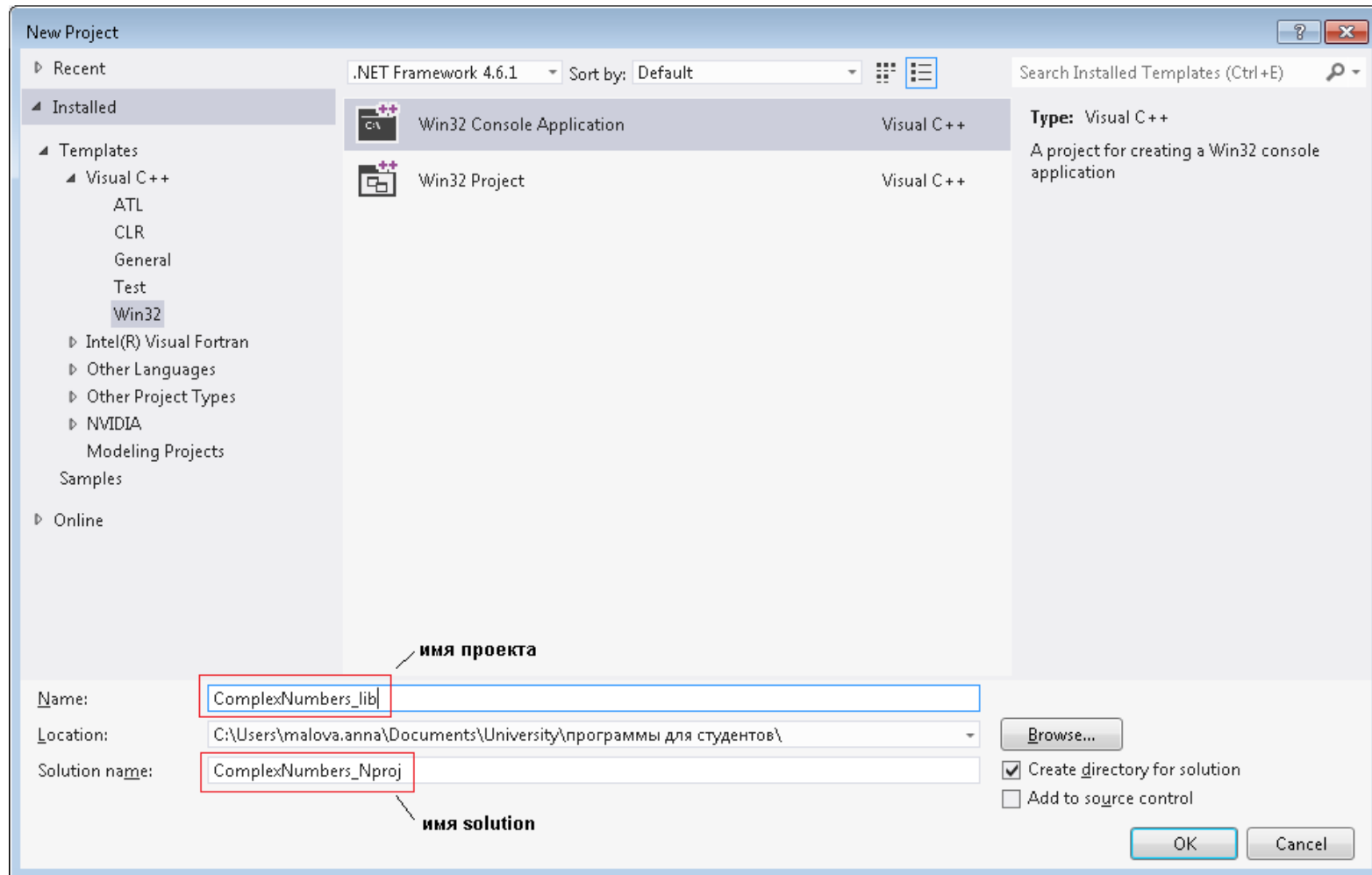


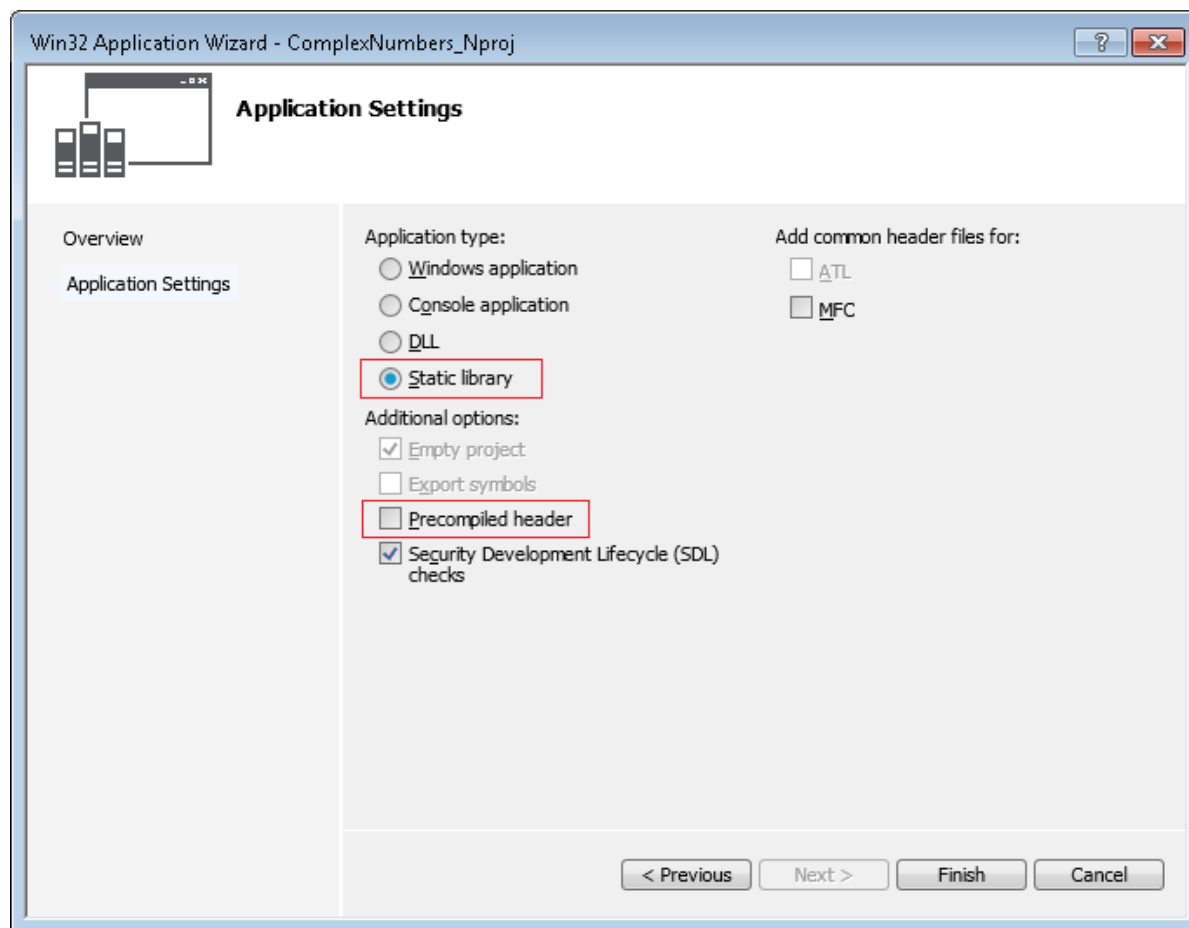
## Как создать solution в VS из нескольких проектов

Создадим программу, разбитую на 2 проекта: библиотеку и приложение.

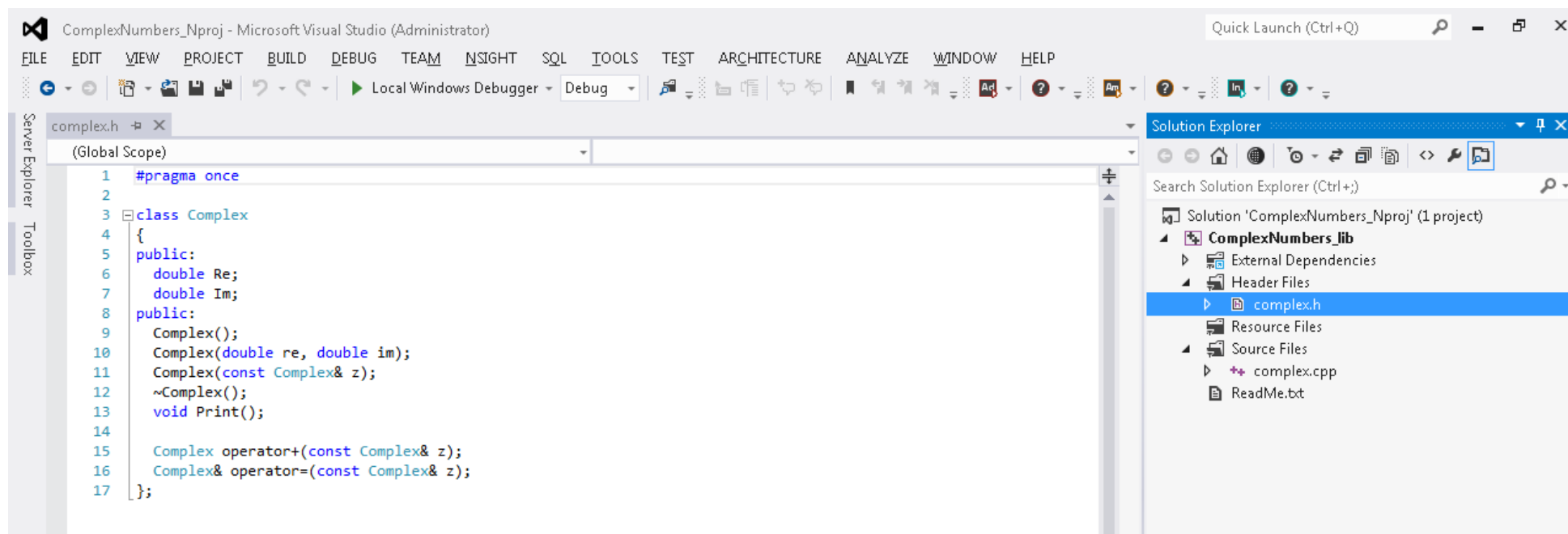
1. Создадим новый solution и в нем первый проект, библиотеку:



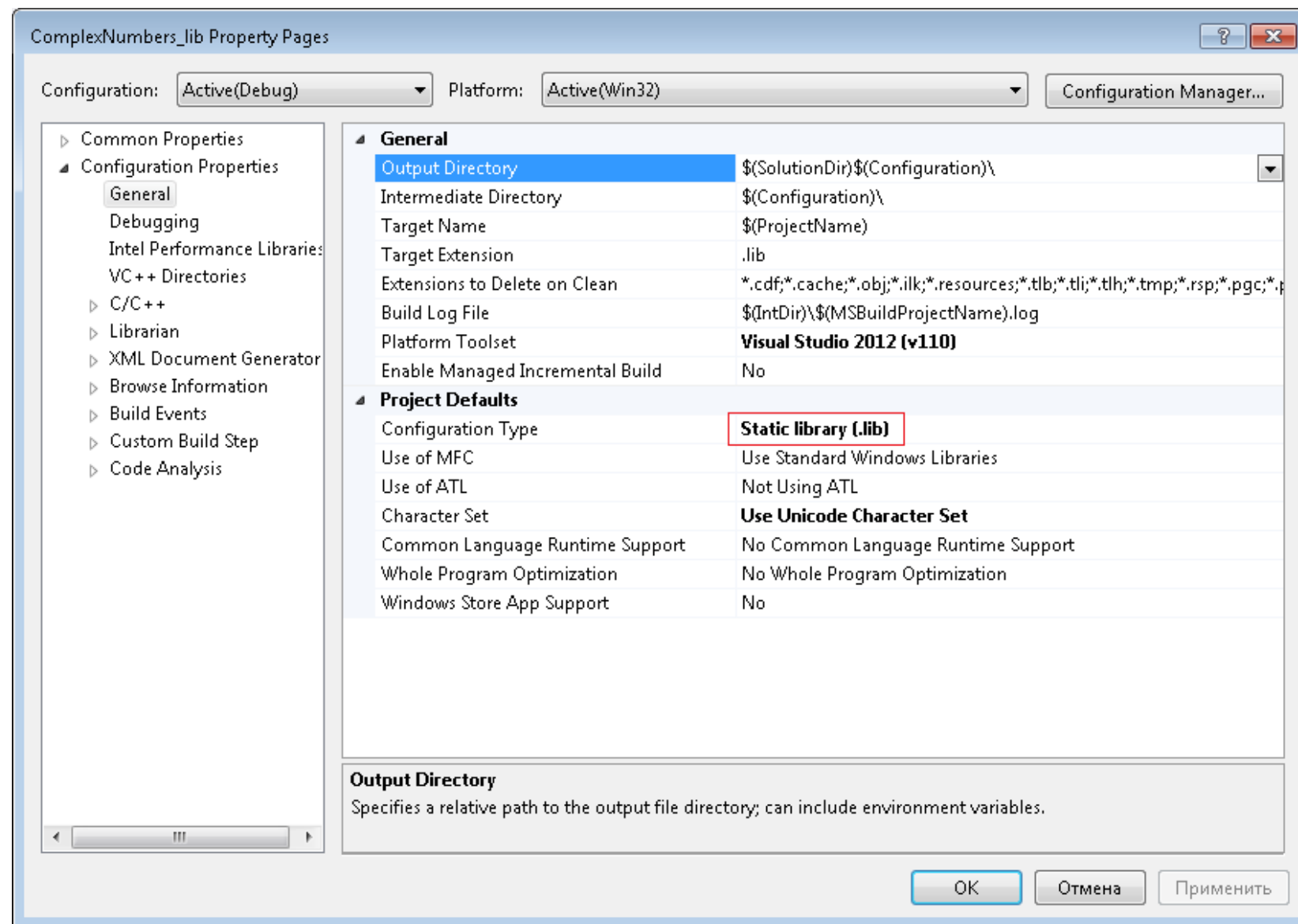
## 2. Выбираем тип – статическая библиотека



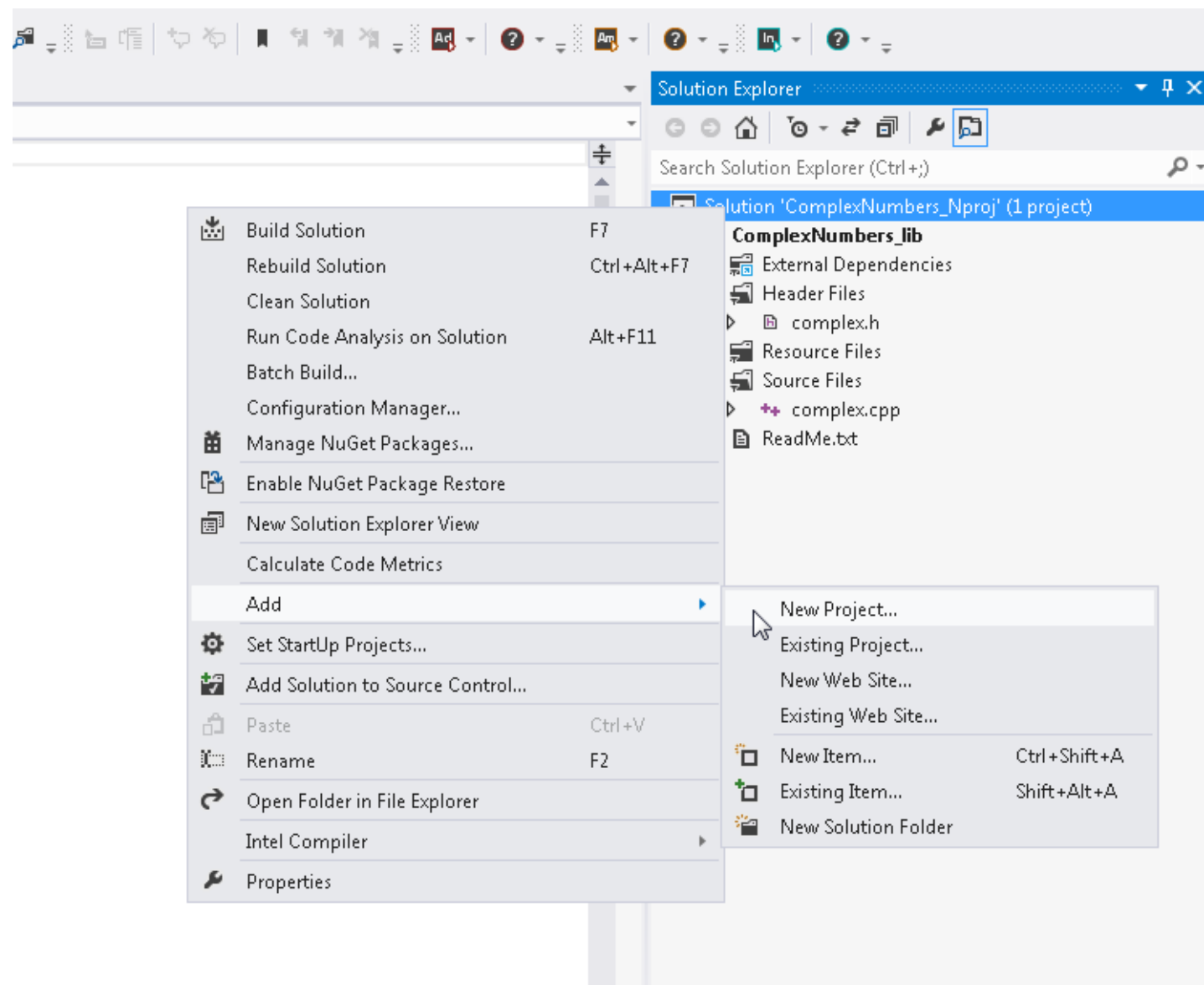
### 3. Добавляем файлы с реализацией



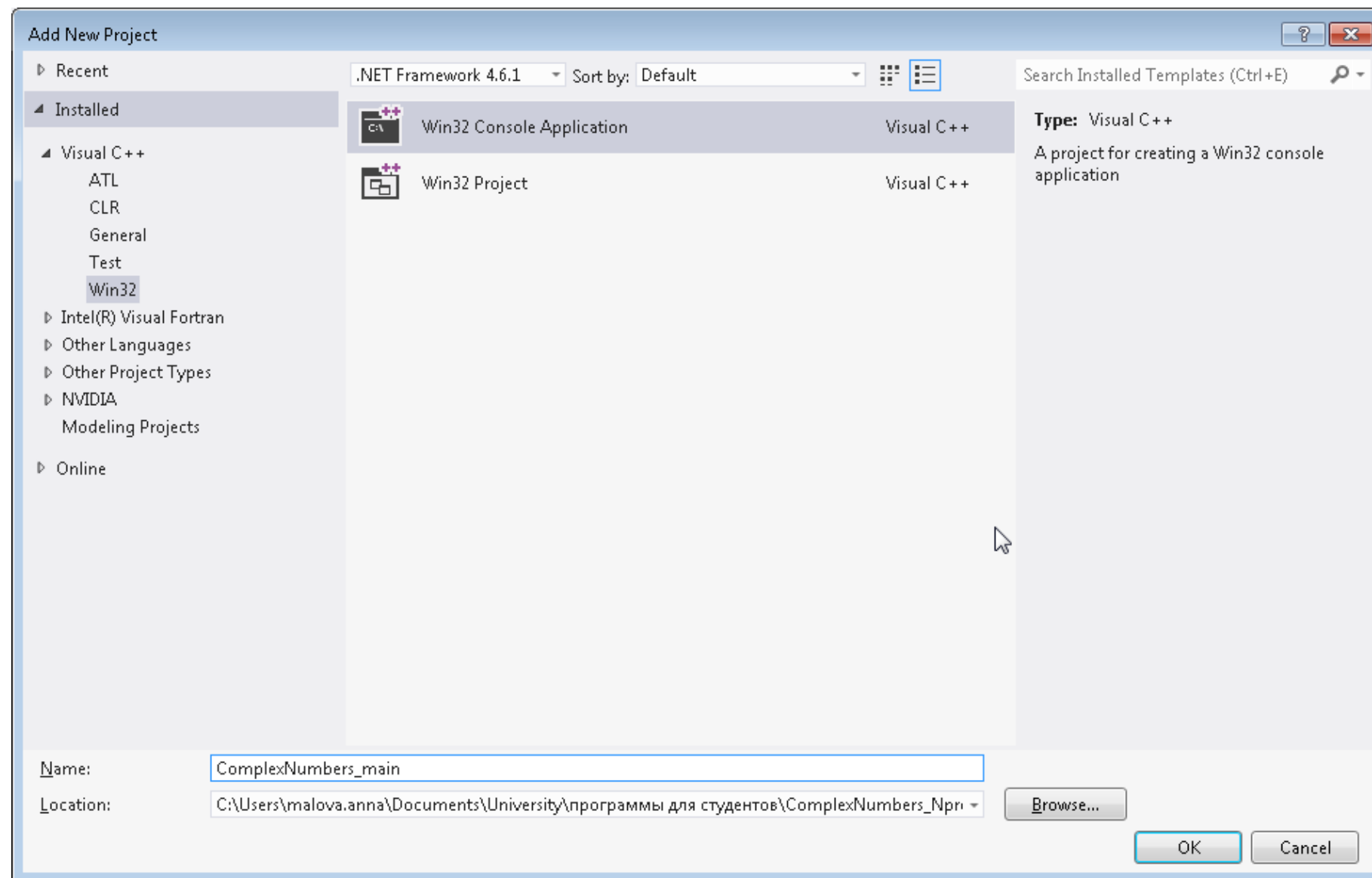
Если открыть вкладку Properties:



4. Добавляем еще один проект в тот же solution:



Это будет приложение:





## Application Settings

Overview

Application Settings

Application type:

- ☐ Windows application
- ☒ Console application
- ☐ DLL
- ☐ Static library

Additional options:

- ☒ Empty project
- ☐ Export symbols
- ☒ Precompiled header
- ☒ Security Development Lifecycle (SDL) checks

Add common header files for:

- ☐ ATL
- ☐ MFC

< Previous

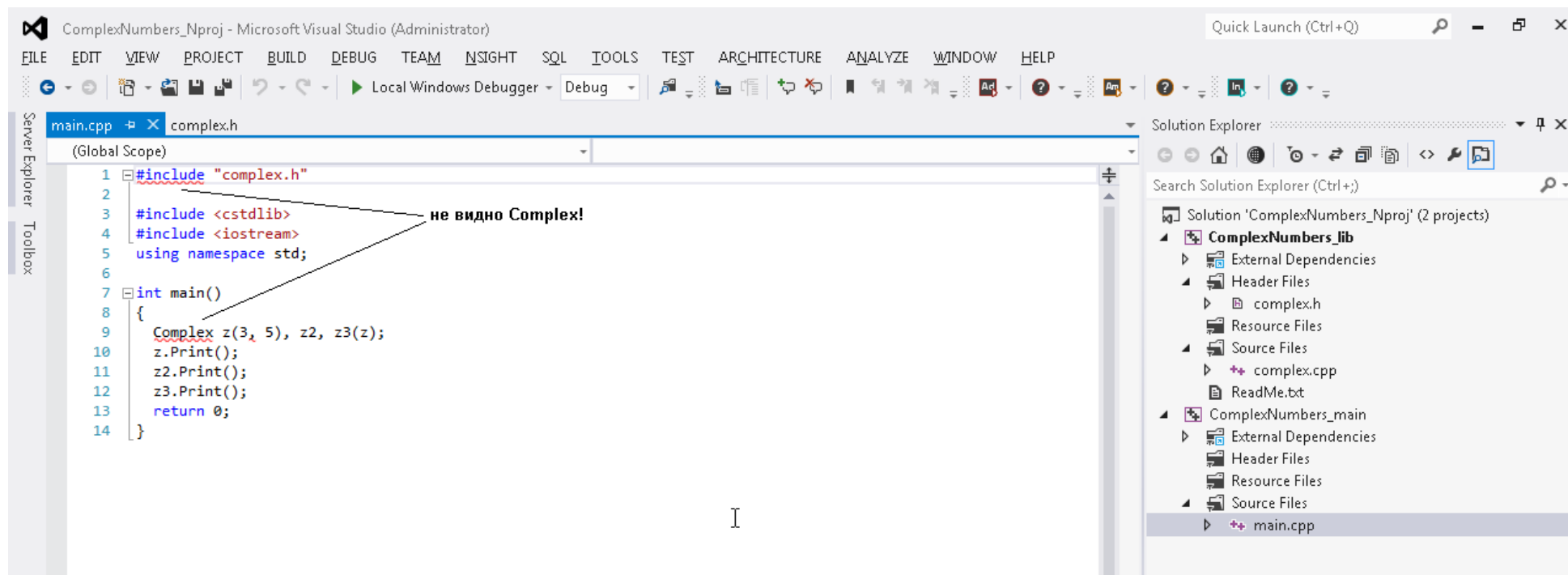
Next >

Finish

Cancel

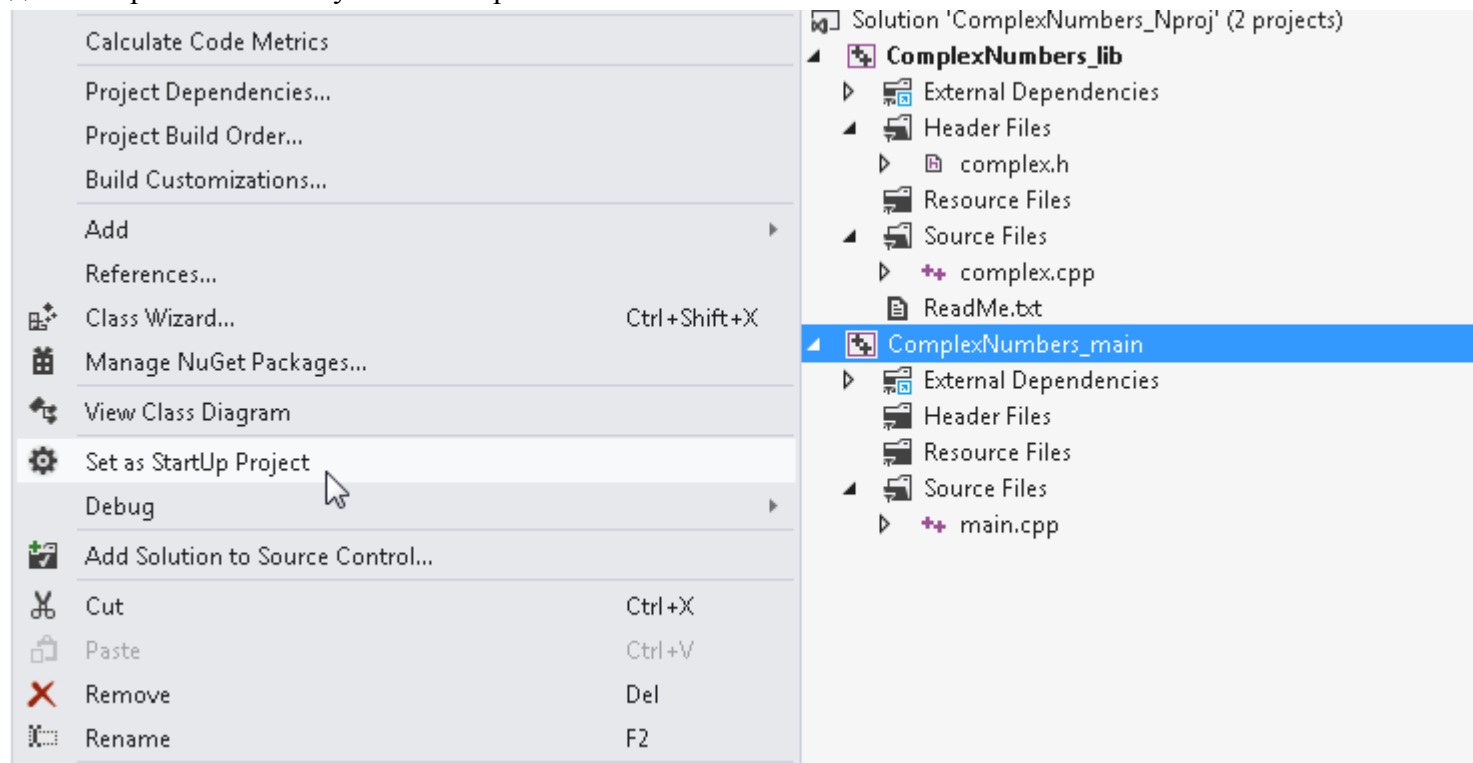
5. Добавляем файлы приложения.

main.cpp не знает об объектах нашей библиотеки:





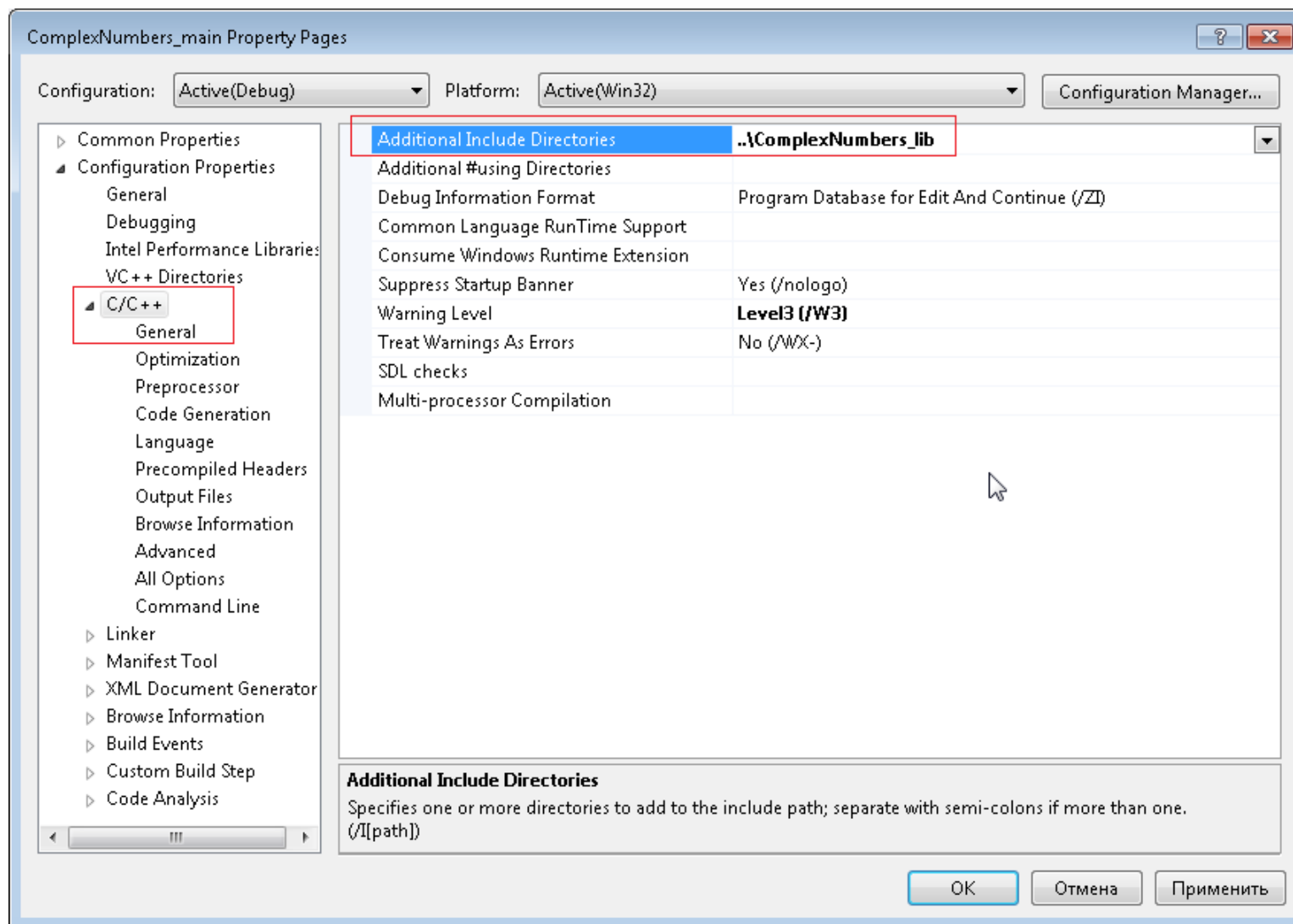
6. Сделаем приложение запускаемым проектом:



**Важно: пункты 7, 8, 9 надо проделать отдельно для Debug и для Release!**

7. В **Properties** -> **C\C++** -> **General** укажем путь к папке, где лежат заголовочные файлы библиотеки.

Примечание: Лучше указать относительный путь. Через «..» указывается подъем на уровень вверх относительно папки с **проектом** (не solution-ом!)



Результат: заголовочный файл найден, реализация функций – нет:

The screenshot shows the Visual Studio IDE with the following components:

- Code Editor:** Displays `main.cpp` with the following code:

```
1 #include "complex.h"
2
3 #include <cstdlib>
4 #include <iostream>
5 using namespace std;
6
7 int main()
8 {
9     Complex z(3, 5), z2, z3(z);
10    z.Print();
11    z2.Print();
12    z3.Print();
13    return 0;
14 }
```

Annotations in the image:

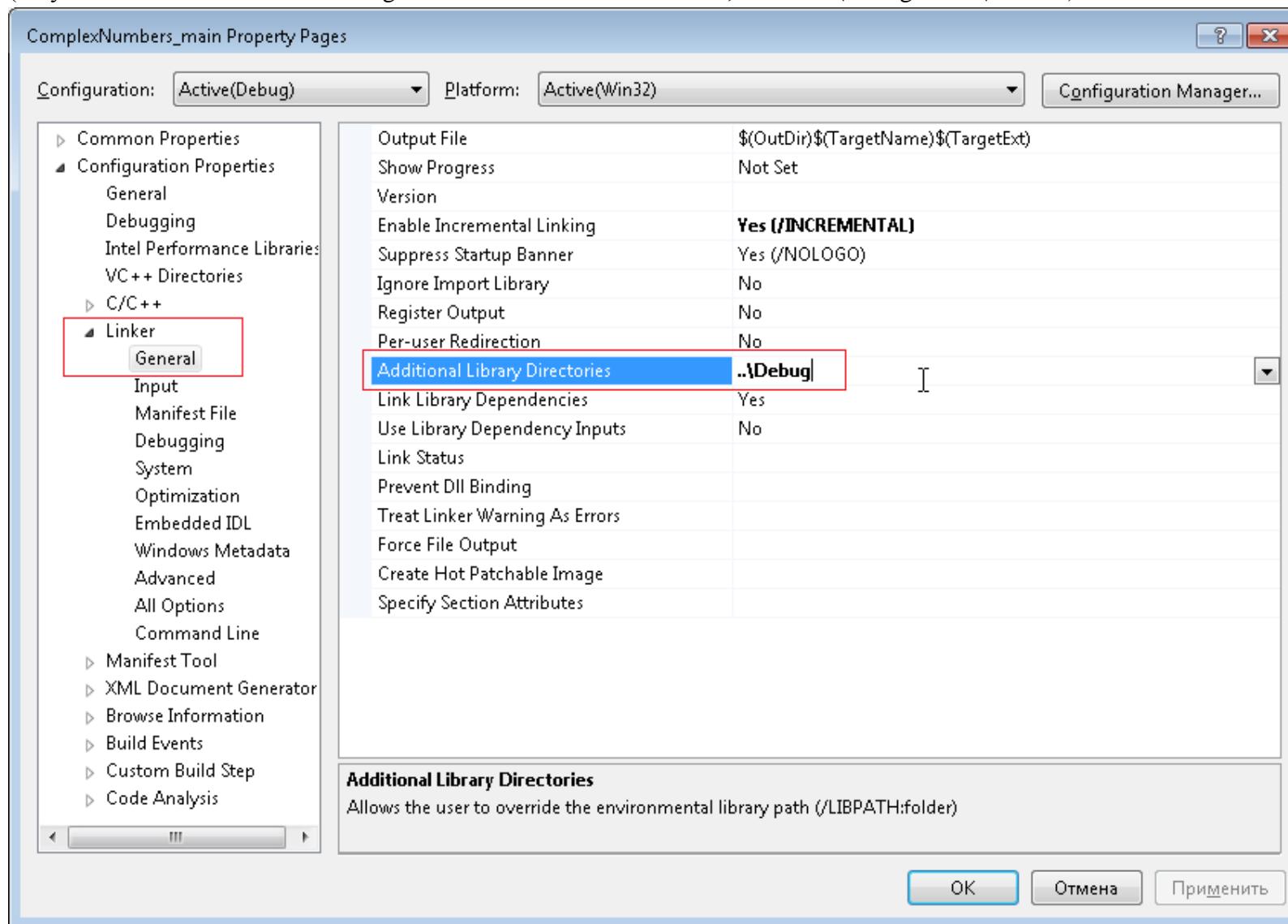
  - A red arrow points from the text "нашли заголовочный файл" (found header file) to the `#include "complex.h"` line.
  - A red arrow points from the text "не нашли реализацию библиотечных функций" (did not find implementation of library functions) to the `Complex` type and its methods in the code.
- Solution Explorer:** Shows the project structure:
  - Solution 'ComplexNumbers\_Nproj' (2 projects)
    - ComplexNumbers\_lib
      - External Dependencies
      - Header Files
        - complex.h
      - Resource Files
    - Source Files
      - complex.cpp
      - ReadMe.txt
  - ComplexNumbers\_main
    - External Dependencies
    - Header Files
    - Resource Files
    - Source Files

- Error List:** Shows 6 errors:

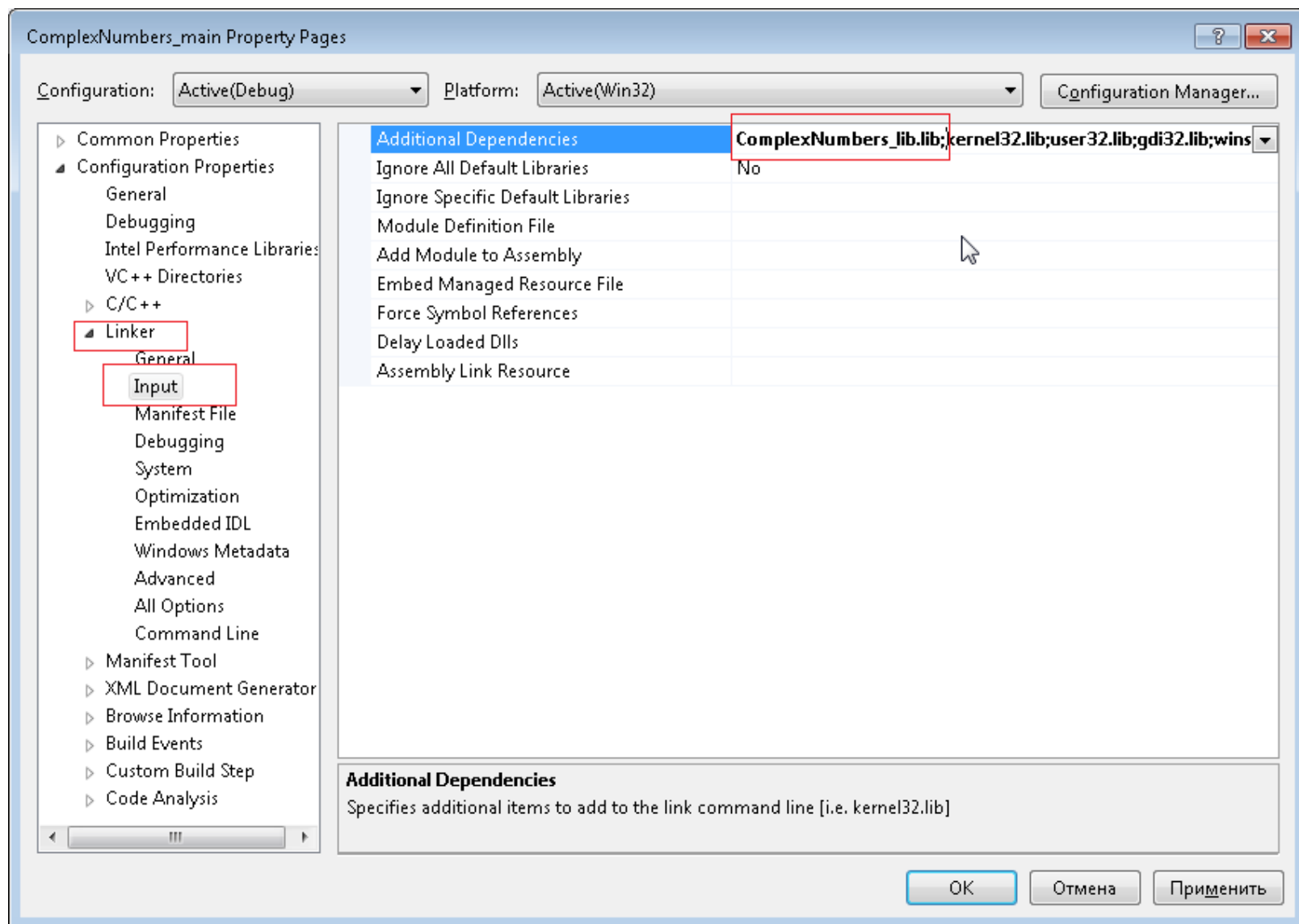
Description	File	Line
error LNK2019: unresolved external symbol "public: __thiscall Complex::Complex(void)" (??0Complex@@QAE@XZ) referenced in function _main	main.obj	
error LNK2019: unresolved external symbol "public: __thiscall Complex::Complex(double,double)" (??0Complex@@QAE@NN@Z) referenced in function _main	main.obj	
error LNK2019: unresolved external symbol "public: __thiscall Complex::Complex(class Complex const &)" (??0Complex@@QAE@ABV0@@@Z) referenced in function _main	main.obj	
error LNK2019: unresolved external symbol "public: __thiscall Complex::~Complex(void)" (??1Complex@@QAE@XZ) referenced in function _main	main.obj	
error LNK2019: unresolved external symbol "public: void __thiscall Complex::Print(void)" (?Print@Complex@@QAE@XZ) referenced in function _main	main.obj	
error LNK1120: 5 unresolved externals	ComplexNumbers_main.exe	

8. В **Properties -> Linker -> General** указываем путь к папке, в которой будет лежать .lib файл:

(по умолчанию – это папка Debug или Release в папке с solution, то есть ..\Debug или ..\Release)

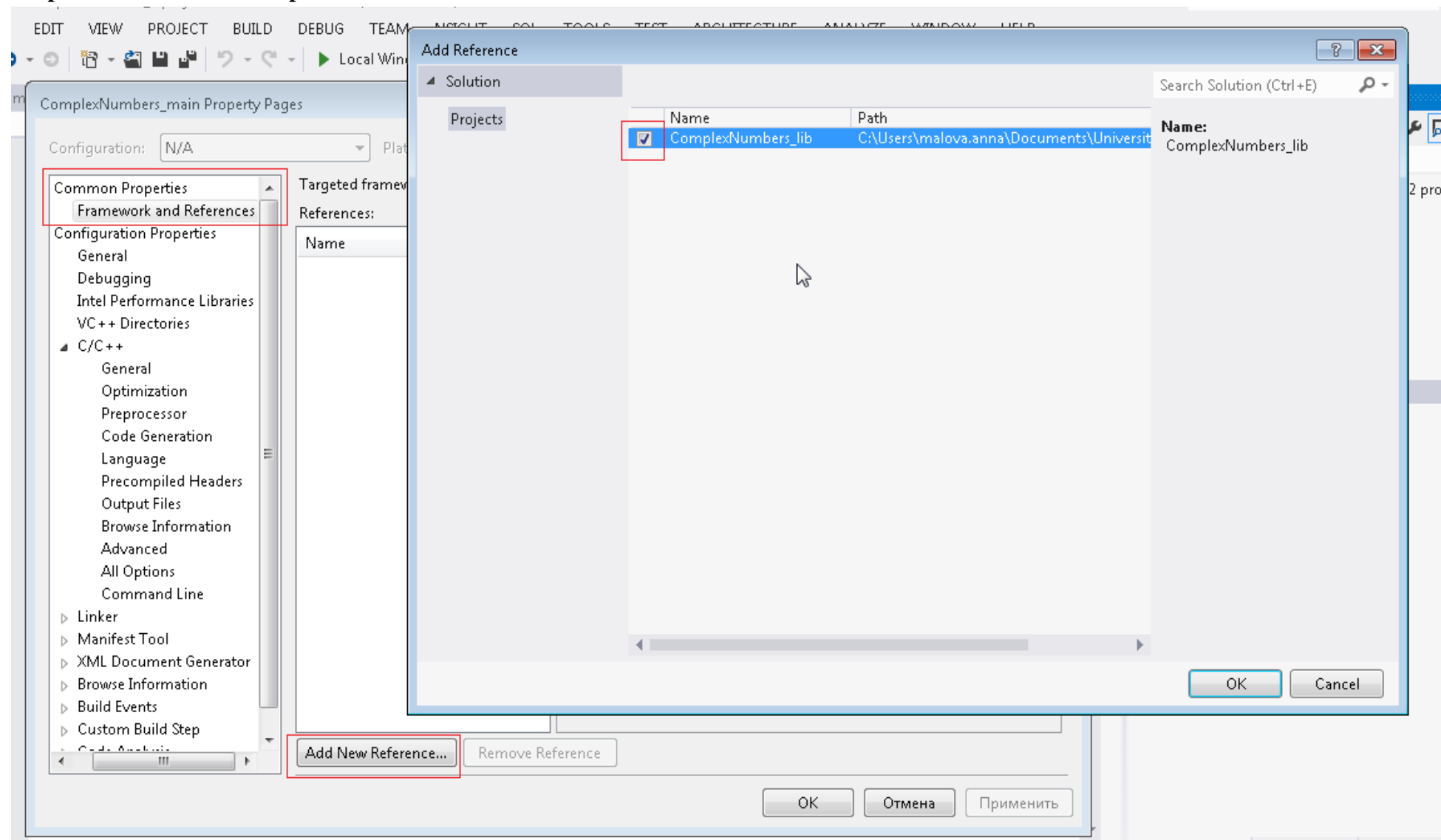


9. В **Properties** -> **Linker** -> **Input** указываем название библиотеки:

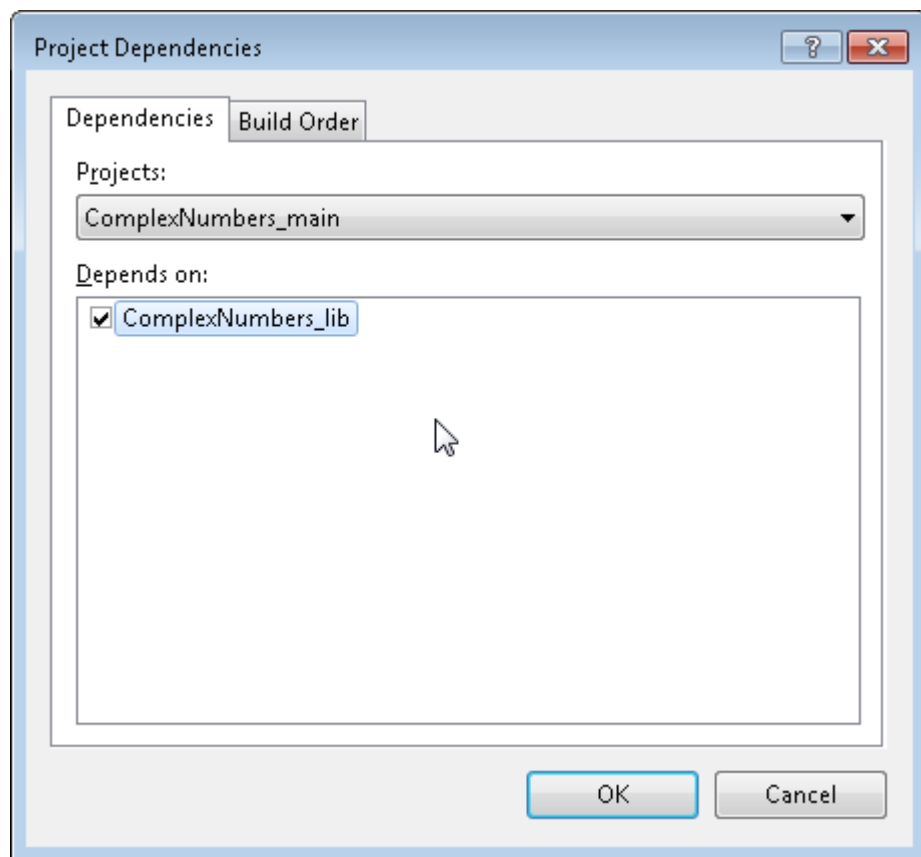


10. Вместо пп. 8 и 9 в VS 2012, 2015 можно указать зависимость между проектами:

**Properties → Common Properties → Add New Reference:**



11. Укажем зависимости сборки: правой кнопкой по названию проекта -> Build Dependencies: (для каждого проекта)



## 12. Результат – успех!

