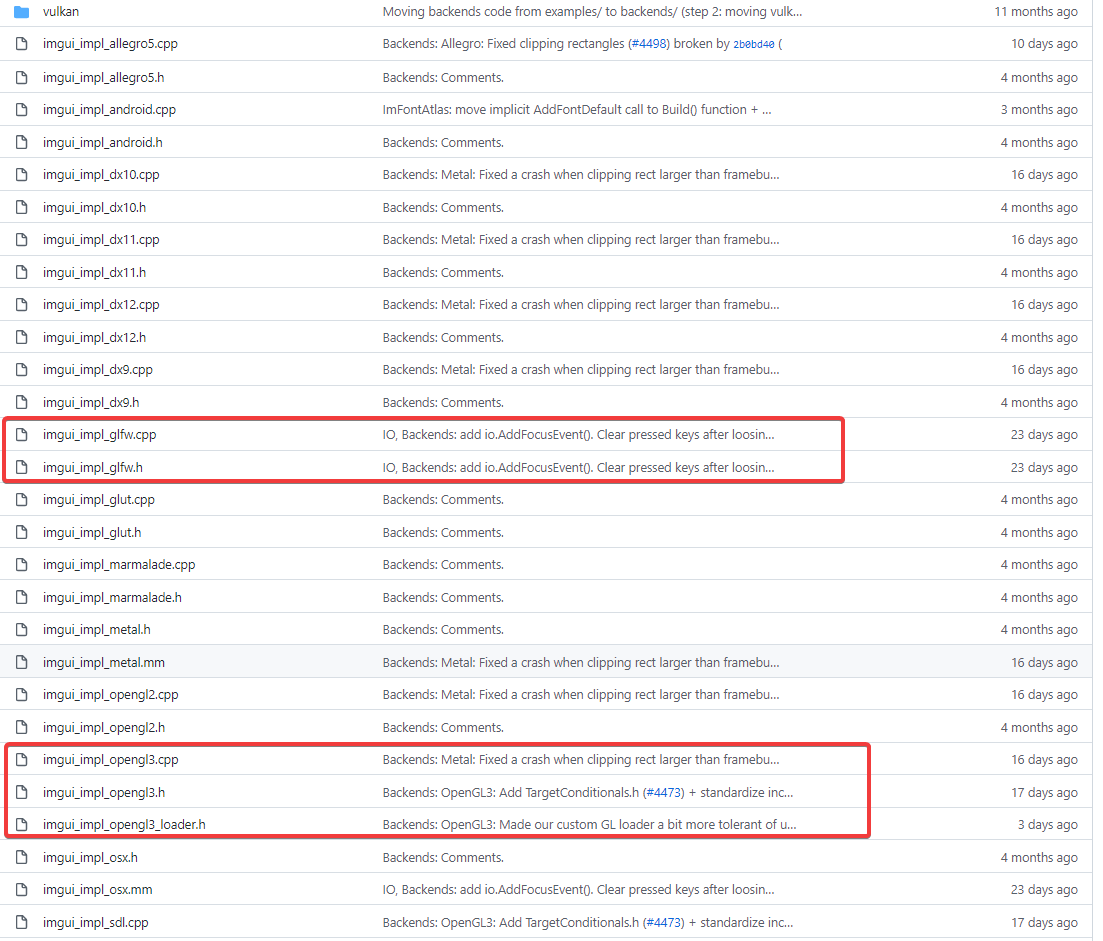
imgui\_impl\_glfw.h

imgui\_impl\_opengl3.h

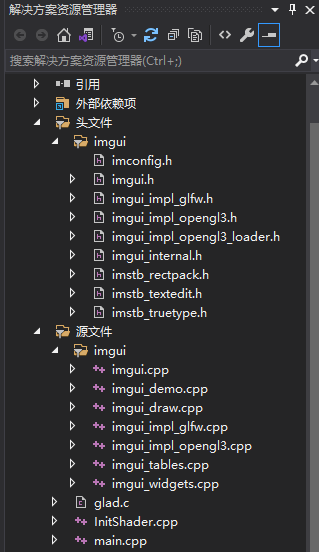
imgui\_impl\_opengl3.cpp

imgui\_impl\_opengl3\_loader.h

也放到刚才的文件夹下



1. 仿照nuget中添加源文件的方法，将.h文件添加到头文件中，.cpp文件添加到源文件之中，如图所示：

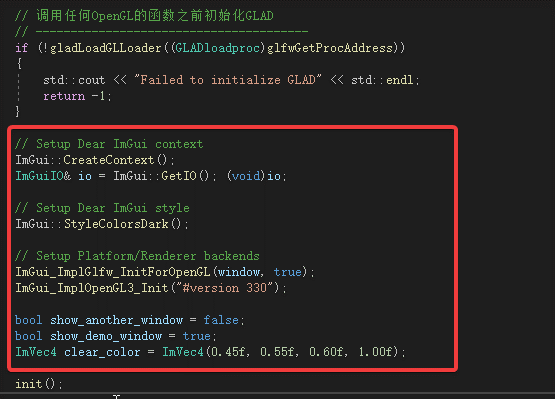


1. Imgui源码下examples文件夹中提供了不同后端的使用样例，我们进入到example\_**glfw**\_opengl3文件夹下，有一个main.cpp文件。现在，我们仿照example\_**glfw**\_opengl/main.cpp里面的代码修改实验1.1中main.cpp的代码：

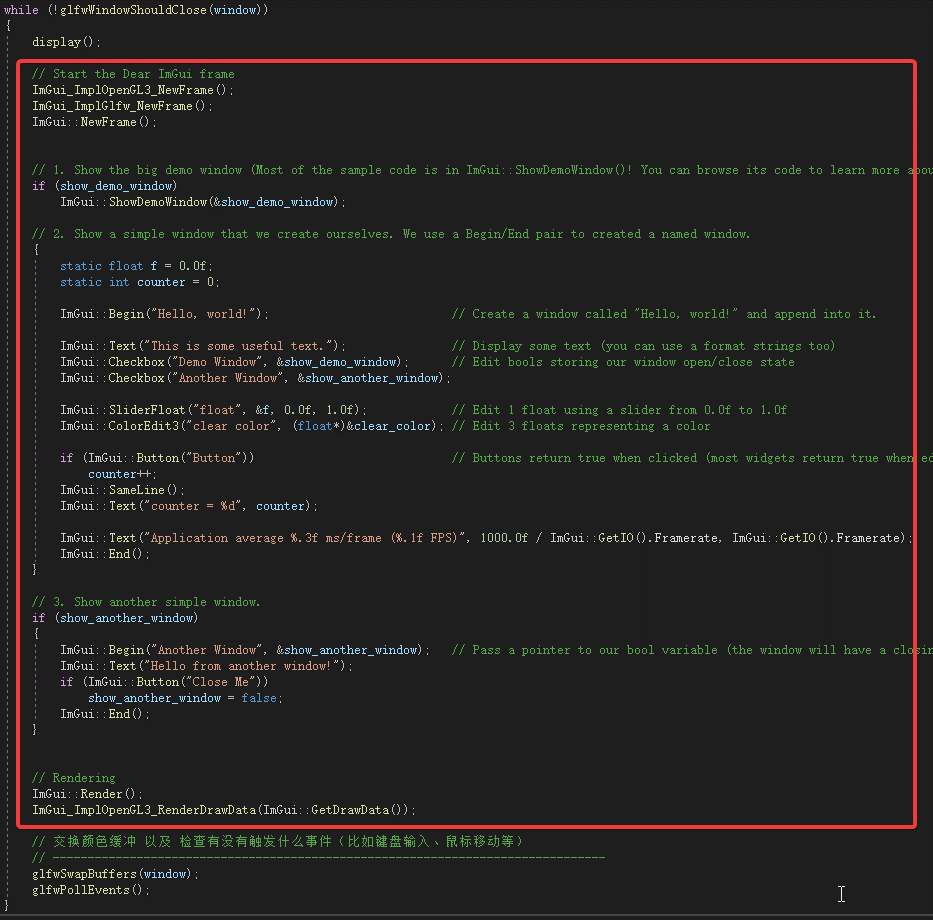
首先，在开头引入如下头文件：



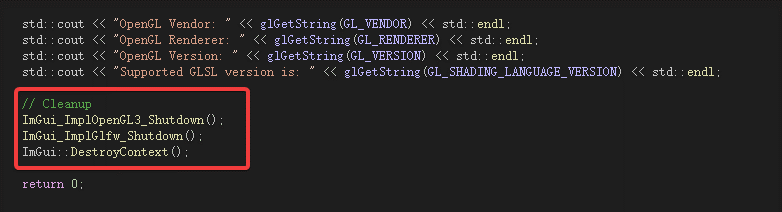
接下来，在**init()之前**插入如下代码：



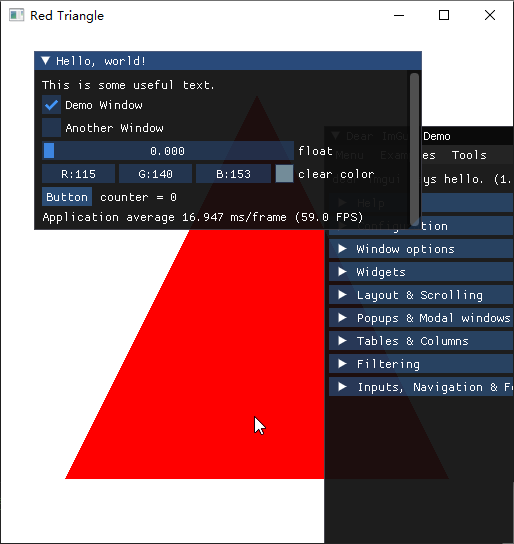
接着，在**while循环**中直接复制示例里相同位置的代码：



最后，在return之前输入如下代码：



1. 编译并运行程序， 可以得到如下界面：



1. Imgui的具体使用方法可以参考Demo中的演示。例如，可以将ColorEdit3中的变量和glClearColor绑定，实现随时更改背景颜色的效果:

