**Лабораторная работа 1(часть 2)**

**Этапы обработки исходного кода**

**Цель работы:** создание приложения на языке программирования С++ в интегрированной среде разработки Visual Studio 202х, исследование свойств проекта и его размещение на жестком диске.

**Введение.**

Создание приложения на языке программирования С++ в интегрированной среде разработки Visual Studio проходит в несколько этапов:

1. компиляция исходного кода – трансляция исходного кода, написанного на одном языке программирования, в исходный код на другом языке (в машинный код). В результате компиляции создается файл с расширением **obj** – объектный модуль программы.

В IDE Visual Studio компиляцию модуля с исходным кодом (файл с расширением cpp) можно осуществить, выбрав:

**Компилировать** в контекстном меню обозревателя решений для нужного файла

с помощью горячих клавиш Ctrl+F7

через пункт главного меню **Сборка** → **Компилировать**;

1. компоновка проекта – процесс связывания одного или совокупности объектных файлов и формирование на их основе загрузочного модуля (пункт меню **Сборка** → **Собрать решение**).

Выполнить полную сборку проекта можно, используя пункт меню   
**Сборка** → **Построить *имя\_проекта.***

Проект консольного приложения на языке С++ в Visual Studio, который при создании был назван HelloWorld, включает файлы и папки решения.

В папке решения **HelloWorld** (рисунок 1) хранятся папки и файлы, относящиеся к проекту:

* файл HelloWorld.sln – файл решения для созданной программы. Решение может состоять из нескольких проектов, в том числе на разных языках программирования;
* папка проекта HelloWorld;
* папка Debug.

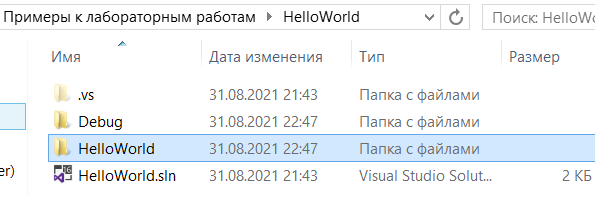


Рисунок 1. Файлы и папки проекта консольного приложения в интегрированной среде Visual Studio 2019.

В папке Debug:

* **HelloWorld.exe – исполняемый файл проекта;**
* HelloWorld.ilk – файл инкрементальной линковки (incremental linker), используемый компоновщиком для ускорения процесса повторной компоновки проекта;
* HelloWorld.pdb – отладочная информация/информация об именах в исполняемых файлах, используемая отладчиком.

В папке HelloWorld:

* **HelloWorld.cpp – файл исходного кода;**
* HelloWorld.vcproj – файл с настойками проекта;
* и другие.

В папке Debug проекта HelloWorld (рисунок 2):

* **Hello.obj – объектный модуль** (результат компиляции)**;**
* **HelloWorld.log – текстовый документ,** содержащий журнал построения.

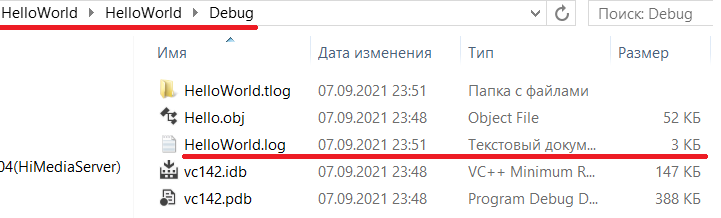
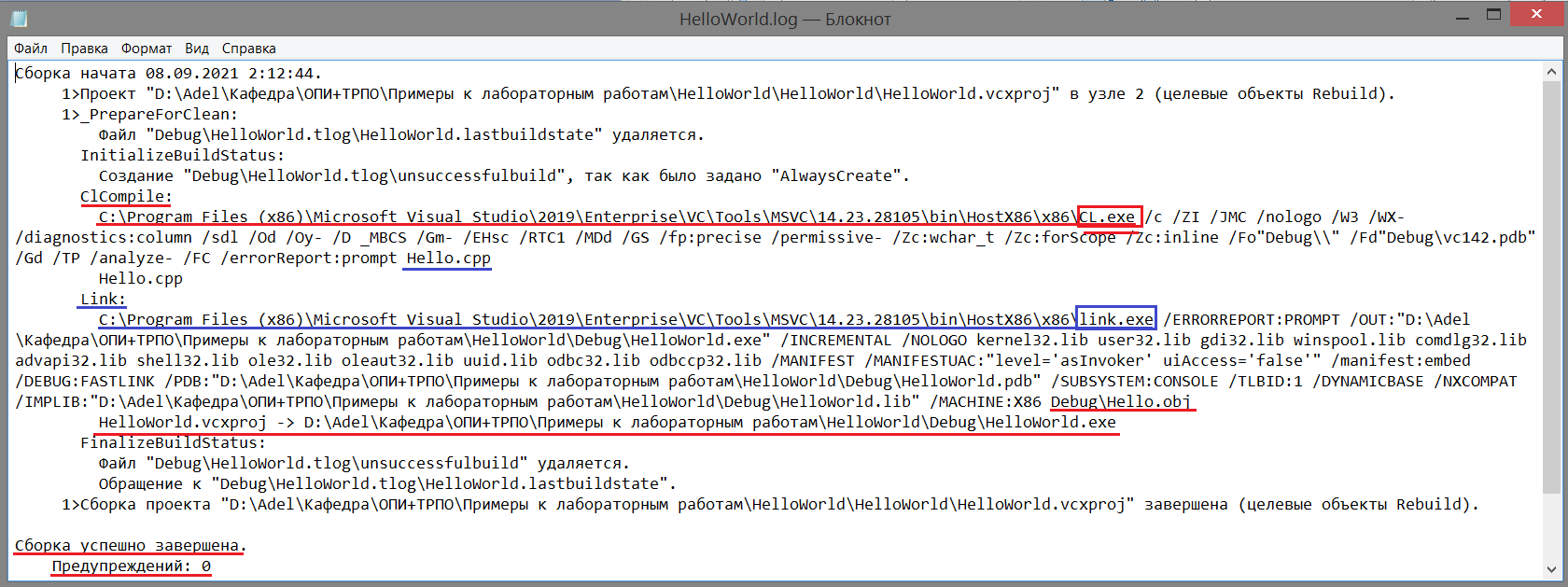


Рисунок 2. Файлы в папке проекта Debug.

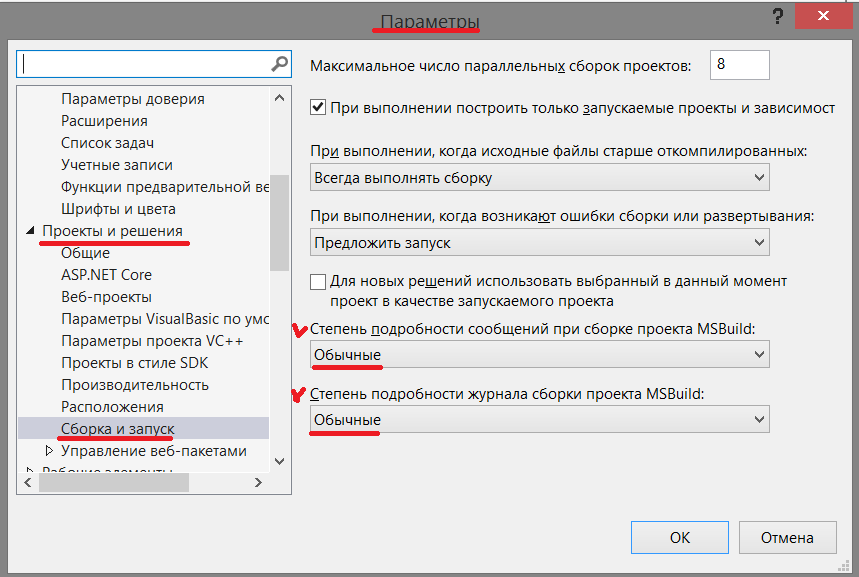
**Файл журнала построения**



В VS2017 и выше:

для изменения объема сведений, включаемых в журнал сборки необходимо:

* в меню ***Сервис*** (***Средства***) → ***Параметры***
* в разделе ***Проекты*** ***и решения*** выбрать ***Сборка и запуск***
* в списках Степень подробности сообщений и Степень подробности журнала сборки при построении проекта MSBuild выбрать ***Обычный***

****

**Иерархическая структура компонентов в Visual C++.**

Глобальный контейнером, то есть компонентом, включающим в себя другие компоненты, является Решение. Решение может содержать один или несколько проектов.

Проекты являются независимыми компонентами. Они имеют собственную структуру, состоящую из четырех основных каталогов:

***Внешние зависимости*** – содержит ссылки на все модули, которые использует программа.

***Файлы заголовков*** – содержит файлы кода С++ с расширением h.

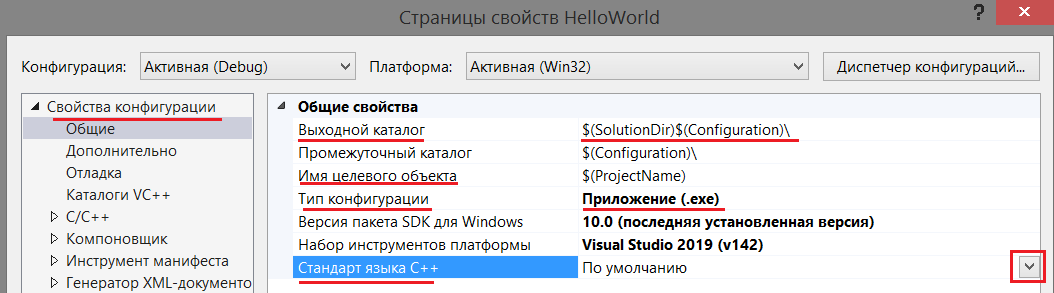
***Исходные файлы*** – содержит файлы кода С++ с расширением срр.

***Файлы ресурсов*** – содержит файлы, непосредственно не относящиеся к языку С++, но необходимые для работы приложения. Например, мультимедийные файлы.

Код программного проекта может иметь сложную структуру и состоять из нескольких файлов исходного кода и конфигурационных файлов.

**Страница свойств проекта.**

Раздел ***конфигурация***:



В **Общие** определен тип проекта (Приложение); отображается текущая версия VS; стандарт языка. Изменить свойства можно, нажав кнопку 

Проект может находиться в конфигурации **Debug** или **Release**.

Текущая конфигурация и целевая платформа, на которой разрабатывается проект, отображается в верхней части окна. Изменить текущие настройки можно в **Диспетчере конфигураций**.

В разделе **C/C++** отображаются текущие настройки C/C++. Некоторые из них:

**Общие** – ключ **Уровень предупреждений** позволяет отключить все предупреждения (/W0), либо ужесточить уровень проверок и считать все предупреждения ошибками (/Wall).

**Предварительно откомпилированные заголовки** позволяют их включить/отключить, определить имя создаваемого предварительно откомпилированного заголовочного файла и местоположение для полученного выходного файла (с расширением pch).

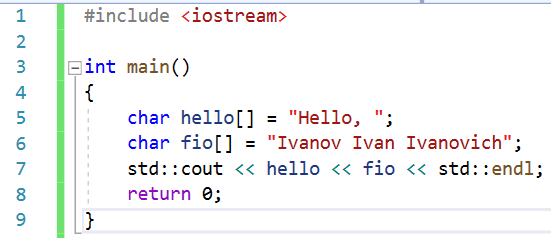
Также настроить имена и папки, в которых будут размещаться различные **Выходные файлы**.

**Командная строка** отображает, с какими параметрами (ключами) выполняется текущая компиляция.

Раздел **Компоновщик** отображает и позволяет изменить текущие настройки и ключи компоновки.

**Задание**

1. Изучите основные возможности интегрированной среды разработки Visual Studio.
2. В решение **HelloWorld**, созданное в лабораторной работе 1,добавьте новый проект с именем Lab\_02. Для создания проекта воспользуйтесь шаблоном ***Пустой проект***.
3. В обозревателе решений в папку проекта ***Исходные файлы*** добавьте новый элемент с именем lab02.
4. Введите исходный текст программы:



В 6 строке запишите вашу фамилию, имя, отчество на английском языке.

1. Используйте контекстно-зависимое меню проекта и назначьте его автозапускаемым.
2. Выполните приложение и убедитесь в его работоспособности.
3. Скомпилируйте текст программы. Найдите в папке проекта созданный объектный модуль. В какой папке он находится, время создания/изменения, его размер?
4. Внесите изменения в исходный текст программы, чтобы в нем содержались ошибки. Посмотрите, как система программирования сообщает об ошибках.
5. Выполните сборку проекта. После успешной сборки найдите в папке проекта исполняемый модуль.
6. Перестройте проект. Проанализируйте директории проекта. Какие изменения произошли? В поддиректории **Debug** найдите файлы с расширением **obj.**
7. В контекстном меню обозревателя решений выберите «Перестроить» для активного проекта. Поясните реакцию системы программирования.
8. В директориях проекта найдите файл с расширением **log** (файл журнала построения). Проанализируйте и объясните его содержимое.
9. Установите конфигурацию проекта Release и снова выполните полную сборку проекта. Откройте папку решения. Какие изменения в ней произошли? Сравните размер отладочной и конечной версии исполняемого модуля. Объясните их различия.

**Вопросы:**

* перечислите расширения исходных файлов проекта С++;
* охарактеризуйте файловую структуру решения;
* назовите этапы обработки исходного кода программы;
* как назначить проект автозапускаемым?
* в результате какого этапа образуются файлы с расширением **obj**? Что в этих файлах содержится?
* в результате какого этапа образуются файл с расширением **exe**?
* какая информация содержится в log-файле проекта?
* как получить доступ к свойствам проекта?
* поясните назначение страниц свойств проекта;
* поясните, в чем отличие конфигурации Debug от Release.