



Учебная практика 2016-2017

ОСНОВЫ C++

День 1

Часть 0

- **Создание проекта**
- **Переключение на русский язык**

Часть 1

- Стандарты C++
- Общие сведения
- Структура простой программы
- Простые типы данных
- Основные операции и операторы

Часть 2

- Массивы, указатели, ссылки



Начальные сведения

СОЗДАНИЕ КОНСОЛЬНОГО ПРИЛОЖЕНИЯ



Создание проекта

Проекты в компьютерных классах – на диске D

Создайте папку «Practice2017», в ней – свою папку (ФИО), далее – «Day1»

VisualStudio 2013 (2015)

Создание нового проекта

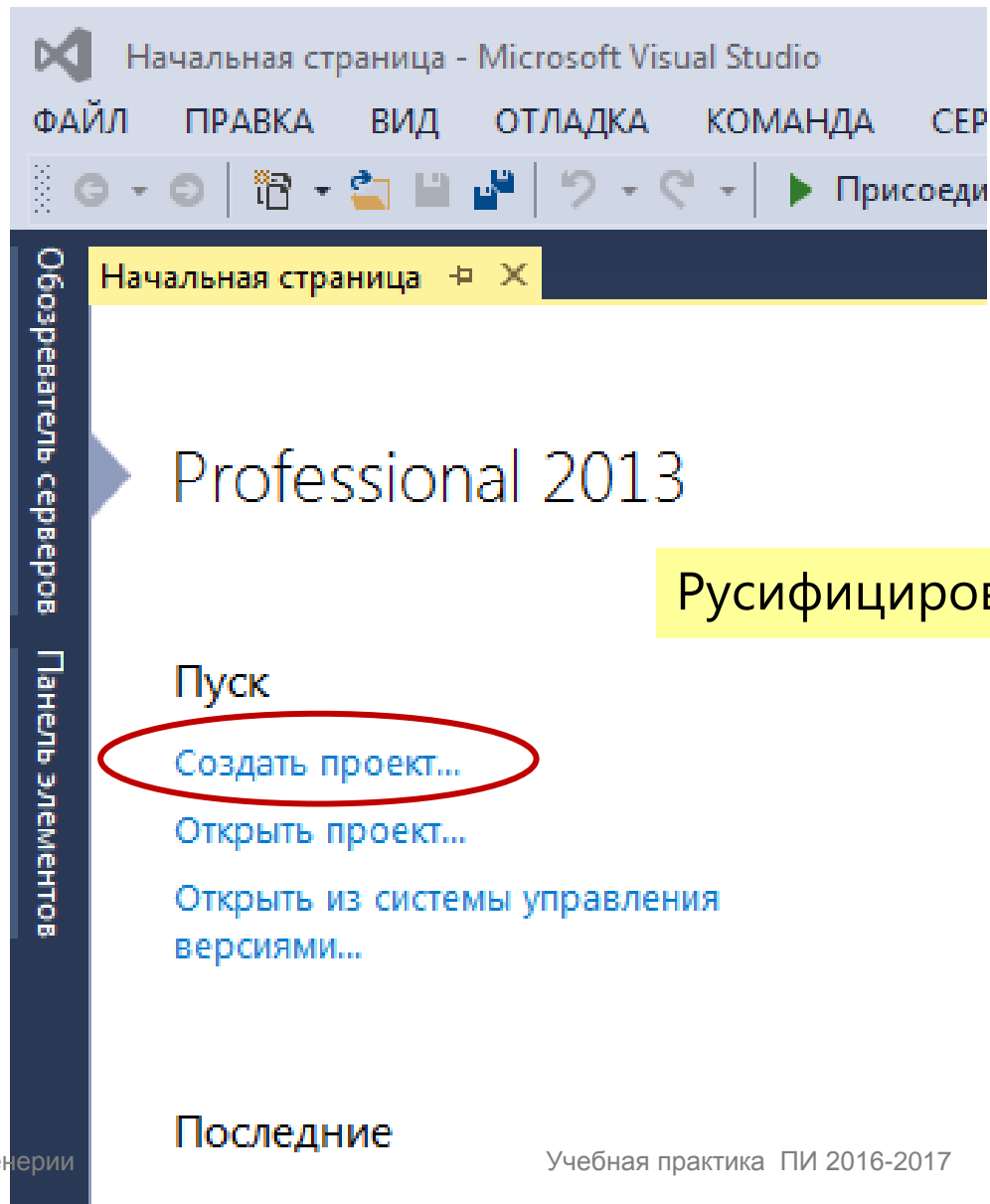
- Create Project
- Выбираем C++, General, Empty Project
- Выбираем папку проекта
- Вводим имя
- Создается пустой проект

Добавляем файл – источник (код)

- Правой кнопкой – на Source Files
- Add -> New Item
- C++ File
- Вводим имя



Начало работы





Создание проекта

New Project

Recent Templates

Installed Templates

- Visual C#
- Other Languages
 - Visual Basic
 - Visual C++
 - ATL
 - CLR
 - General
 - MFC
 - Test
 - Win32
 - Visual F#
- Other Project Types
- Database
- Test Projects

Online Templates

.NET Framework 4 Sort by: Default

Search Installed Templates

| | | |
|--|------------------|------------|
| | Empty Project | Visual C++ |
| | Custom Wizard | Visual C++ |
| | Makefile Project | Visual C++ |

Type: Visual C++
An empty project for creating a local application

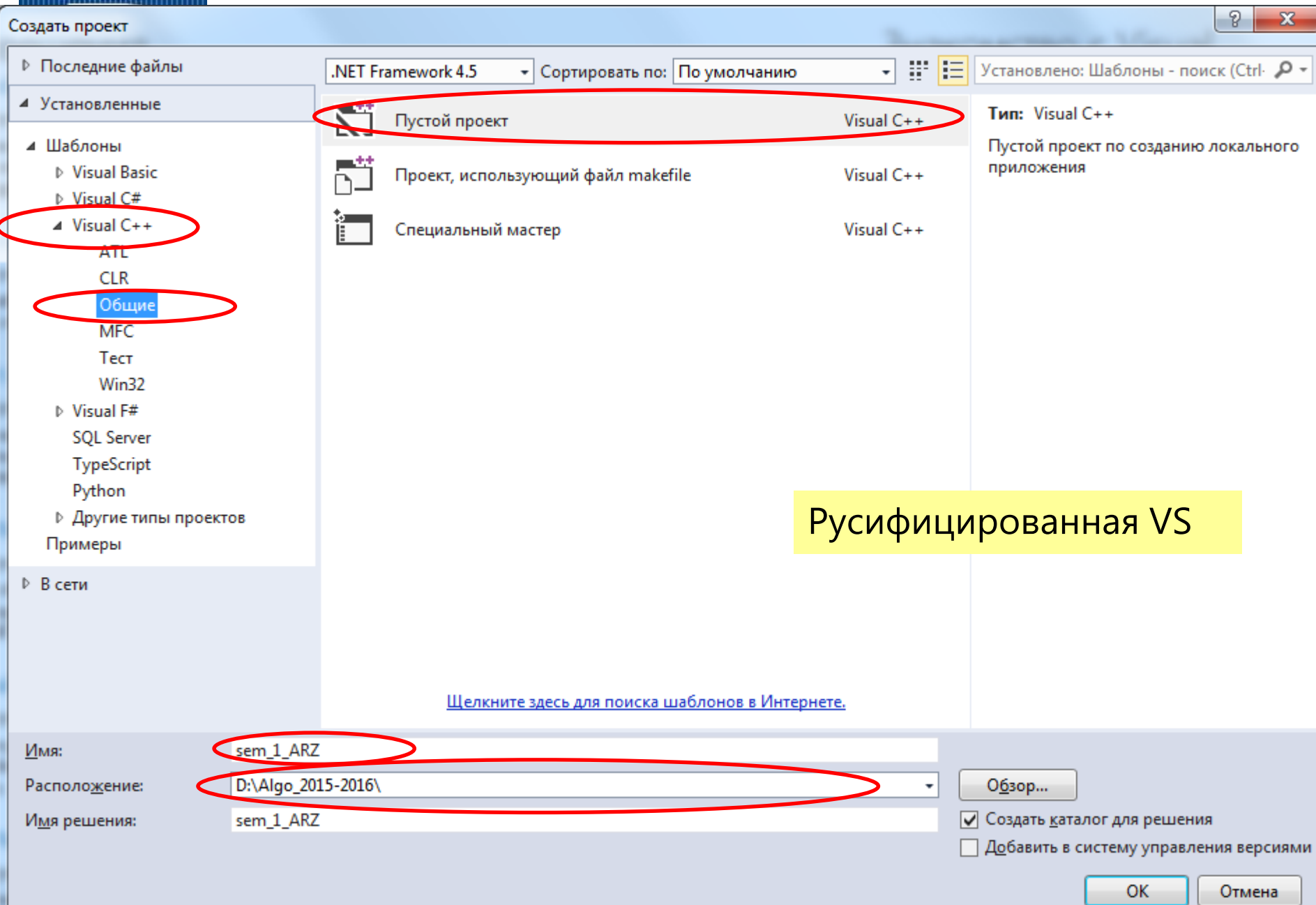
Name: Seminar 1

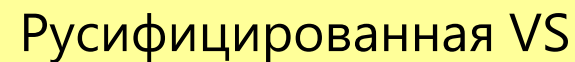
Location: D:\HSE\Ахметсафина\Учебные материалы\07 Построение и анализ алгоритмов\2013-2014\Р Browse...

Solution name: Seminar 1

☒ Create directory for solution
☐ Add to source control

OK Cancel



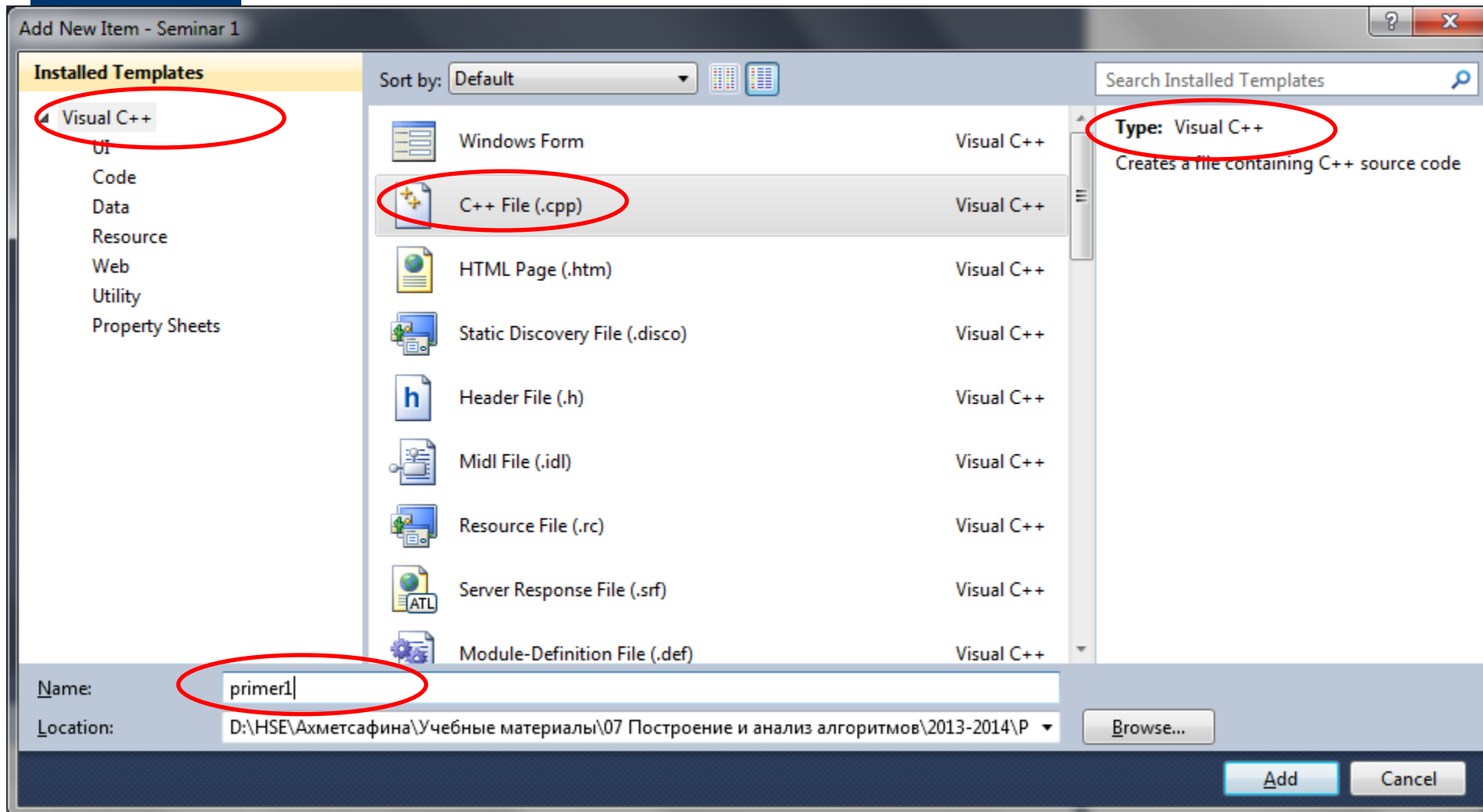


ФКН, Департамент Программной инженерии

Учебная практика ПИ 2016-2017

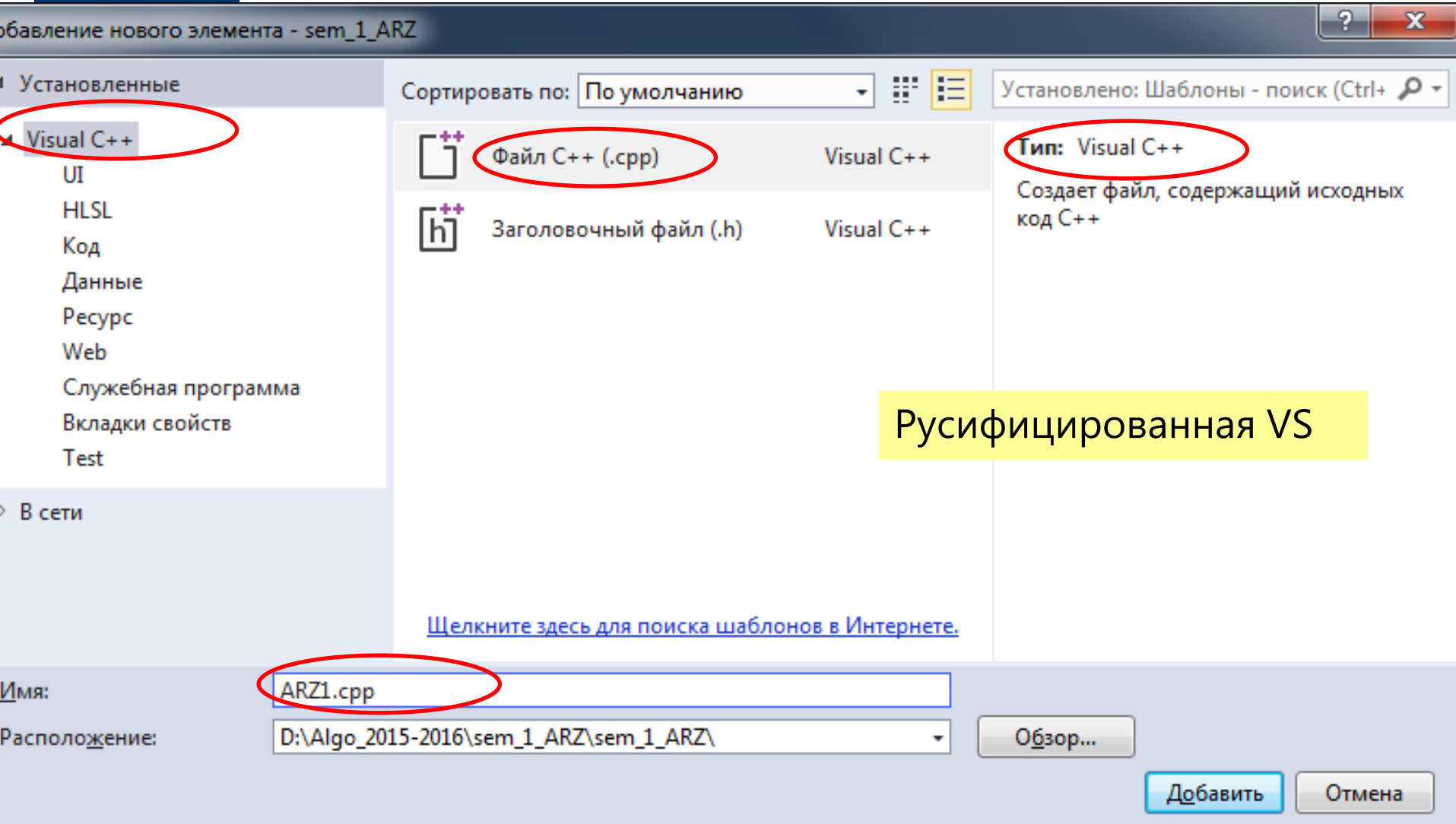


Добавить файл-источник



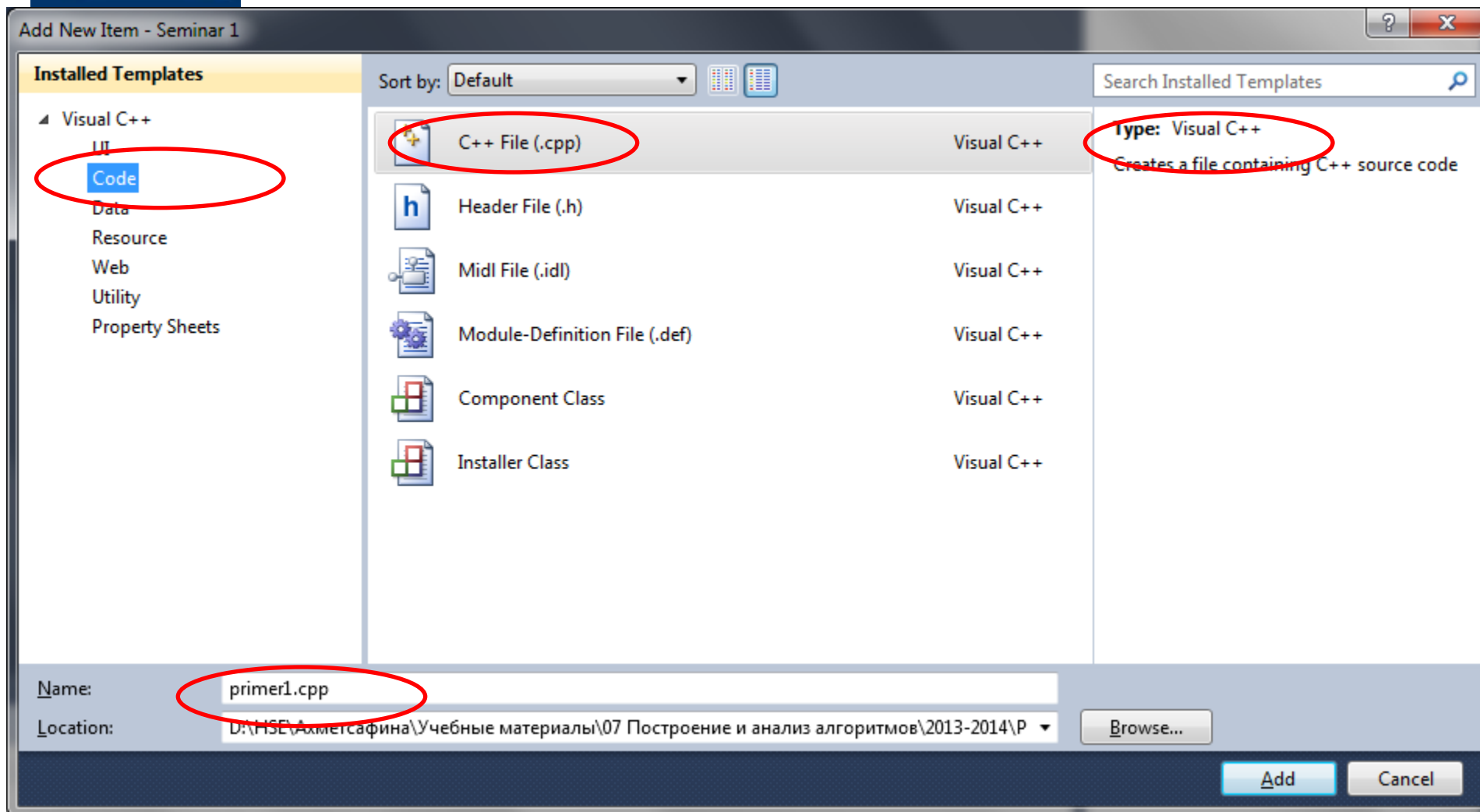


Добавить файл-источник



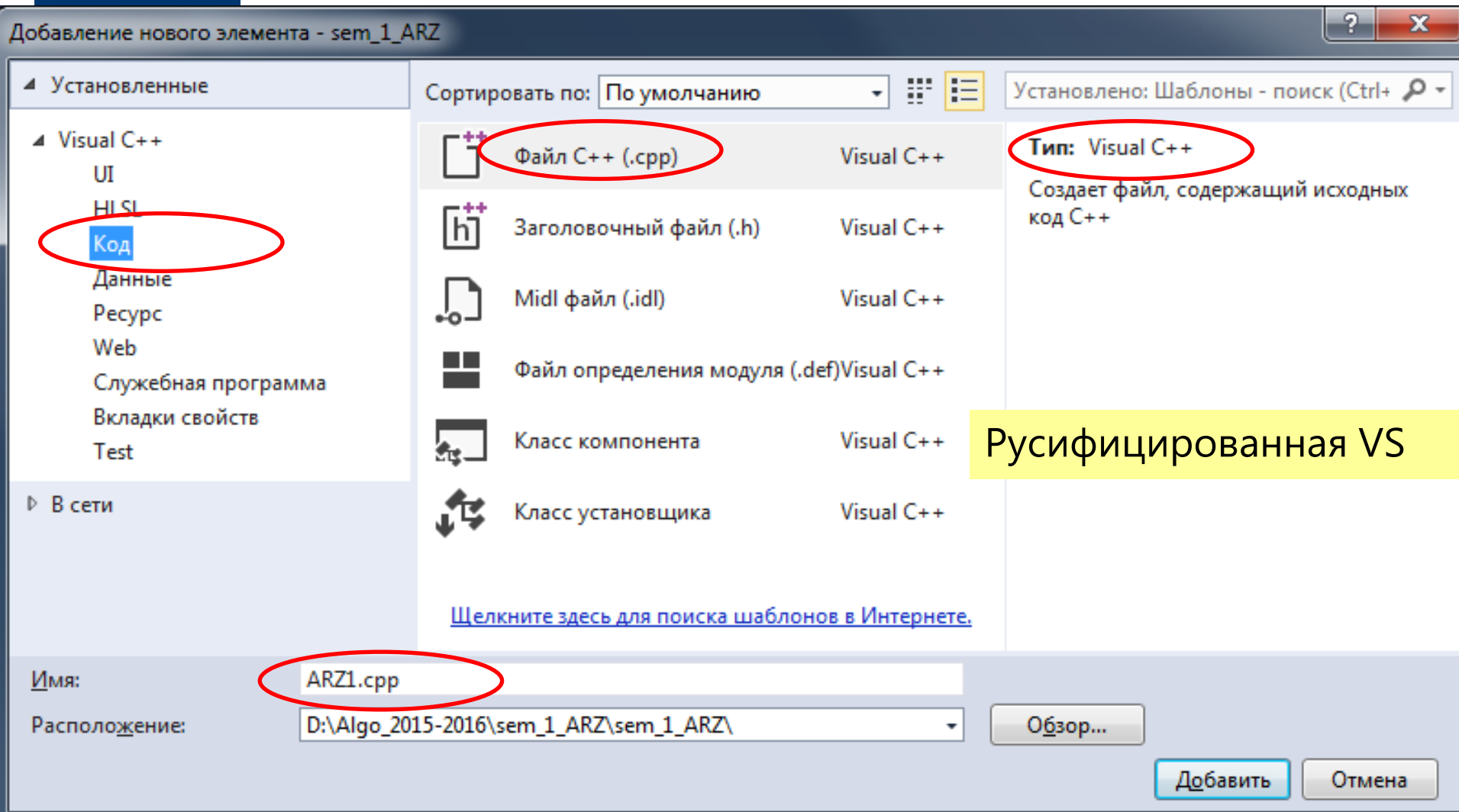


Добавить файл-источник





Добавить файл-источник





Начальные сведения

ОЧЕНЬ ПРОСТОЙ ПРИМЕР ПРОГРАММЫ



Пример программы на C++

```
1.#include <iostream>
2.using namespace std;
3.void main()
4.{
5.    int a, b;
6.    cout << "Hello World!" << endl;
7.    cout << "Enter two numbers" << '\n';
8.    cin >> a >> b;
9.    cout << "sum:" << a+b << endl;
10.    system("pause");
11.}
```



Переключаем на русский_1

Переключение локали
Прототип:

```
char * setlocale( int category, const char * locale );
```

Локаль — это набор параметров, включая набор символов, язык пользователя, страну, часовой пояс, а также другие предустановки, которые пользователь ожидает видеть в пользовательском интерфейсе.



Переключаем на русский_2

```
char * setlocale( int category, const char * locale );
```

Параметр `category` может быть установлен в

`LC_ALL` изменение всех параметров локали

`LC_CTYPE` Влияет на поведение всех функций заголовочного файла `<cctype>`, кроме `isdigit` и `isxdigit` и на функции обработки многобайтовых и широких символов.

Параметр `locale`

`"C"` - Минимальная Си-локаль.

`""` - Локаль используемая средой разработки по умолчанию

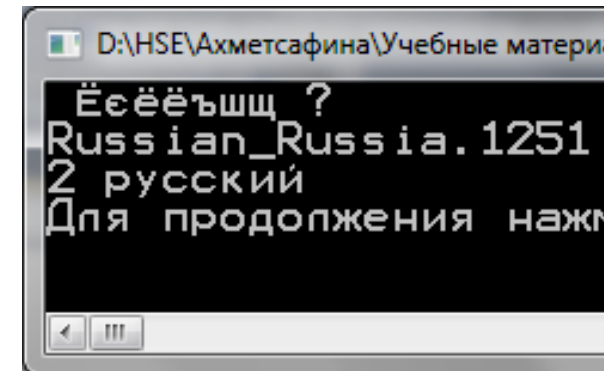
`Null` - возвращает текущую локаль

`"Russian"`



Переключаем на русский_3

```
1. #include <iostream>
2. using namespace std;
3. void main()
4. {
5.     cout<<" русский ? \n";
6.     cout<<setlocale(LC_CTYPE, "")<<' \n
7.     setlocale(LC_ALL, "Russian");
8.
9.     cout<<"2 русский \n";
10.    system("pause");
11. }
```





Какие файлы фактически войдут в проект, зависит от типа проекта и параметров, выбранных при работе с мастером.

ФАЙЛЫ РЕШЕНИЯ И ПРОЕКТА



Project (Проект)

С логической точки зрения проект содержит все файлы исходного кода, значки, изображения, файлы данных и прочие элементы, которые будут скомпилированы в исполняемую программу, а также все остальное, что необходимо для выполнения компиляции.

Проект также содержит все параметры компилятора и другие файлы конфигурации, которые могут потребоваться разным службам или компонентам, с которыми программа будет взаимодействовать.

В Visual Studio файл проекта используется обозревателем решений для отображения содержимого и параметров проекта.

При компиляции проекта подсистема MSBuild использует файл проекта для создания исполняемого файла.

Можно настроить проекты для создания выходных данных другого типа.



Solution (Решение)

С точки зрения логики и файловой системы проект содержится в рамках *решения (solution)*, которое может содержать один или несколько проектов вместе с информацией о сборке, параметрами окна Visual Studio и любыми прочими файлами, которые не связаны с каким-либо проектом.

Фактически решение является текстовым файлом в собственном уникальном формате; его обычно не изменяют вручную.

Решение включает связанный SUO-файл, в который входят параметры, предпочтения и сведения о конфигурации для каждого пользователя, работавшего над проектом.



Решение и проект

Создание проекта Visual C++:

- новое решение (solution)
- новый проект (project), добавляемый к решению (solution).

Часто приложения разрабатываются как решения, содержащие несколько проектов.

Обычно выходным файлом проекта является EXE- или DLL-файл.

Проекты могут зависеть друг от друга; в процессе сборки среда разработки Visual C++ проверяет зависимости как внутри проектов, так и между проектами.



Решение и проект

Каждый проект имеет основной исходный код *.cpp.

В зависимости от вида проекта, он может содержать другие файлы, например, заголовочные *.h

Указанием на содержимое этих файлов являются их расширения. По расширениям файлов определяется способ обработки их содержимого в ходе построения.



Файлы проекта и решения

| имя_файла | Описание |
|----------------------------|---|
| <i>Имя_решения.sln</i> | Файл <i>решения</i> . Используется для организации всех элементов проекта или нескольких проектов в единое решение. |
| <i>Имя_проекта.suo</i> | Файл <i>параметров решения</i> . Сохраняет настройки решения, чтобы при любом открытии проекта или файла в решении оно выглядело и вело себя необходимым образом. |
| <i>Имя_проекта.vcxproj</i> | Файл <i>проекта</i> . Хранит информацию, относящуюся к каждому проекту. (В более ранних версиях этот файл назывался <i>Имя_проекта.vcproj</i> или <i>Имя_проекта.dsp</i> .) |
| <i>Имя_проекта.sdf</i> | Файл <i>базы просмотра</i> . Поддерживает функции просмотра и навигации, такие как Перейти к определению , Найти все ссылки и Представление классов . Создается путем анализа файлов заголовков. Очень большой файл!!! |



Файлы проекта и решения

| Имя_файла | Описание |
|------------------------------------|---|
| <i>Имя_проекта.vcxproj.filters</i> | Файл <i>фильтров</i> . Указывает, куда поместить файл, который добавляется в решение. Например, H-файл помещается в узле Файлы заголовков . |
| <i>Имя_проекта.vcxproj.user</i> | Файл <i>пользователя миграции</i> . После миграции проекта из Visual Studio 2008 в этом файле появляются данные, преобразованные из любых VSPROPS-файлов. |



Файлы проекта и решения

| Имя_файла | Описание |
|------------------------|---|
| <i>Имя_проекта.idl</i> | (Для конкретных проектов) Содержит исходный код на языке описания интерфейсов (IDL) для библиотеки типов элементов управления. Используется Visual C++ для создания библиотеки типов. Созданная библиотека предоставляет доступ к интерфейсу элемента управления другим клиентам автоматизации. |
| Readme.txt | Файл <i>сведений</i> . Создается мастером приложений и описывает файлы в проекте. |



Файлы проекта и решения

| Имя_файла | Описание |
|-----------|---|
| *.cpp | Исходный код на C++ |
| *.h | Заголовочный файл |
| *.obj | Объектные файлы — скомпилированные, но не скомпонованные. |
| *.map | Текстовый файл, содержащий информацию для компоновщика. |

