

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ
ФЕДЕРАЦИИ**
Федеральное государственное автономное образовательное учреждение
высшего профессионального образования
«Санкт-Петербургский государственный университет
аэрокосмического приборостроения»

Компьютерная графика

Методические указания к выполнению лабораторных работ для
студентов по направлениям “программная инженерия”,
“математическое обеспечение и администрирование
информационных систем”, “прикладная математика и
информатика”

Санкт-Петербург
2015

Использование среды Microsoft Visual Studio

Подготовка к выполнению лабораторных работ

В данном разделе пойдёт речь о создании приложения (решения, проекта) в IDE **Microsoft Visual Studio** на примере **Microsoft Visual Studio 2008 Express Edition** для работы с библиотекой OpenGL в рамках выполнения лабораторных работ по дисциплине “Компьютерная графика”.

Необходимые инструменты и библиотеки

Для выполнения ЛР по компьютерной графике необходимо иметь

Одну из ниже перечисленных IDE Microsoft Visual Studio

- **2005 Standard Edition**
- **2008 Express Edition**
- **2010 Express Edition**

Библиотеки

- **glut** или **free glut**
- **glaux**

Предполагается, что на вашем компьютере уже установлена одна из выше перечисленных версий IDE **Microsoft VS**, или вы знаете как установить данную IDE. Далее необходимо загрузить с сервера ГУАП архивы библиотек glut (или free glut) и glaux разархивировать их и переместить файлы библиотек следующим образом:

1) Заголовочные файлы *.h из папок **include** необходимо поместить в **c:\Program Files\Microsoft SDKs\Windows\%version%\Include\gl**

version\$ - версия используемых библиотек и компилятора. Для MS Visual Studio 2008 это версия v6.0A тогда строка будет выглядеть как C:\Program Files\Microsoft SDKs\Windows\v6.0A\

2) Библиотеки *.lib из папок **lib** необходимо поместить в **c:\Program Files\Microsoft SDKs\Windows\%version%\Lib**

Создание (проекта)

Для того, чтобы создать решение (проект) в среде Microsoft VS 2008 требуется из верхнего меню выбрать пункт «Файл». В открывшемся меню выбрать «Создать» и далее «Проект» (рис.1). В различных версиях IDE Microsoft VS вместо слова «решение» может использоваться проект или project в случае английской версии программы.

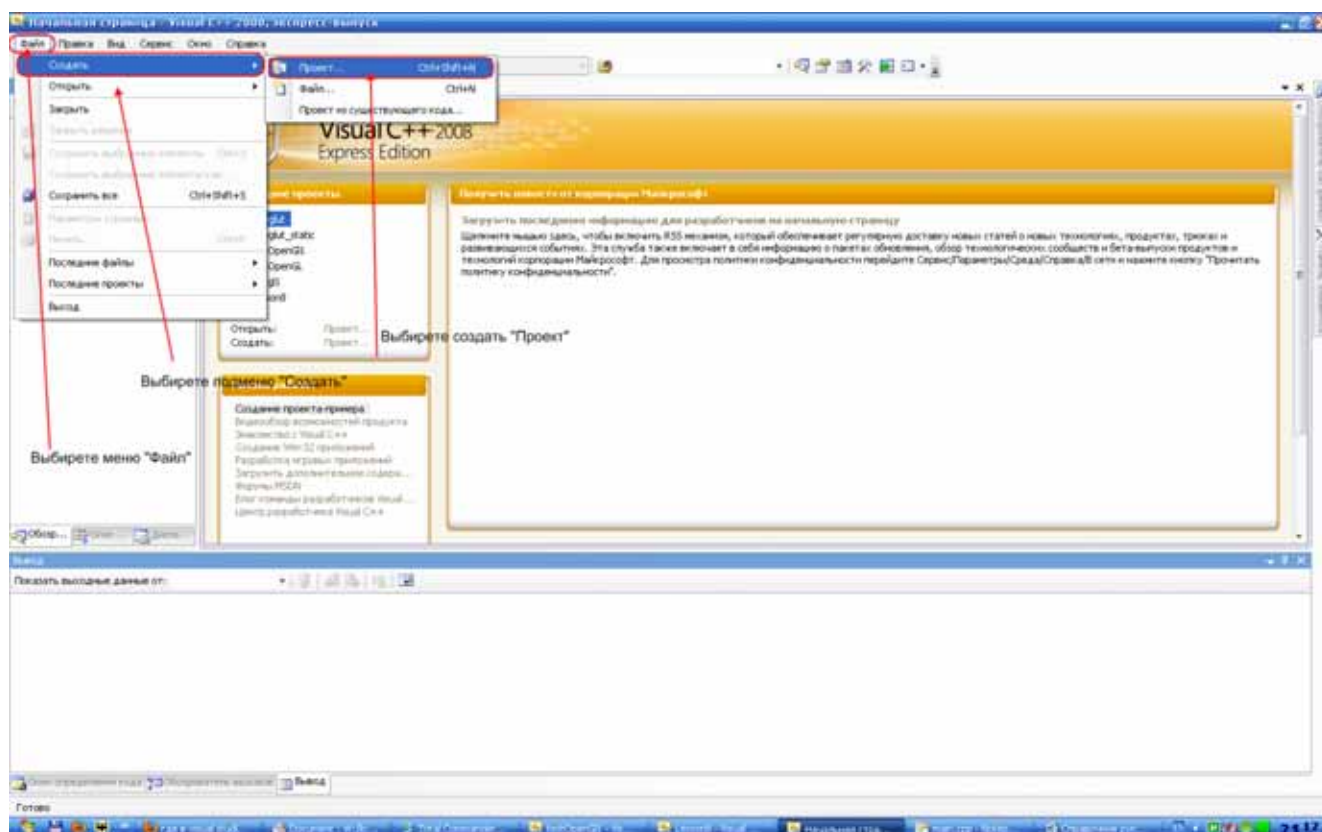


Рис. 1 : Создание проекта. «Файл» «Создать» «Проект»

Далее откроется диалоговое окно, изображенное на рисунке 2. В данном окне требуется указать, что проект будет являться проектом Win32, ввести уникальное имя проекта, а так же указать каталог, в который Вы хотите сохранить проект и все его файлы. После того, как данные операции проделаны, нужно нажать Ок, после чего откроется следующее окно, изображенное на рисунке 3.

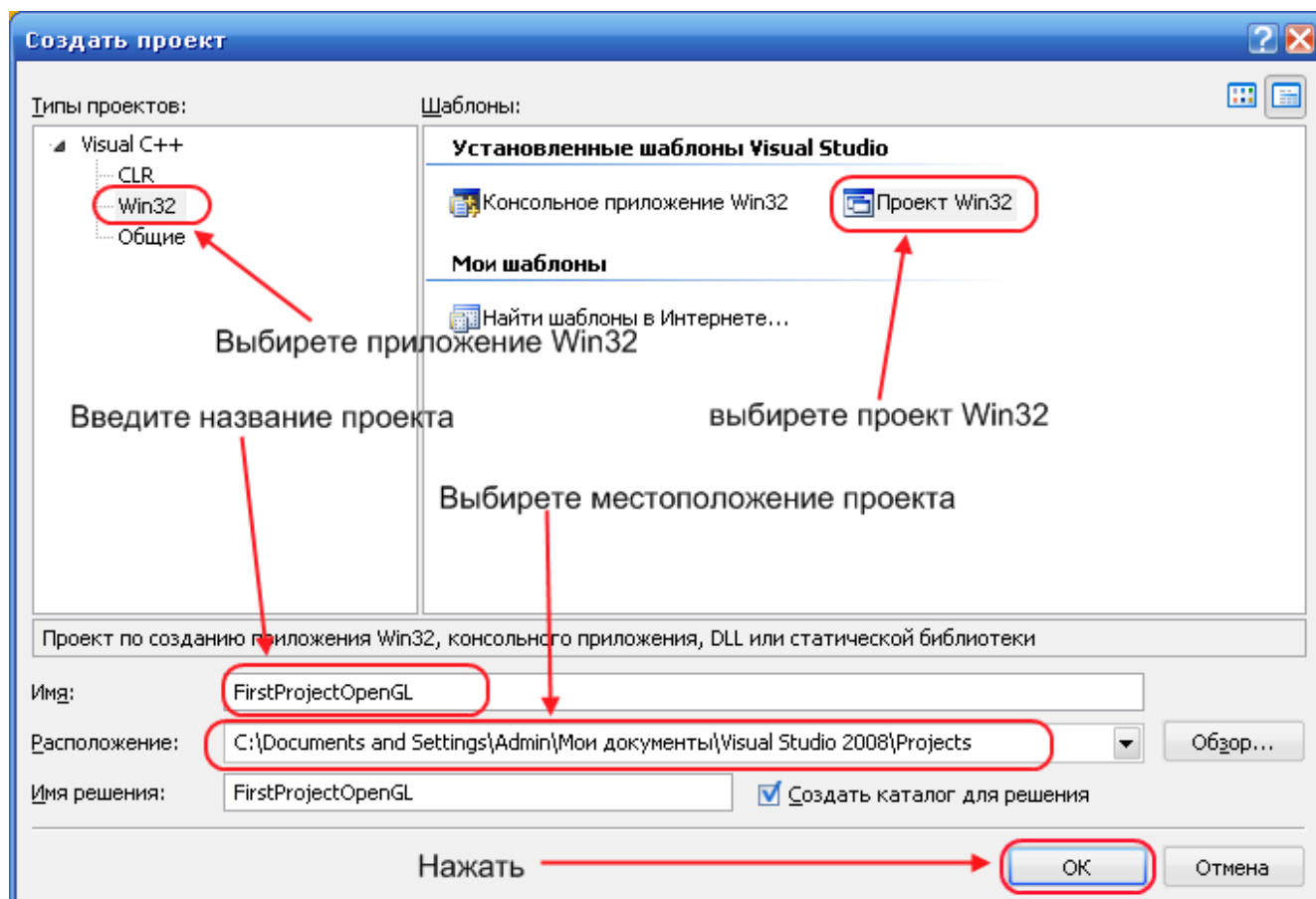


Рис.2 : Окно «Создать проект»

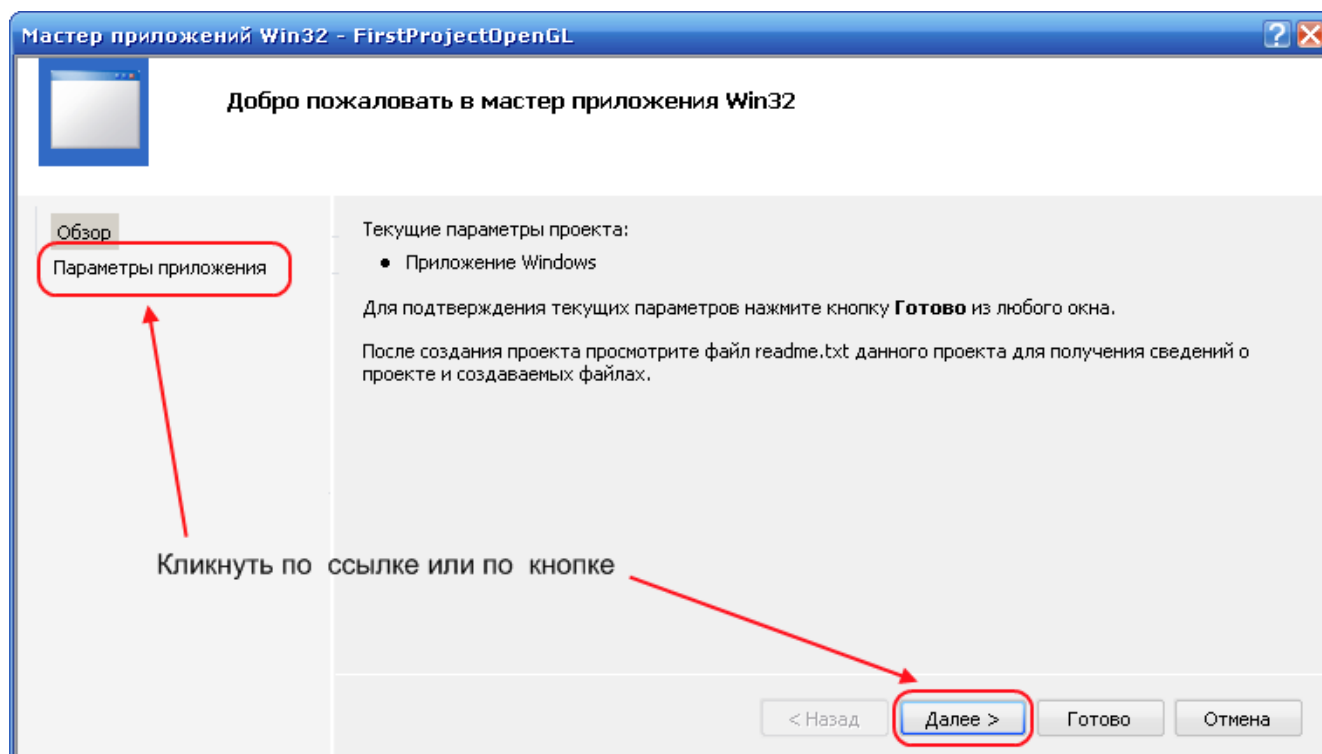


Рис.3 : окно «Мастер приложений Win32»

В данном окне необходимо перейти к закладке «Параметры приложения» из меню в левой части окна и, как показано на рисунке 4, убедиться, что типом приложения указано «Приложение Windows», а в дополнительных параметрах стоит галочка напротив пункта «Пустой проект».

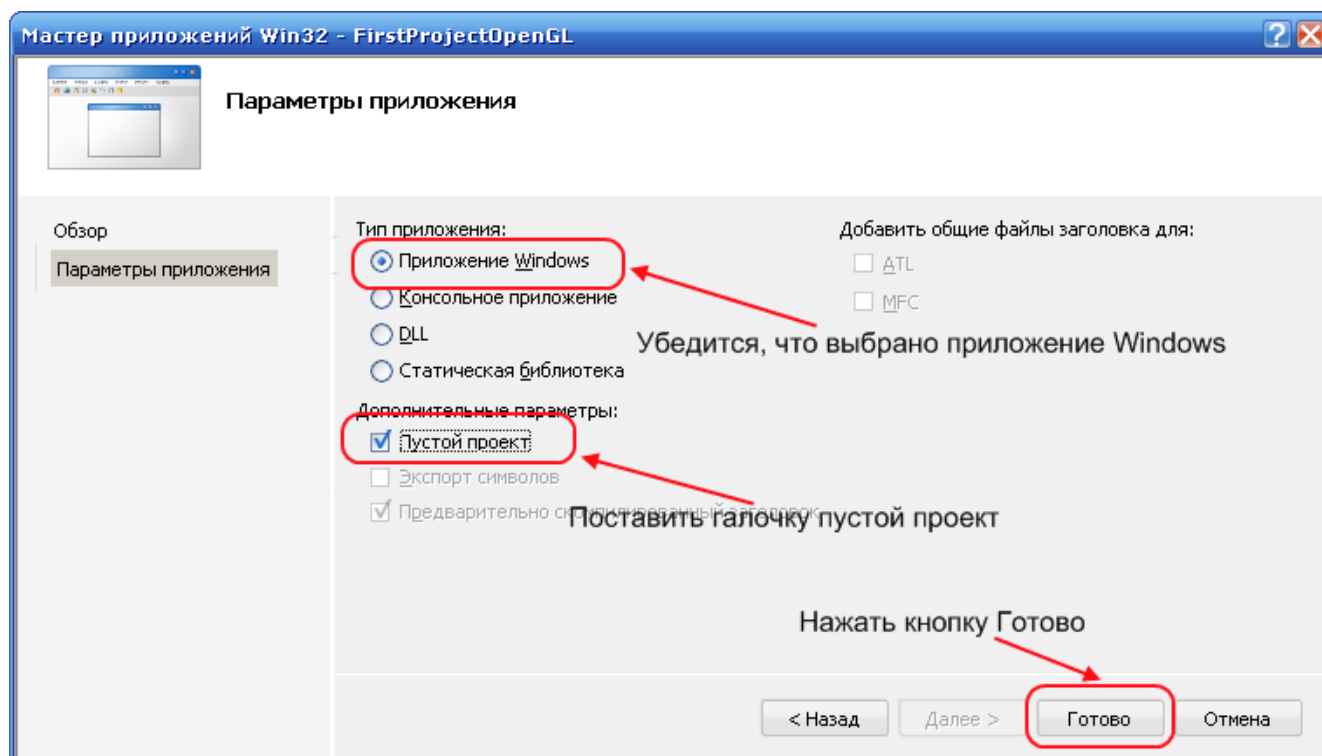


Рис.4 : окно «Мастер приложений» Win32. Параметры приложения

Проект создан. Теперь требуется добавить в проект файлы, необходимые для работы с библиотекой OpenGL. Требуется щелчком правой кнопки мыши на пункте «Файлы исходного кода» в окне «Обозреватель решений» открыть выпадающее меню, далее в нем выбрать пункт «Добавить» и далее «Создать элемент...», как показано на рисунке 5.

Настройка решения (проекта)

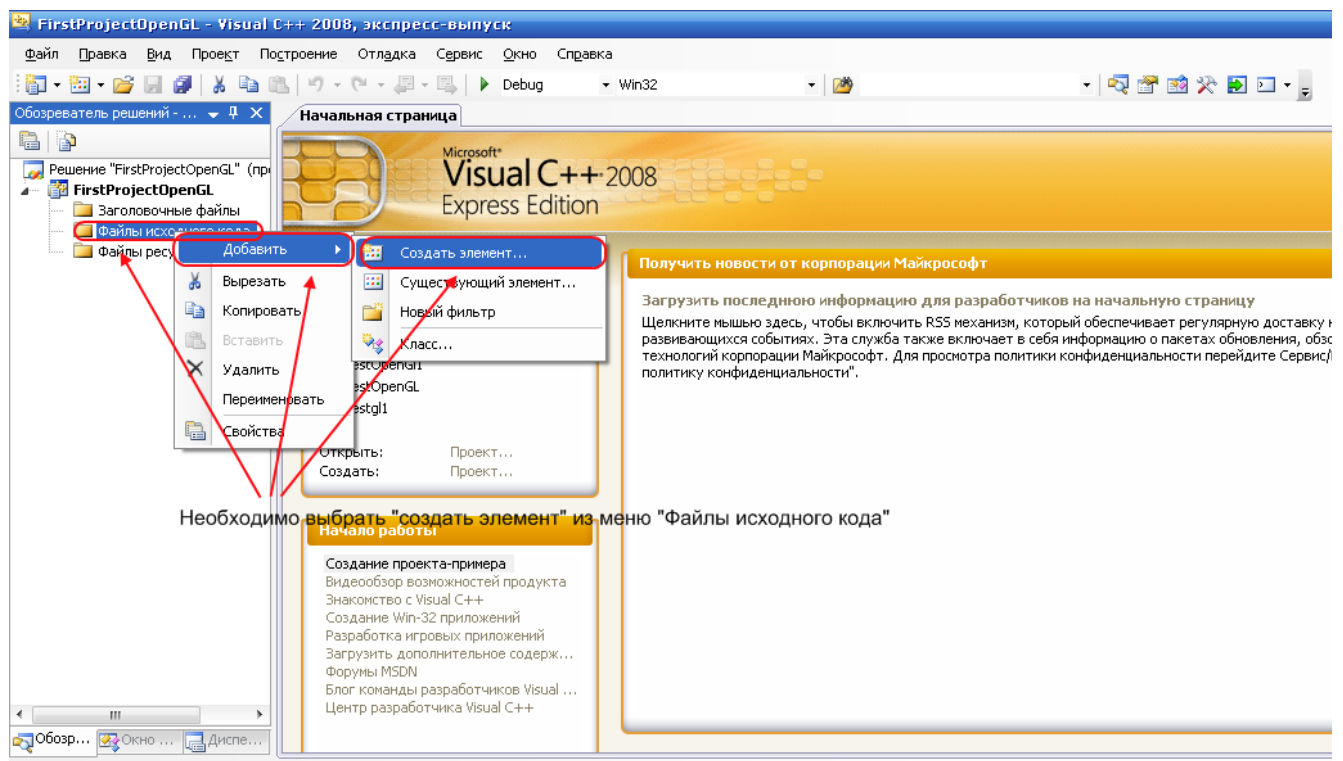


Рис. 5: Добавление файлов к проекту.

В открывшемся окне, изображенном на рисунке 6, требуется из списка в левой части окна выбрать пункт «Код» элемента Visual C++, в изменившемся окне справа выбрать «Файл C++(.cpp)», ввести уникальное имя файла и нажать кнопку «Добавить»

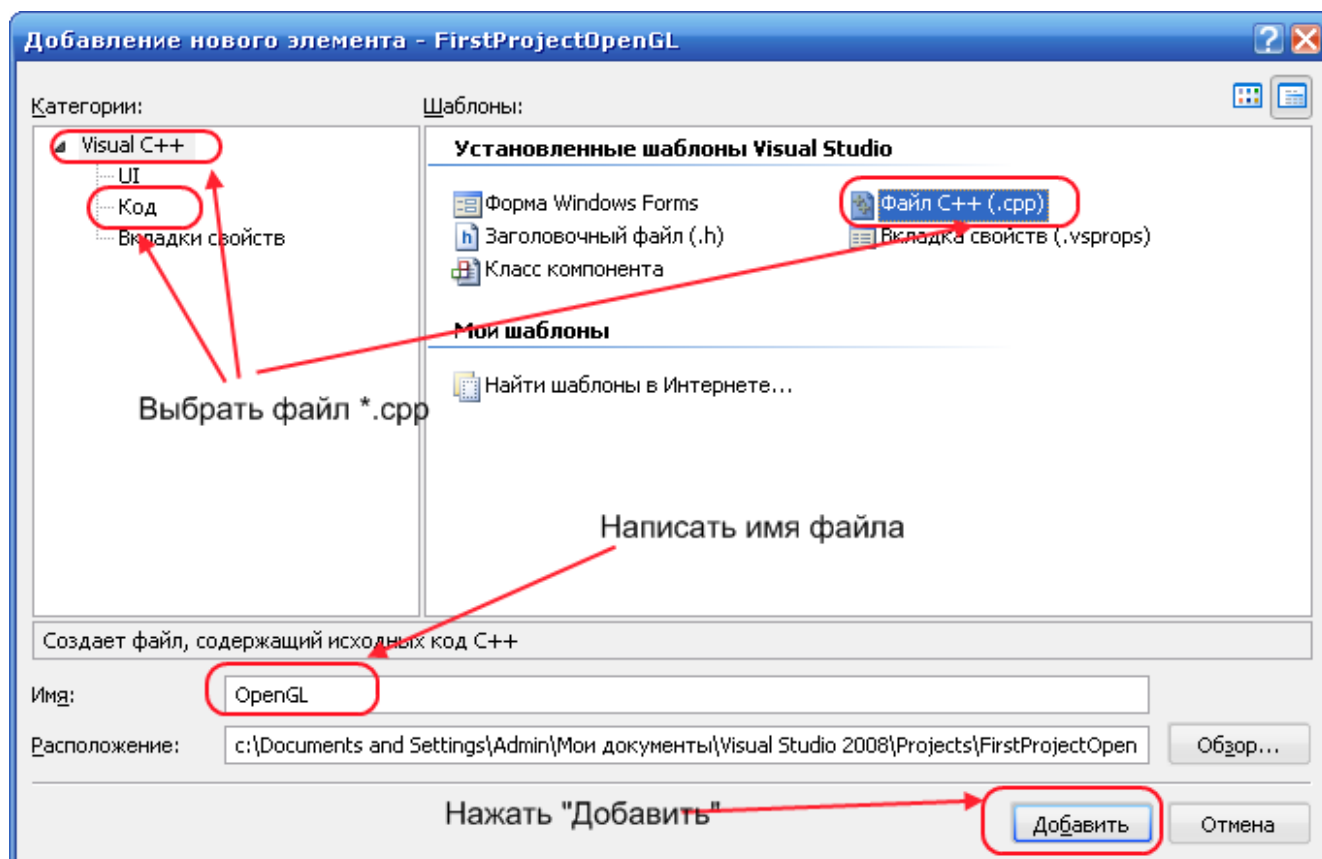


Рис. 6 : окно «Добавление нового элемента»

Далее необходимо настроить проект соответствующим образом. Для этого нужно добавить в проект информацию о используемых библиотеках, а так же изменить некоторые настройки проекта. Перейдём к настройке данных параметров выбрав пункт верхнего меню «Проект» и далее «Свойства: <имя Вашего проекта>», как показано на рисунке 7.

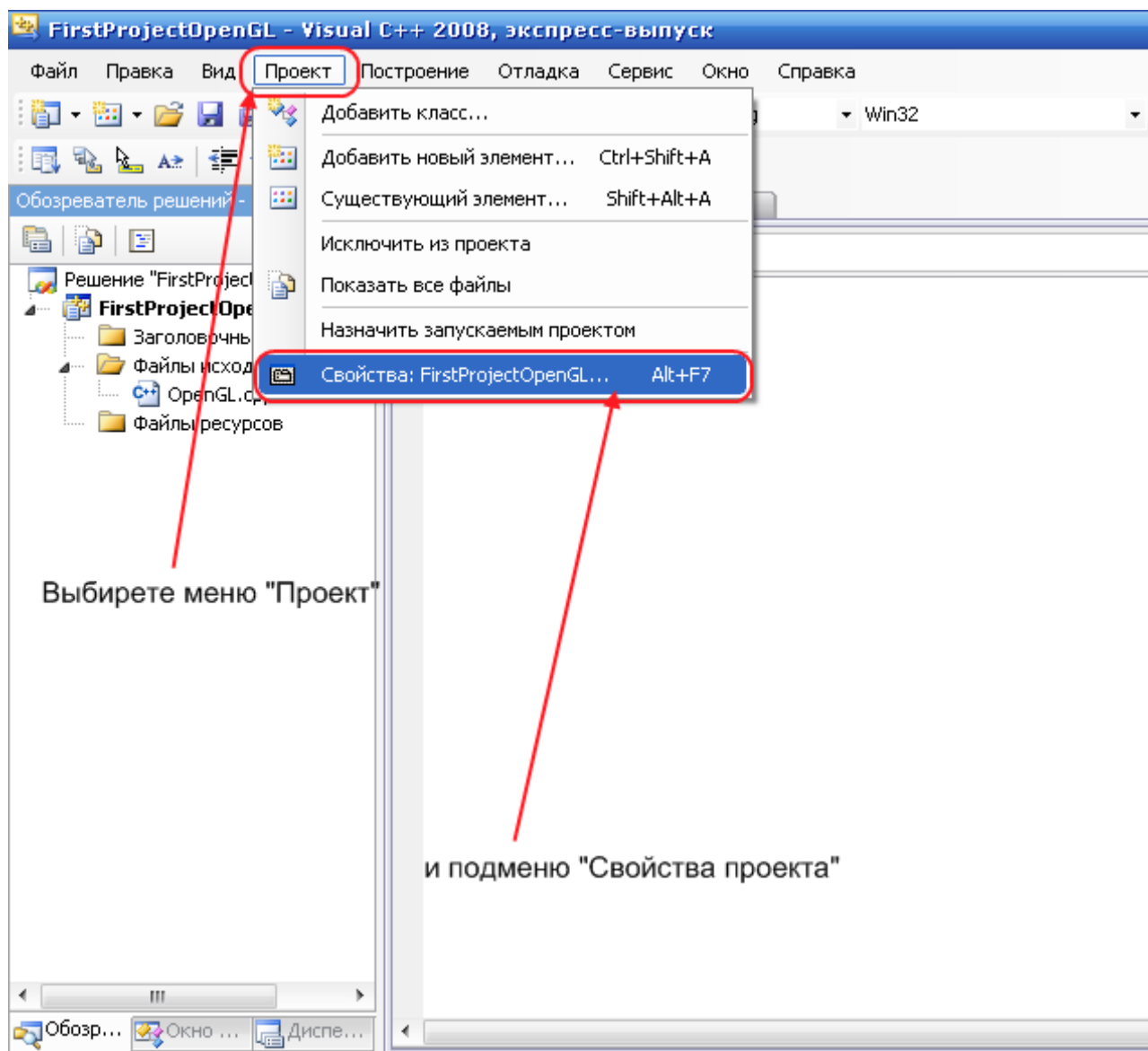


Рис. 7 : Переход к свойствам проекта.

В появившемся диалоговом окне из списка в левой части окна нужно выбрать «Свойства конфигурации», а в появившемся списке выбрать «Общие». В правой части окна нужно для свойства «Набор знаков» выбрать значение «Использовать многобайтовую кодировку». Нажимаем «Применить»

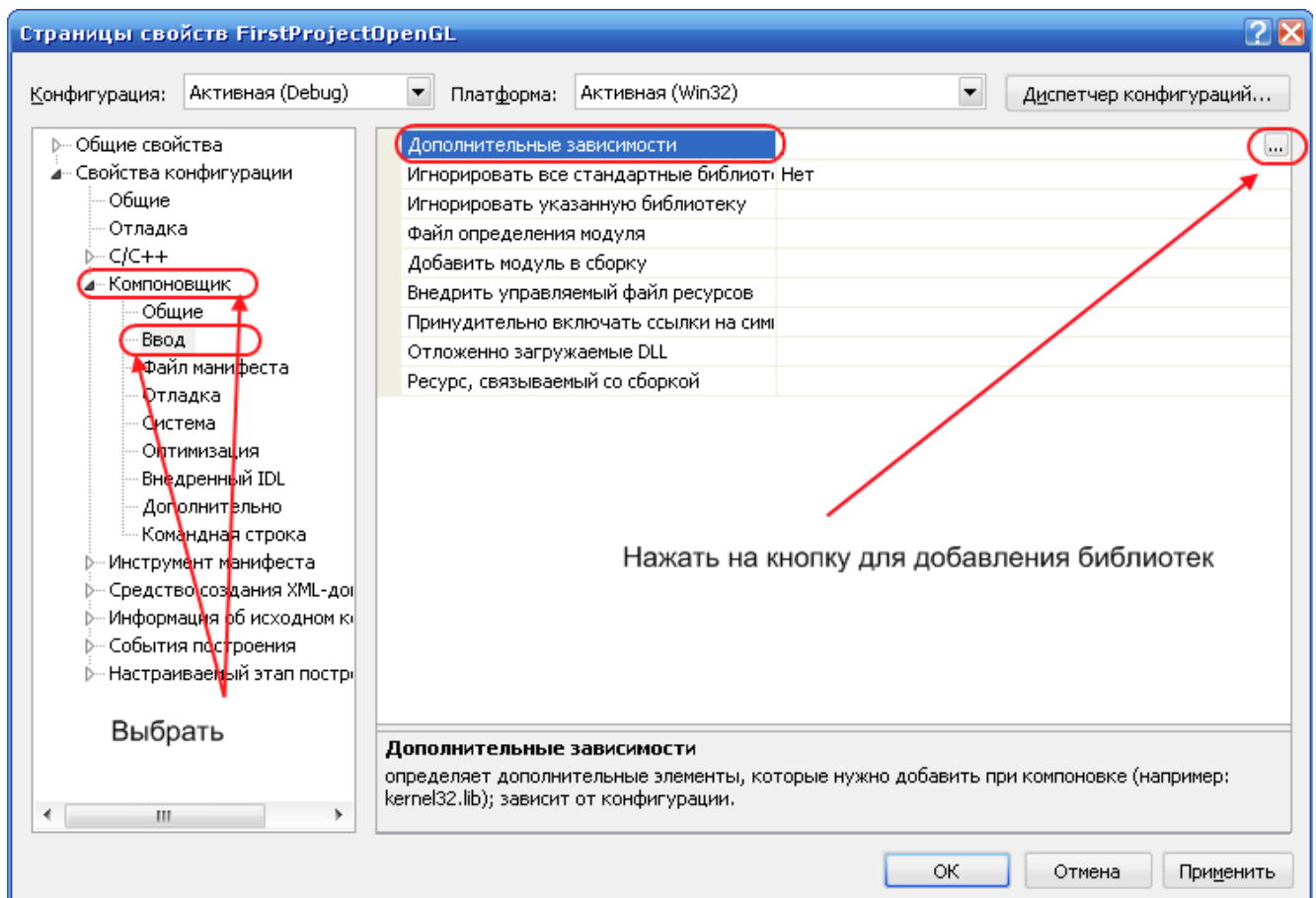


Рис. 9 : Вид окна свойства «Компоновщик» «Ввод»

В появившемся диалоговом окне необходимо вписать используемые библиотеки (opengl32.lib ; glu32.lib ; glaux.lib) в поле ввода в верхней части окна (рис. 10), и подтвердить ввод нажатием кнопки «Ок»

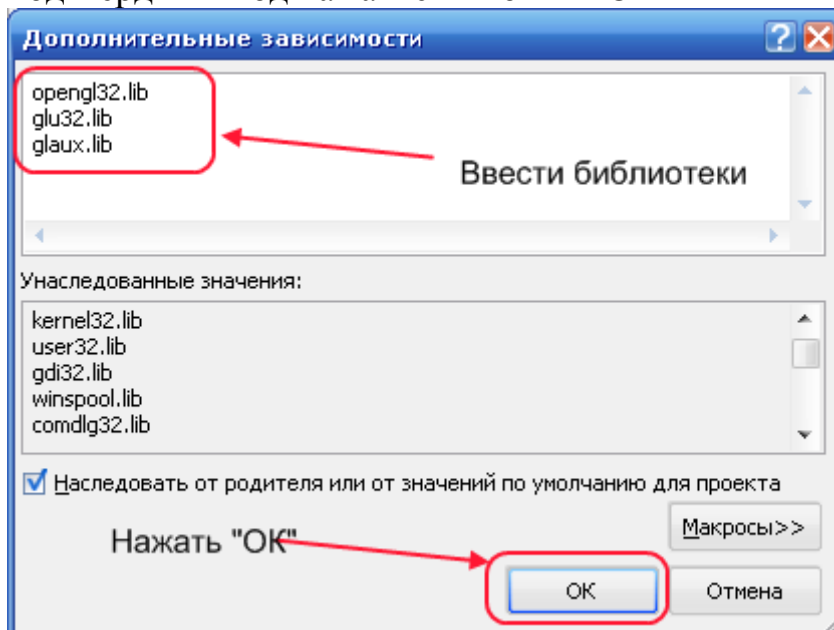


Рис. 10 : Добавление библиотек.

Далее Вы вернётесь к предыдущему диалоговому окну, в котором надо нажать кнопку «ОК»

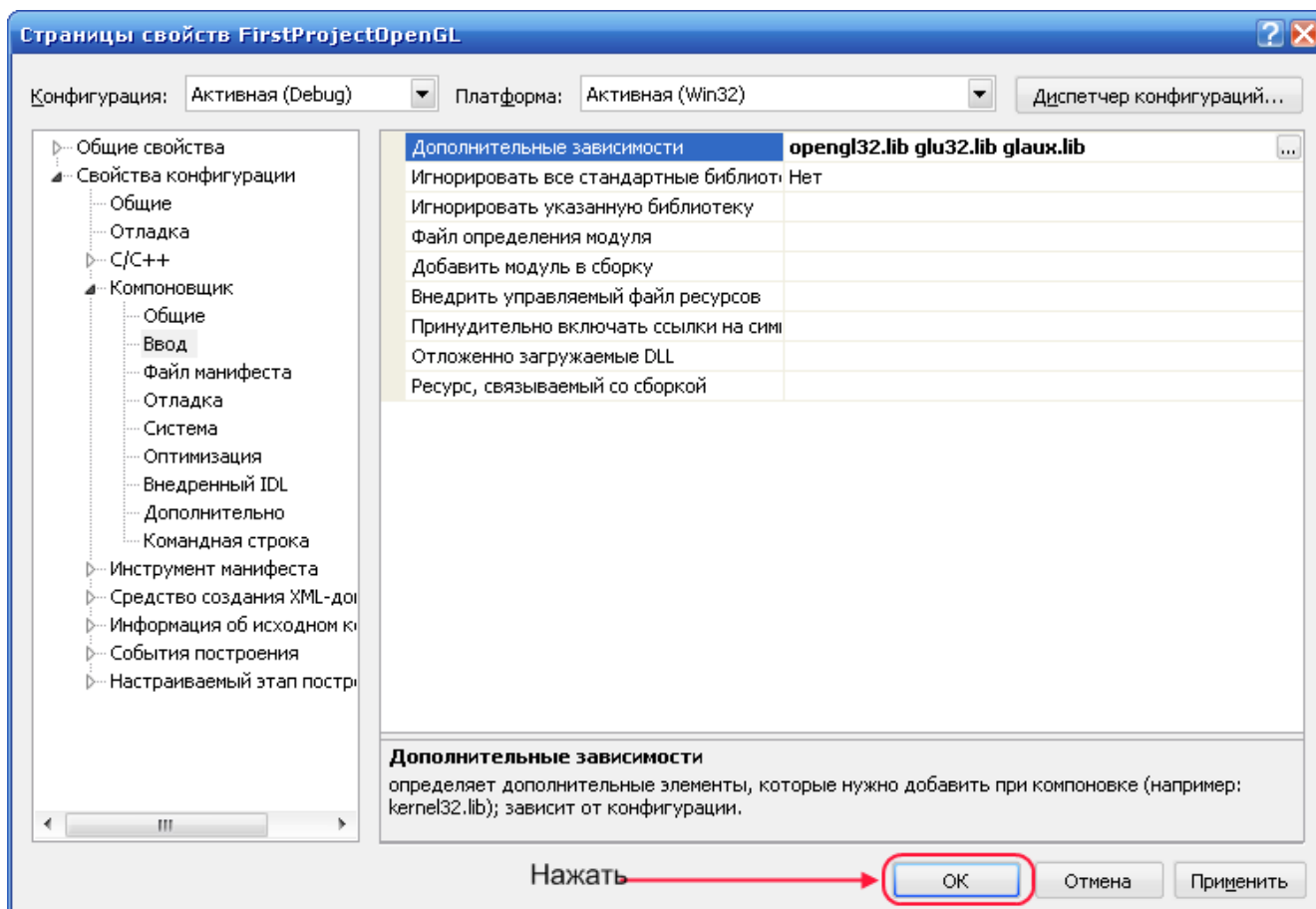


Рис. 11 : Добавление библиотек

На этом подготовительные работы по созданию проекта окончены. Далее в созданный вами файл исходного кода необходимо ввести текст программы и построить решение (создать исполняемый код приложения). Сделать это можно выбрав пункт верхнего меню «Построение» и в выпадающем меню выбрать «Построить решение», или же нажать кнопку F7.

Использование среды Среда Qt Creator

Qt Creator – кросс-платформенная среда для разработки приложений с двумерным и трехмерным графическим интерфейсом. Среда свободно распространяется вместе с вариантом библиотеки Qt с открытыми исходными текстами в различных вариантах для различных операционных систем.

Для выполнения лабораторных работ рекомендуется использовать вариант Qt Creator IDE for Windows MinGW, который содержит полный комплект

необходимых инструментов и библиотек вместе с компилятором C/C++ MinGW. Данный вариант содержит подробную документацию, учебные материалы и большое количество примеров программ, иллюстрирующих самые разные аспекты создания пользовательских интерфейсов программ, включая интеграцию Qt с вызовами трехмерной графической библиотеки Open GL.

После установки не требуется какой-либо настройки для использования библиотек и заголовочных файлов, необходимых для работы с Open GL

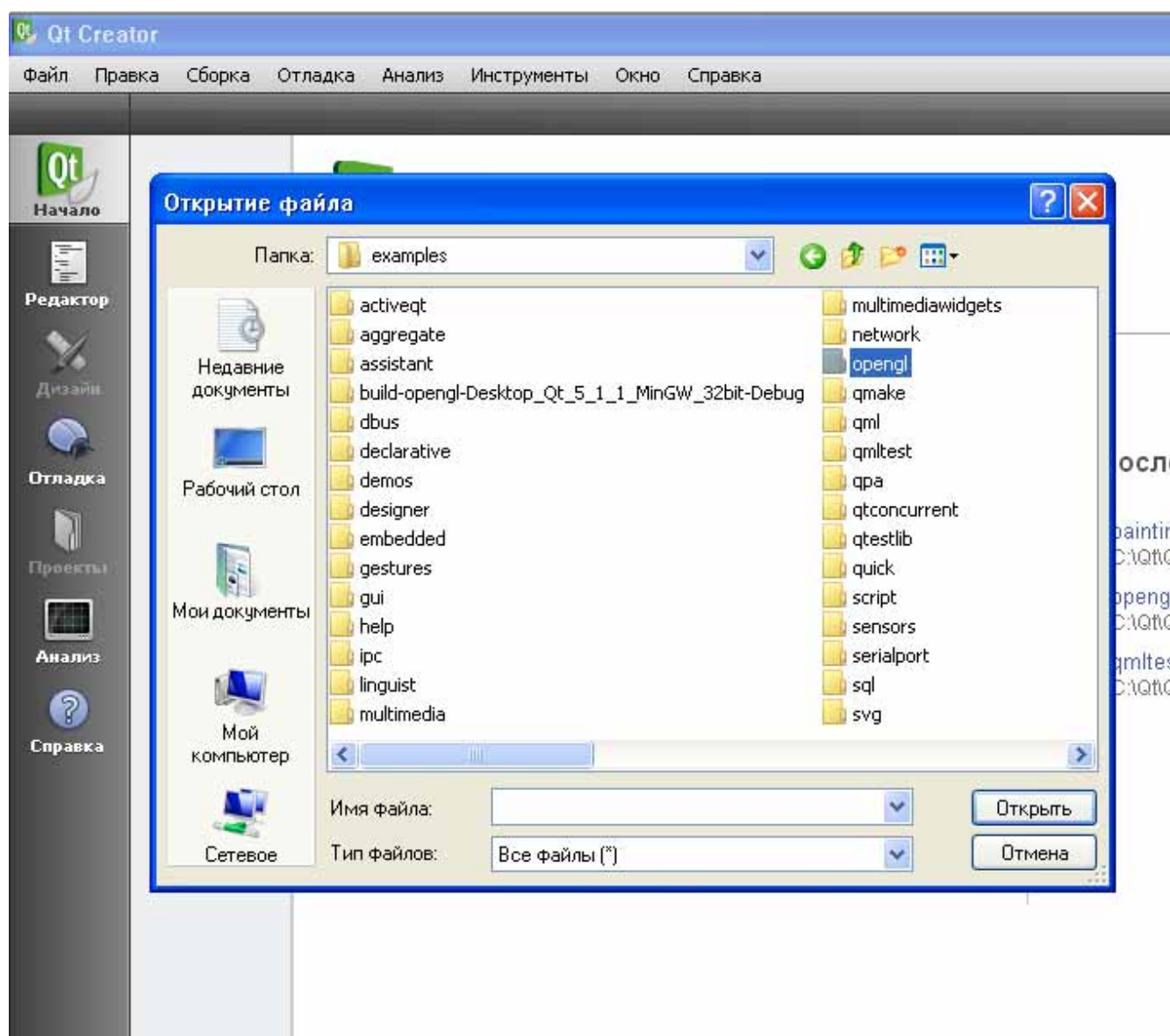


Рис. 12 : Общий каталог с примерами программ в среде Qt Creator

Qt\Qt*.**.*.*\mingw##_32\examples - общий каталог с примерами, отдельные подкаталоги которого иллюстрируют различные аспекты работы с Qt (рис. 12)

..* – версия библиотеки Qt

##_ версия компилятора MinGW

Примеры программ с Open GL (рис. 13, 14) находятся в каталоге

Qt\Qt*.**.*.*\mingw##_32\examples\OpenGL

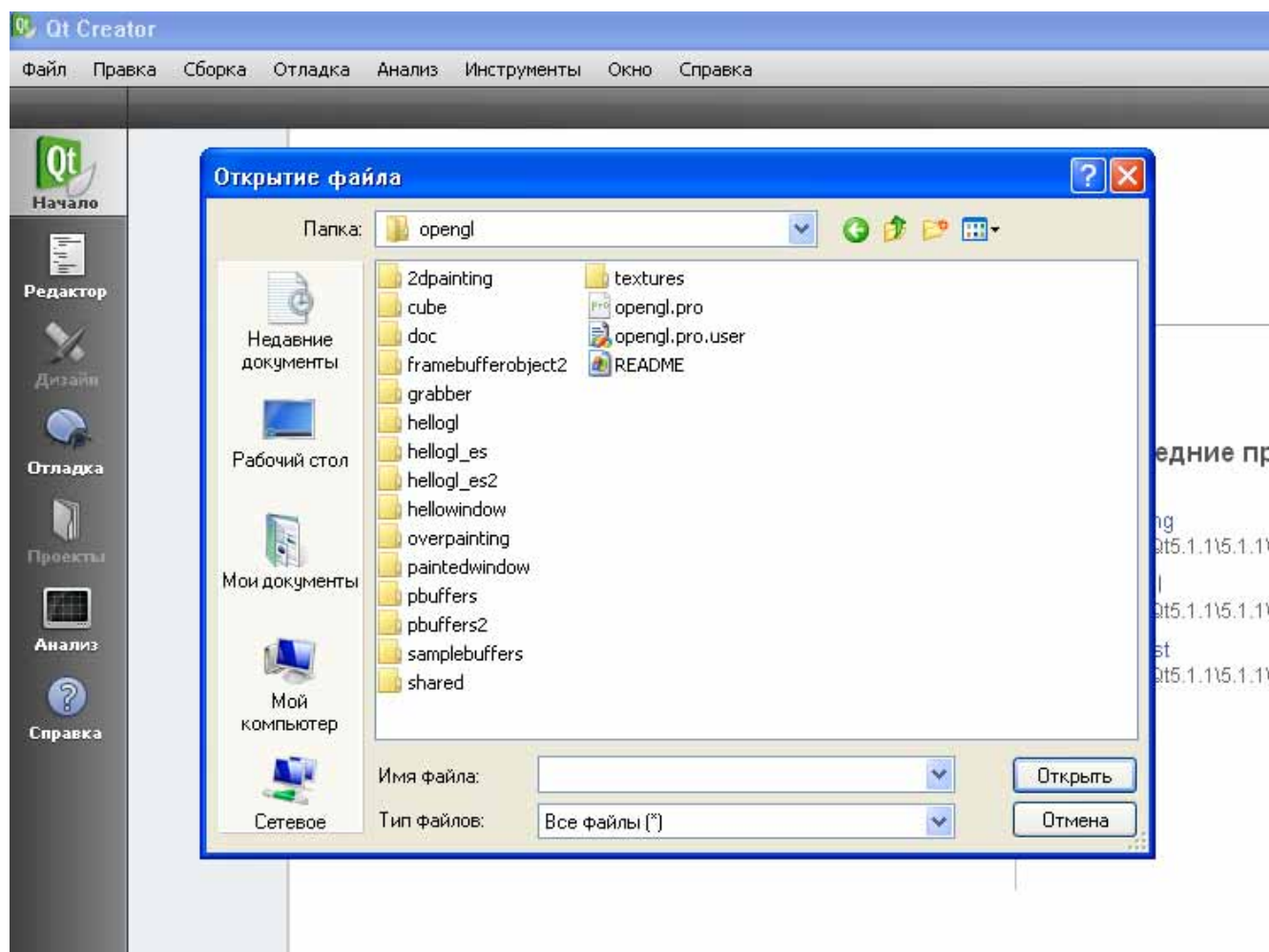


Рис. 13 : Каталог с примерами программ в среде Qt Creator с использованием вызовов Open GL

Qt Creator IDE и документацию можно найти на сайте <http://www.qt.io/developers/>

При работе в ОС Windows рекомендуется загрузить и установить свободно распространяемую среду разработки Qt Creator с библиотеками Qt 5.1.1 и Open GL и компилятором MinGW 4.9.2 со страницы

<http://www.qt.io/ru/download-open-source/#section-2>

(пункт меню Qt 5.1.1 for Windows 32-bit_ MinGW 4.9)

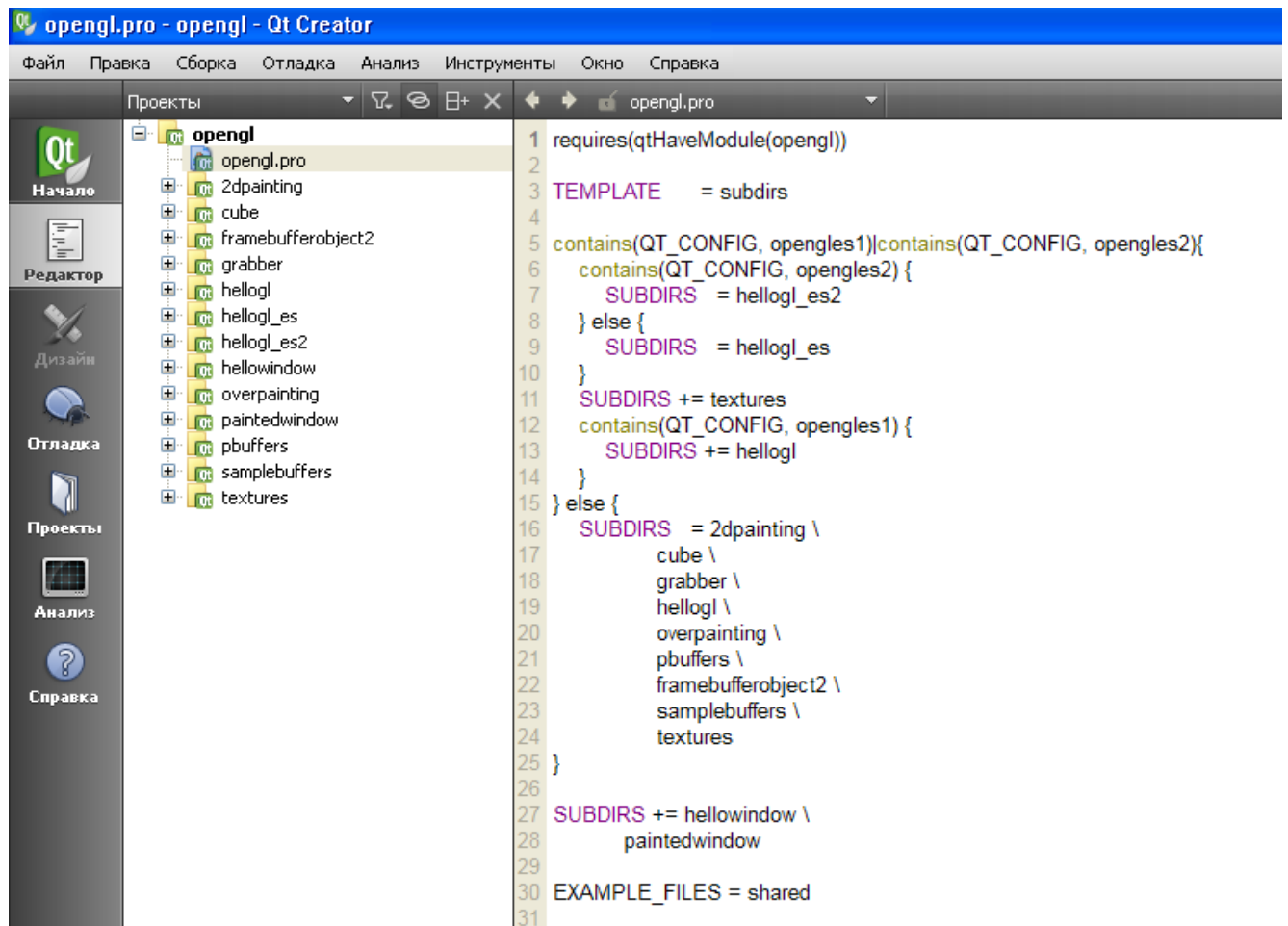


Рис. 14 : Примеры программ в среде Qt Creator с использованием вызовов Open GL в виде проекта Qt

Задания на лабораторные работы

Лабораторная работа №1

Освоение работы с библиотекой OpenGL

Создать приложение, выводящее объемный объект средствами Open GL.
Среда программирования – Visual C++ или QT Creator с MinGW и Open GL .

Лабораторная работа №2

Работа с источником света и свойствами материала поверхности.

Включить источник света, задать отражающие свойства поверхностей: диффузное, зеркальное отражение, цвет поверхности.

Вывести несколько объемных объектов. Каждый из них должен иметь различные свойства поверхностей (доминирующее зеркальное или диффузное отражение, различный цвет внешних поверхностей).

Лабораторная работа №3

Работа с источником света и свойствами источников и материалов.

Включить 3 источника света, задать отражающие свойства поверхностей, положение и цвет источников света. Рекомендуется цвет поверхностей сделать одинаковым, а отражающие свойства – разными.

Вывести несколько объемных объектов. Источники света должны иметь различный цвет. Необходимо организовать вращение сцены, управляемое с клавиатуры (сцена с объектами объекты вращается, источники неподвижны)

Лабораторная работа №4

Работа с буфером глубины, прозрачностью.

Вывести несколько пересекающихся объектов, как минимум один из них должен быть прозрачным. Включить 3 источника света, задать свойства поверхностей и источников. Организовать раздельное вращение источников света вокруг неподвижной сцены, управляемое с клавиатуры или мышью.

Лабораторная работа №5

Работа со списками, текстурой и NURBS-поверхностями.

Вывести сложный трехмерный объект, сохранить его в виде списка и размножить в виде нескольких копий различного масштаба на экране путем повторного вывода списка. Выводимые объекты располагаются над рельефом, задаваемым NURBS-поверхностью. Наложить текстуру на выводимые объекты, сцена должна быть освещена, необходимо предусмотреть возможность включения эффекта тумана (при помощи клавиатуры или мыши).

Лабораторная работа №6

Вывод трехмерных объектов с динамическим расчетом проекционных теней.

Вывести трехмерную сцену с движущимся объектом, который отбрасывает тень на другой объект (напр. плоскость). Тень должна перемещаться вместе с движением объекта исходя из взаимного положения источника света, объекта, который отбрасывает тень и объектов, на которые тень проецируется.

Лабораторная работа №7

Освоение работы с программируемыми шейдерами

Выполнить на выбор один из вариантов:

- Шейдер с сохранением данных в текстуре,
- Процедурный, текстурный шейдер,
- Традиционные шейдер,
- Фрагментный шейдер,
- Вершинный шейдер.

Примеры из пакета с компилятором Cg не допускаются

Лабораторная работа №8

Освоение Open Scene Graph

Вывести трехмерную сцену средствами библиотеки Open Scene Graph. В работе должны быть использованы только вызовы OSG.

Состав сцены (количество и внешний вид объектов создаваемой сцены) для каждого варианта уточняется с преподавателем.

Лабораторная работа №9

Разработка приложения с использованием Open Scene Graph

Вывести трехмерную динамическую сцену средствами библиотеки Open Scene Graph.

В работе должны быть использованы только вызовы OSG.

Изображение должно динамически меняться (перемещение одного или нескольких объектов относительно сцены с некоторой скоростью)

Состав сцены (количество и внешний вид объектов создаваемой сцены), динамика объектов для каждого варианта уточняется с преподавателем.

Требования к отчетам

Отчет о лабораторной работе должен содержать
титульный лист,
задание,
текст программы,
выводы

Отчет может предоставляться в электронном виде в виде файлов в формате doc, rtf, odf

Литература

- 1.Эйнджел Э. Интерактивная компьютерная графика. Вводный курс на базе OpenGL - М.: Вильямс, 2001. - 592 с
2. Хирн, Бейкер. Компьютерная графика и стандарт Open GL – М.: Вильямс, 2005, 1168 с.
3. Роджерс Д. Алгоритмические основы машинной графики- М.: Мир, 1989. - 512 с.
4. Земсков Ю.В. Qt 4 на примерах. - СПб.: БХВ - Петербург, 2008. - 608 с. : рис. + 1 эл. опт. диск (CD-ROM)
5. Порев Компьютерная графика, учебное пособие, – СПб.: ВHV-СПб, 2005, 432 с
6. Хилл OpenGL. Программирование компьютерной графики-СПб: Питер, 2002 – 1088 с

7. Краснов М. OpenGL. Графика в проектах Delphi – СПб.: BHV-СПб 2000, 352 с.;

8. Тихомиров OpenGL. Программирование трехмерной графики (2-е издание) (с дискетой) , – СПб.: BHV-СПб, 2002, 304 с.

9. Р. С.-мл. Райт, Бенджамин Липчак OpenGL : суперкнига. - 3-е изд. . : Вильямс, 2006- 1040 с.

10. М. Ву [и др.] OpenGL: руководство по программированию . - 4-е изд. - СПб. : ПИТЕР, 2006. - 624 с.