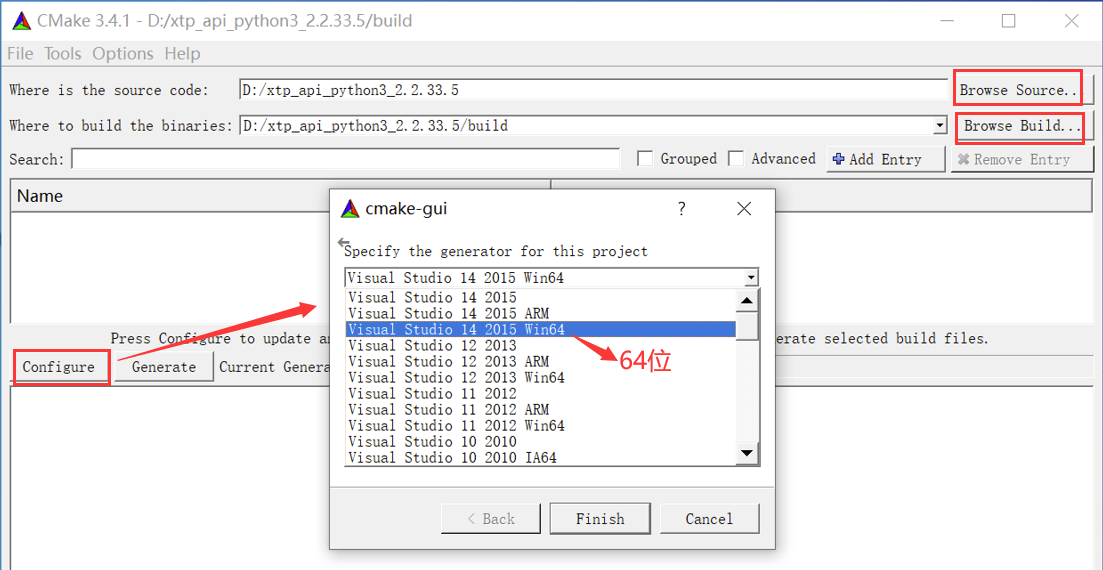
# 使用CMake工具生成工程

在Windows下：

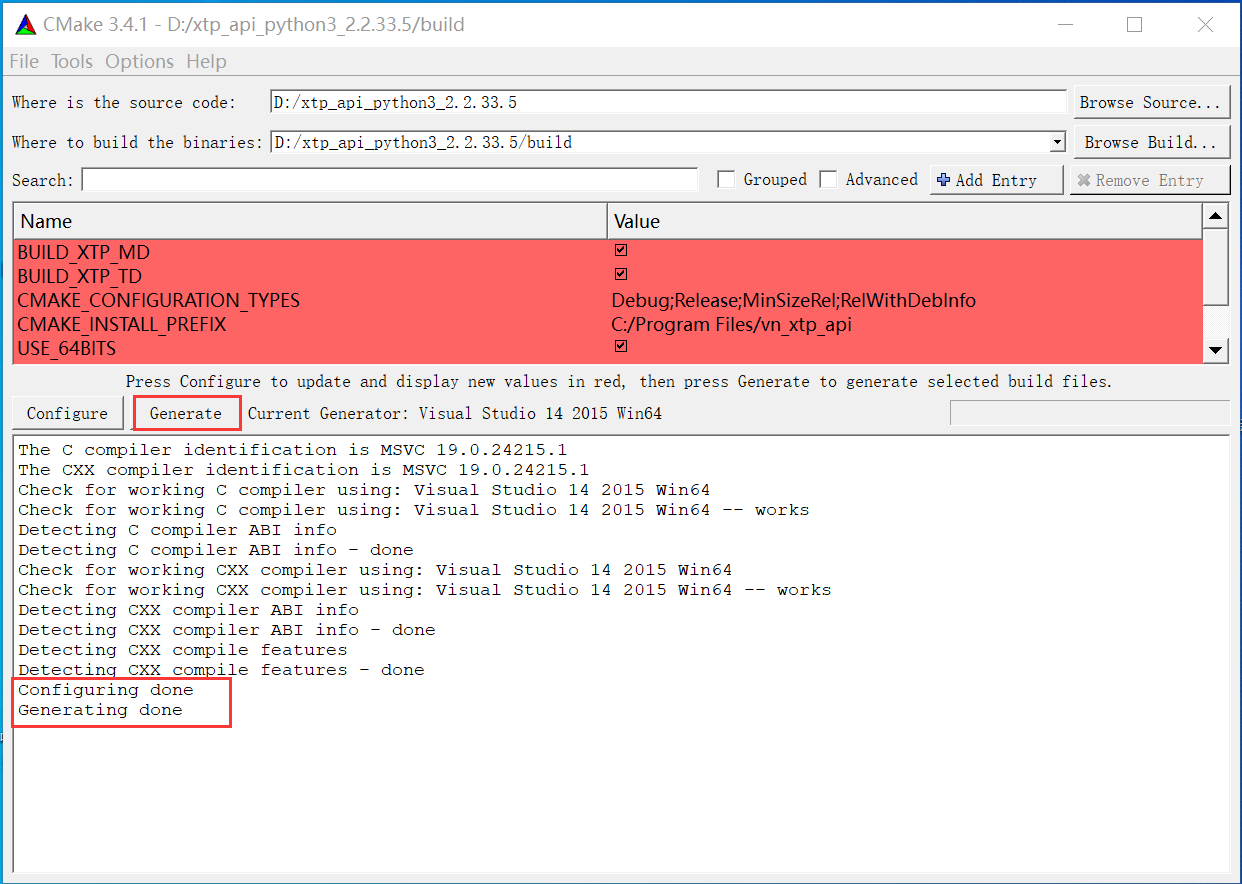
1. 进入source/windows/xtp\_api\_python3\_2.2.33.5路径下，将文件拷贝至demo文件夹，并新建build目录。



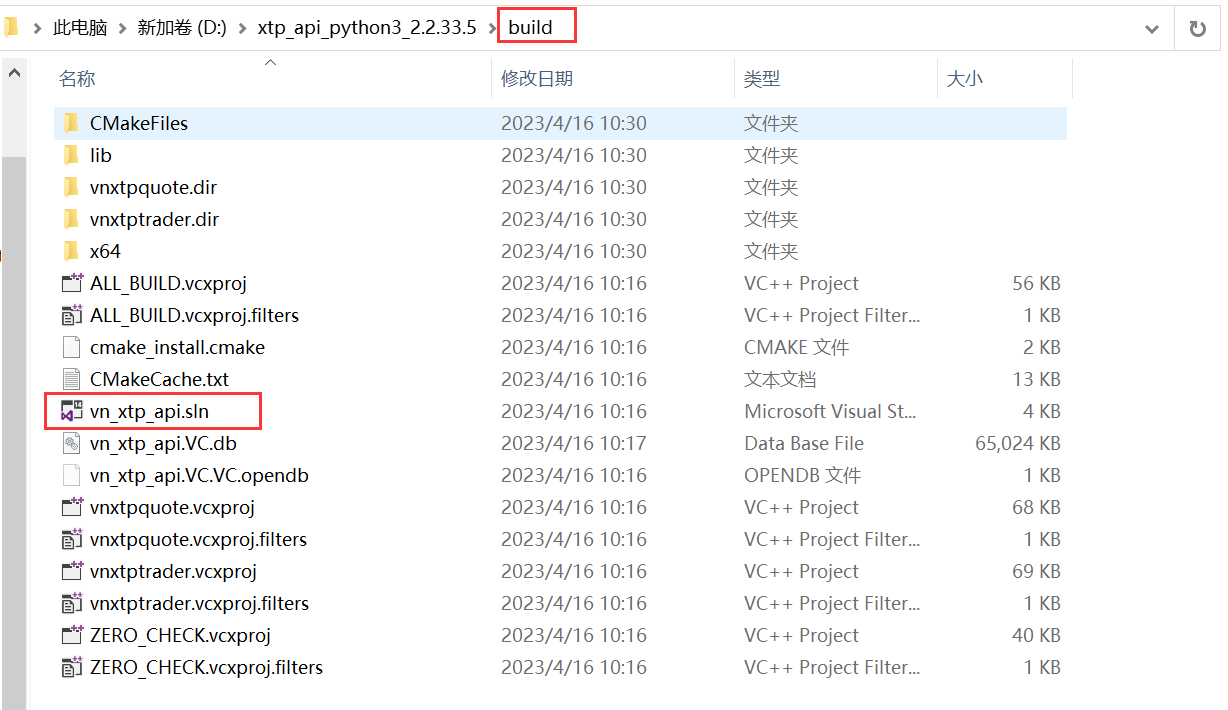
1. 安装windows版cmake工具
2. 打开cmake-gui程序
3. 选择source路径
4. 选择build路径（生成的工程文件路径，这里我们直接新建build文件夹）
5. 点选Configure按钮
6. 弹出框中选择需要使用的win64的编译器（图中选择的是Visual Studio 14 2015 Win64），然后点选Finish按钮。（注意，请选择win64）



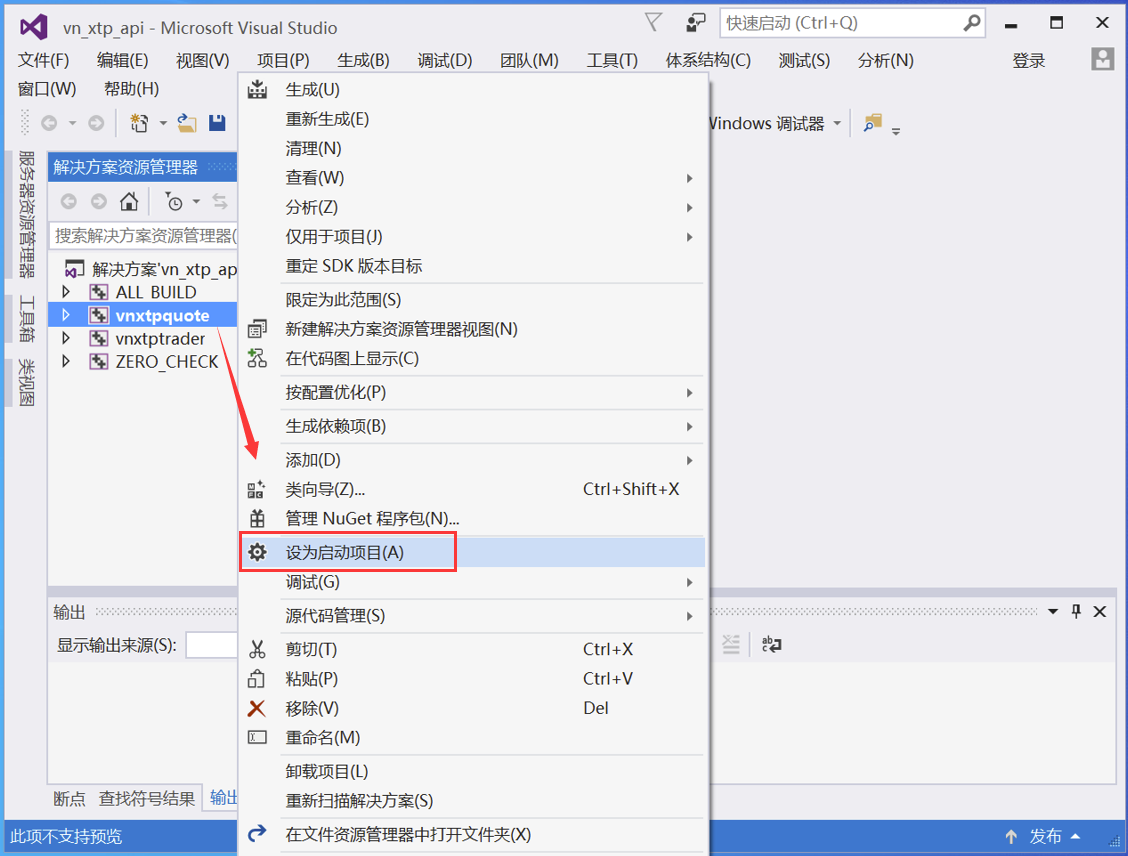
1. 待Configure完成后，点选Generate按钮



1. 等生成工程文件后，用选择的编译器打开生成的工程文件（build路径下）

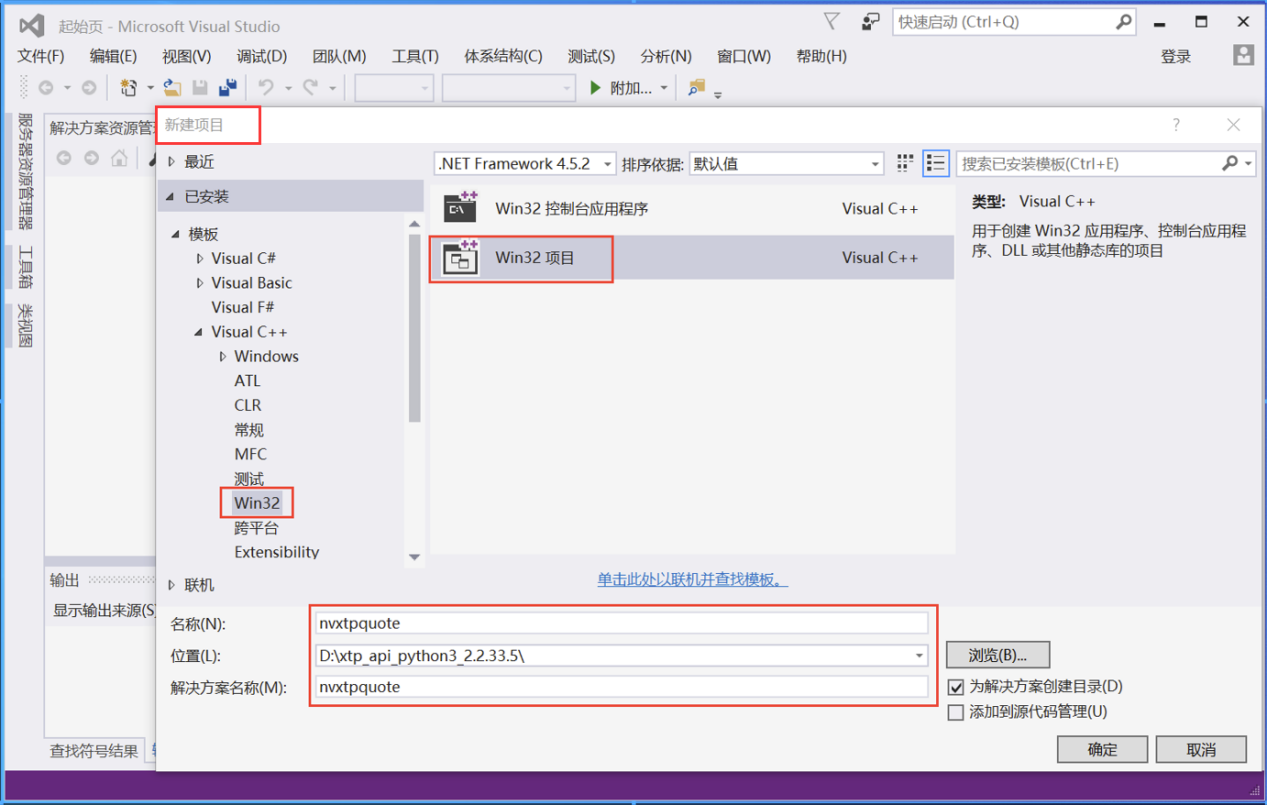


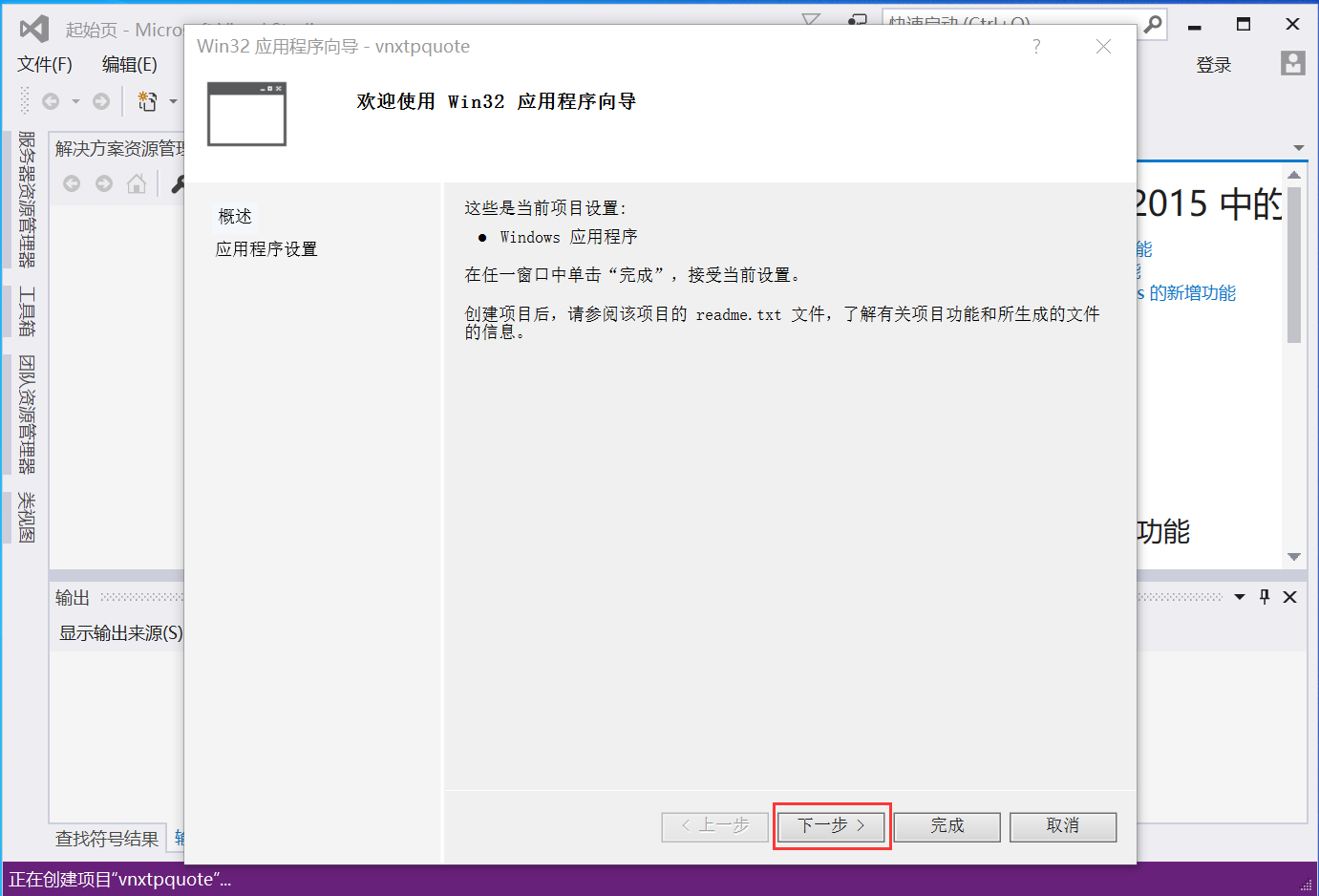
1. 在VS中右键点选vnxtpquote，然后设为启动项目



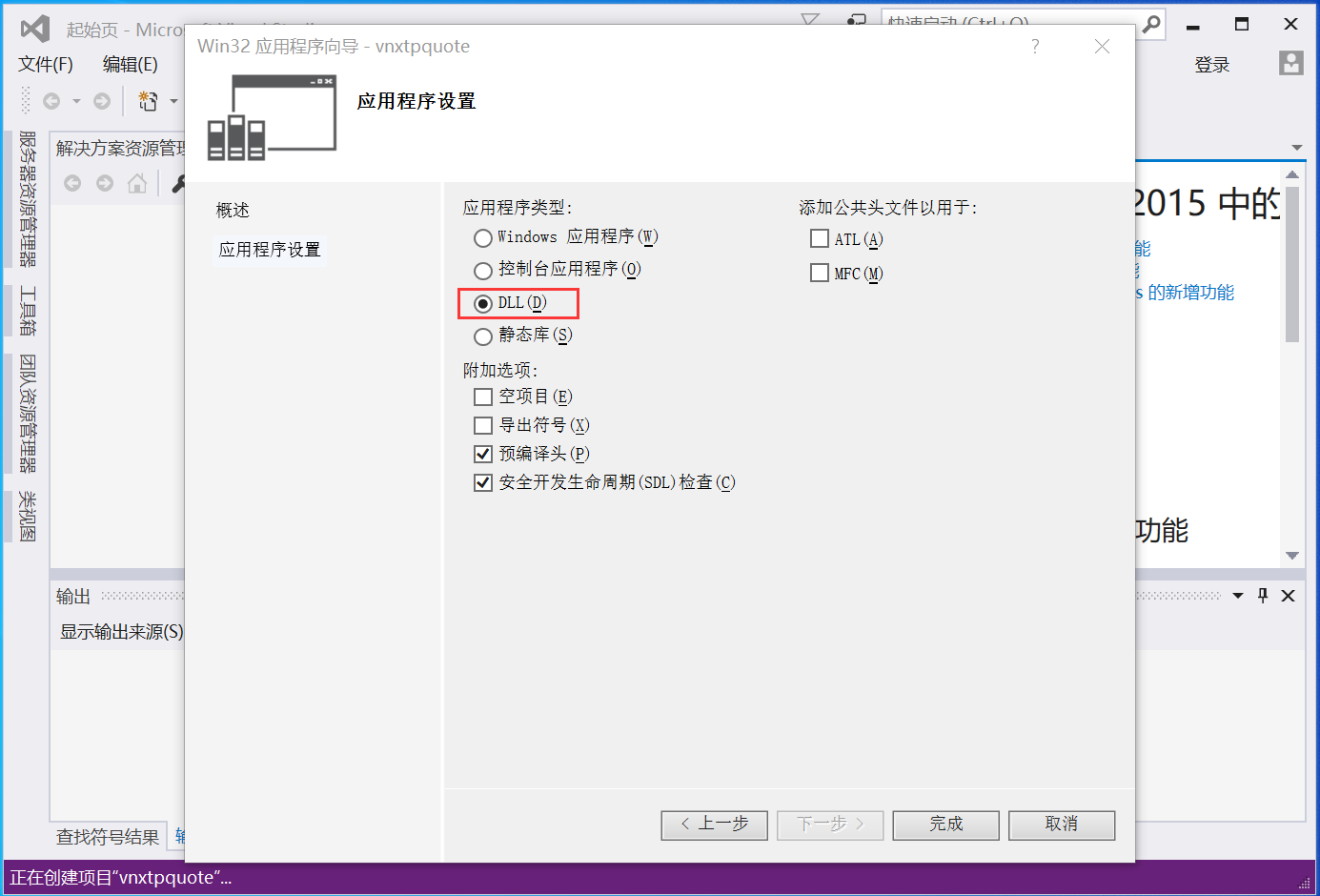
1. 编译运行

# 直接新建配置vs工程

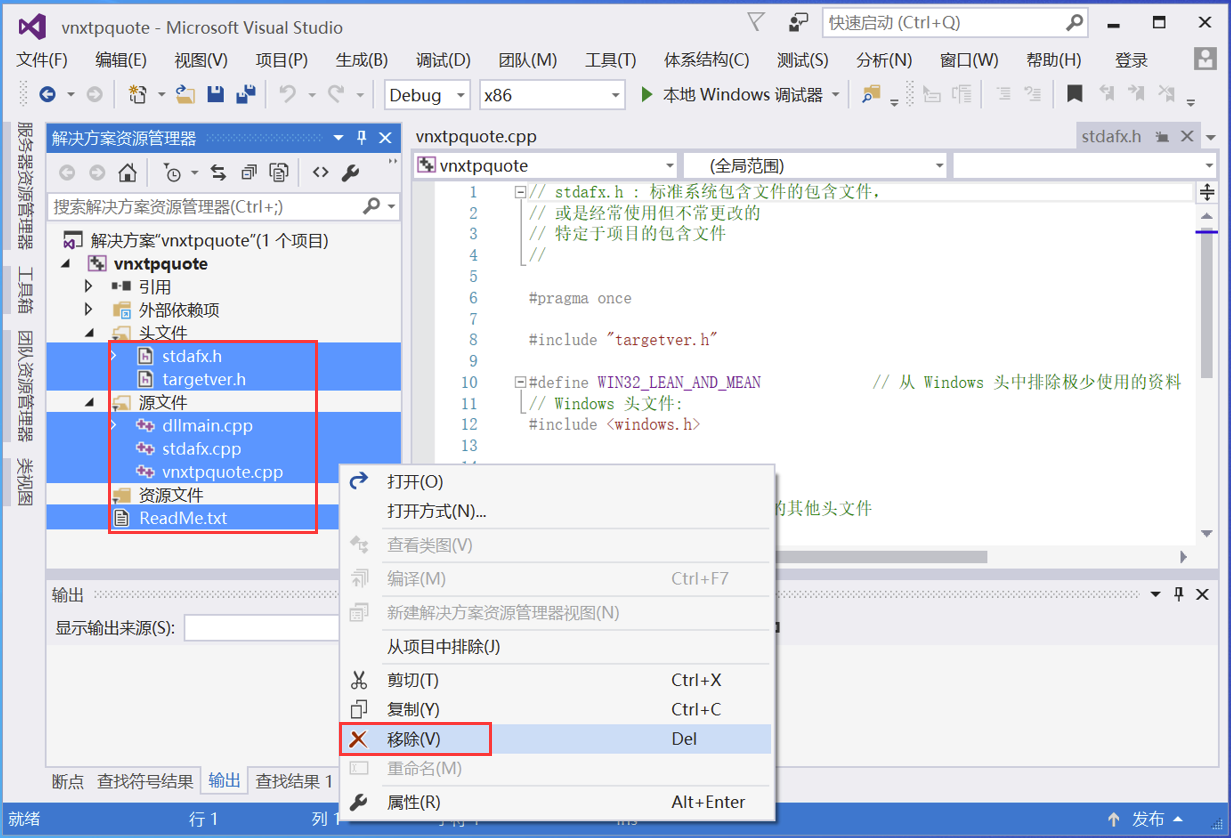
1. 新建Win32项目，设置项目名称、位置
2. 点击确定，进入下一步

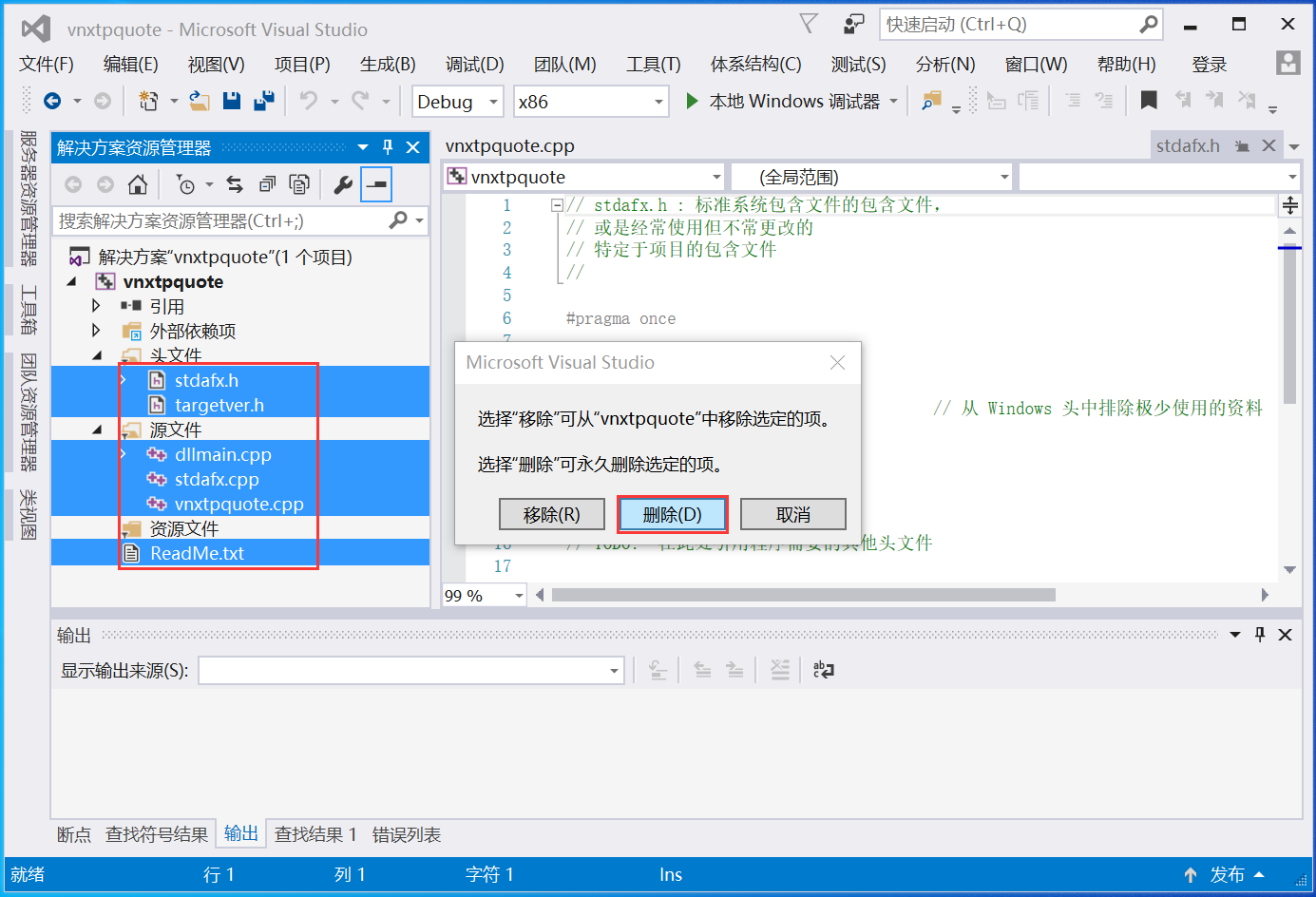


1. 应用程序类型选择“DLL”

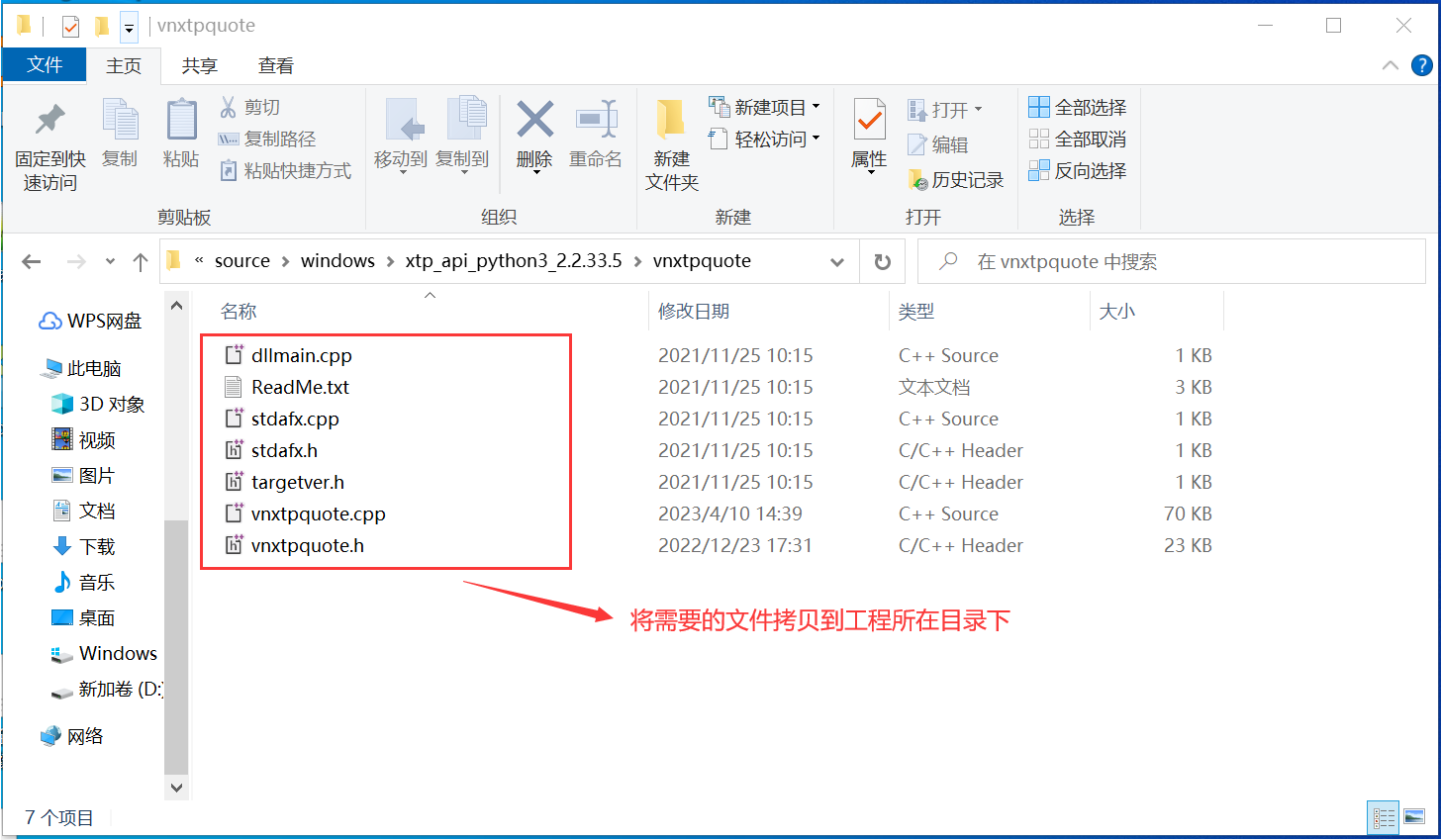


1. 将默认生成的文件从项目中删除

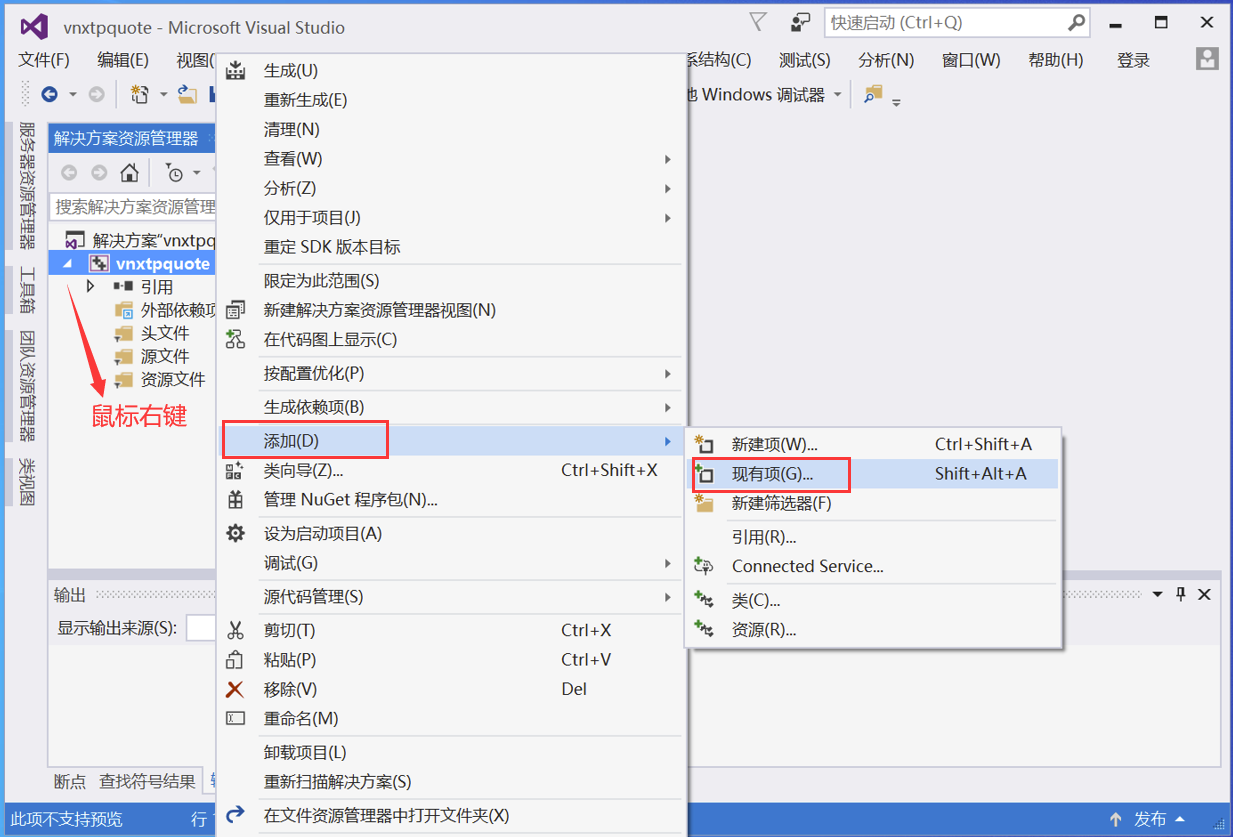


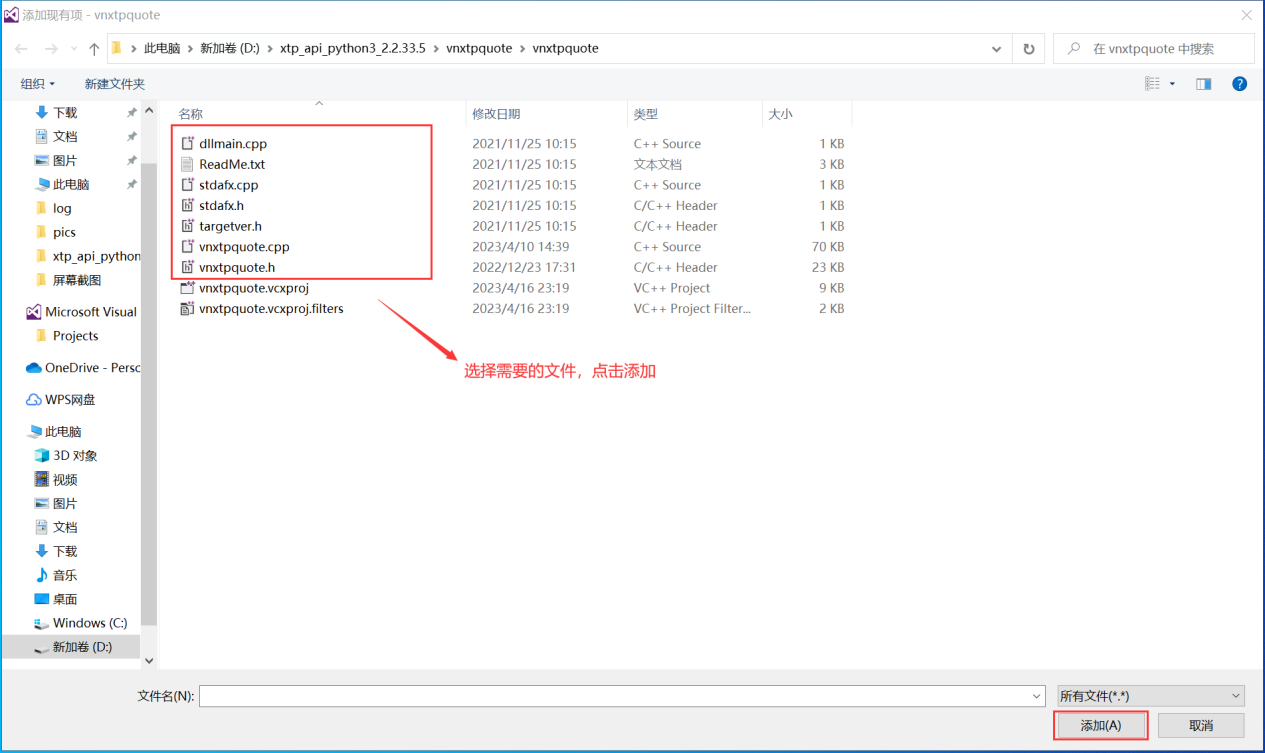


1. 将source/windows/xtp\_api\_python3\_2.2.33.5/vnxtpquote下的源码拷贝至工程所在目录下。



1. 对vnxtpquote项目右键，将拷贝过来的文件作为现有项添加到该项目

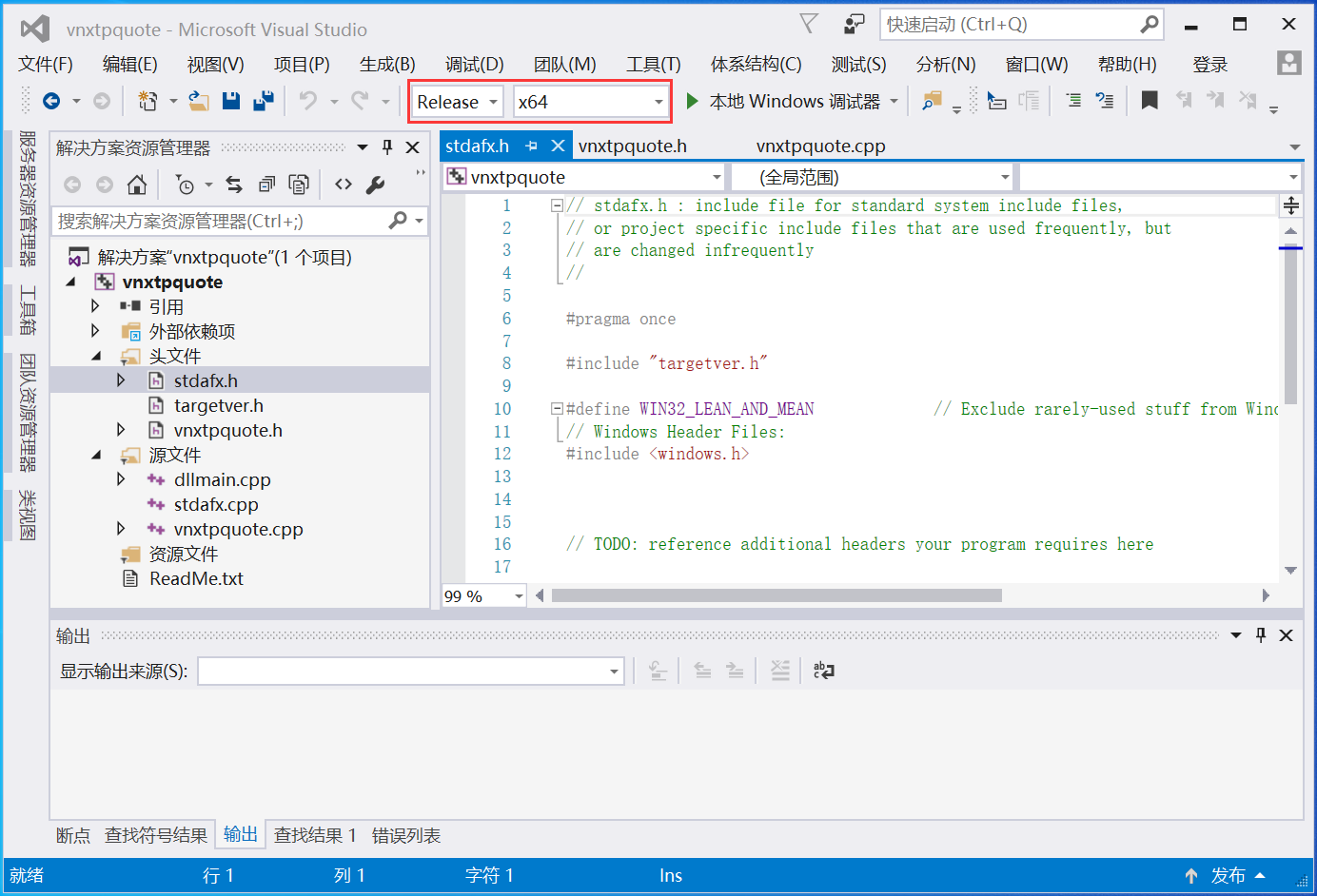




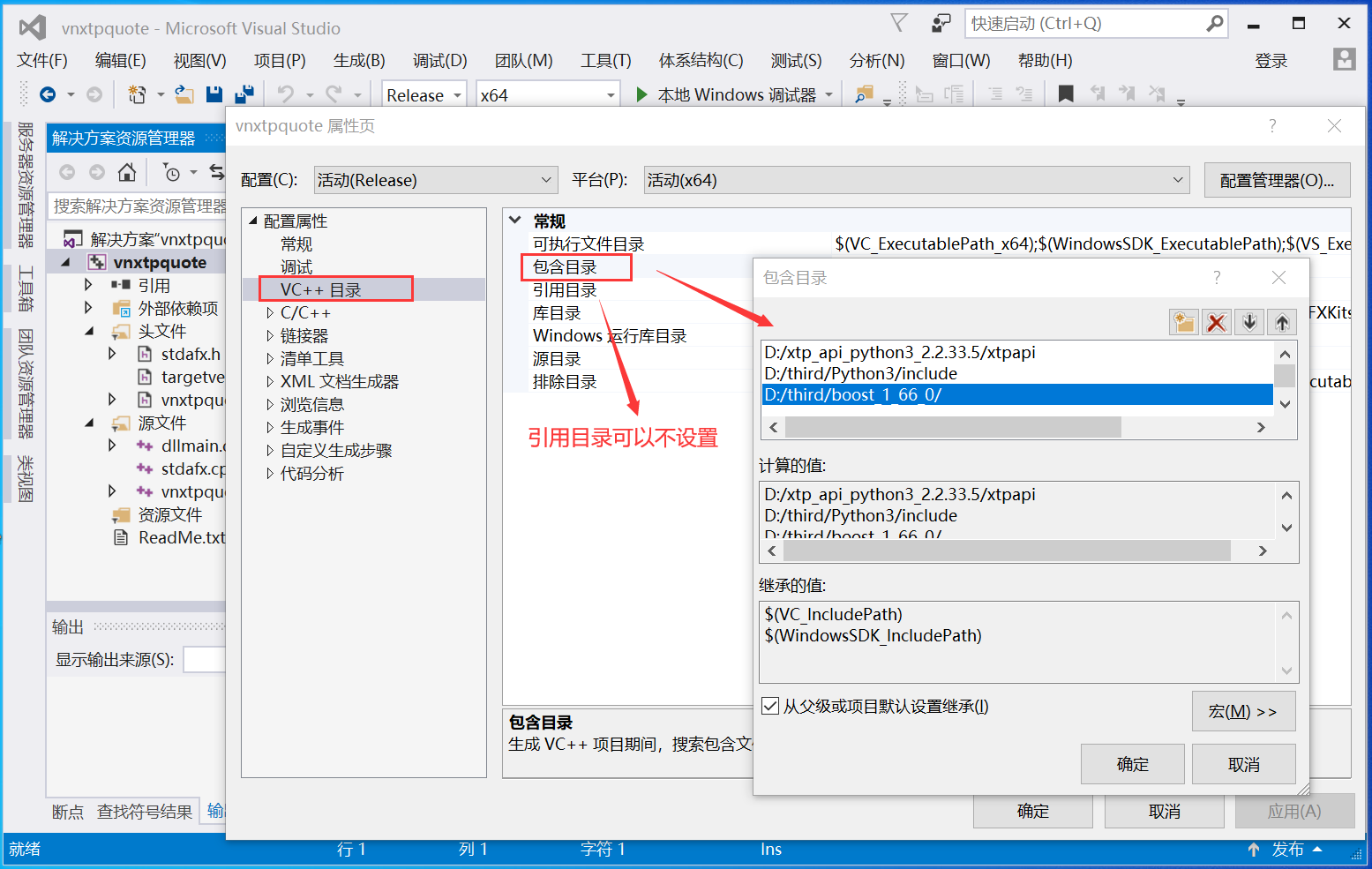
1. 编译运行

# 编译64位的release版本

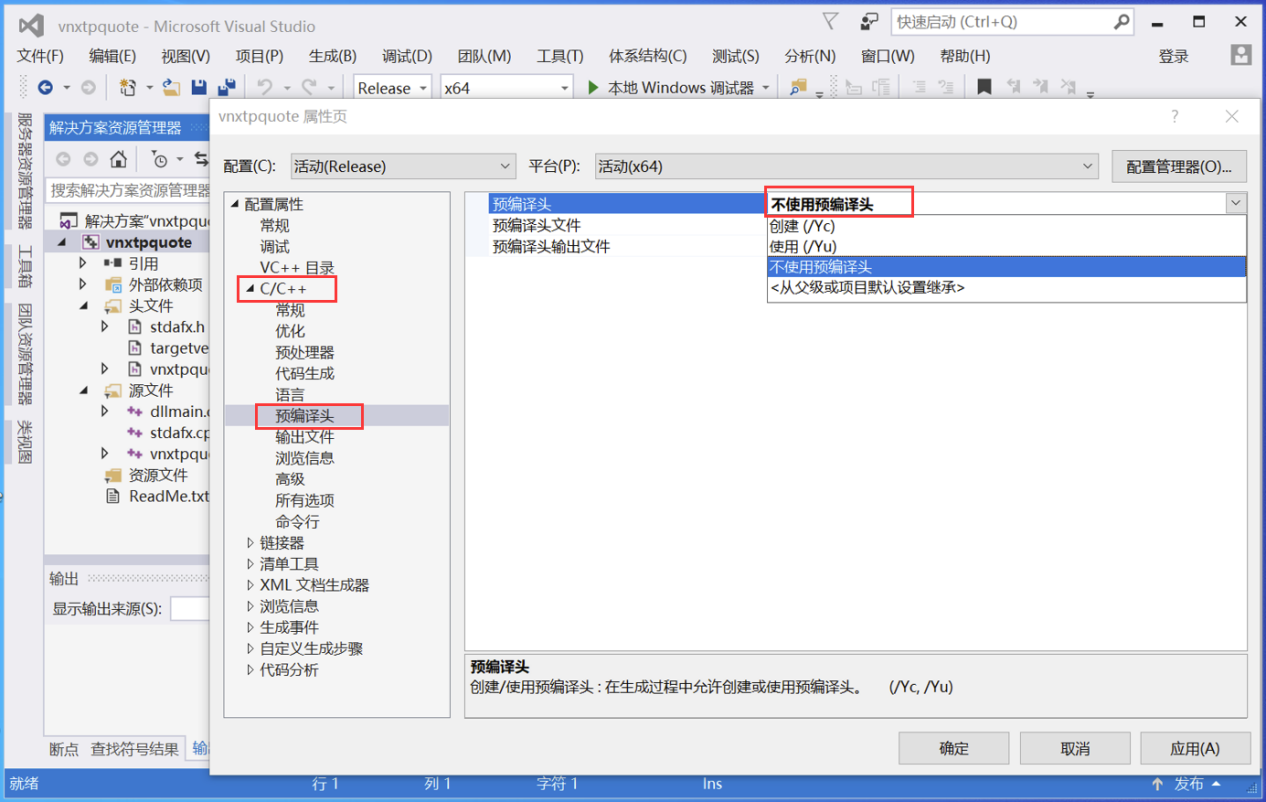
1. 解决方案配置，设置位Release，并选择x64



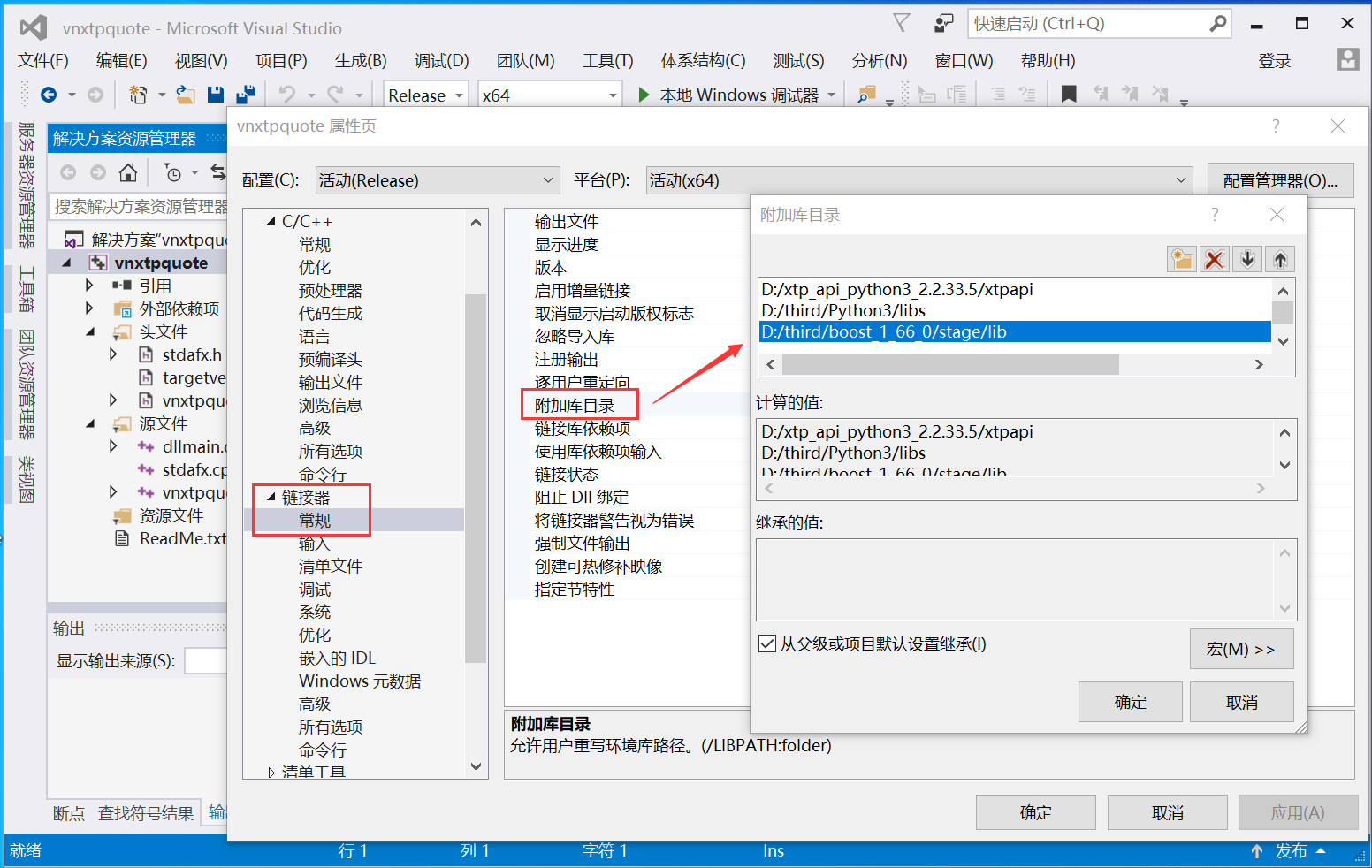
1. 对vnxtpquote项目右键打开属性，点击【VC++目录】，设置【包含目录】



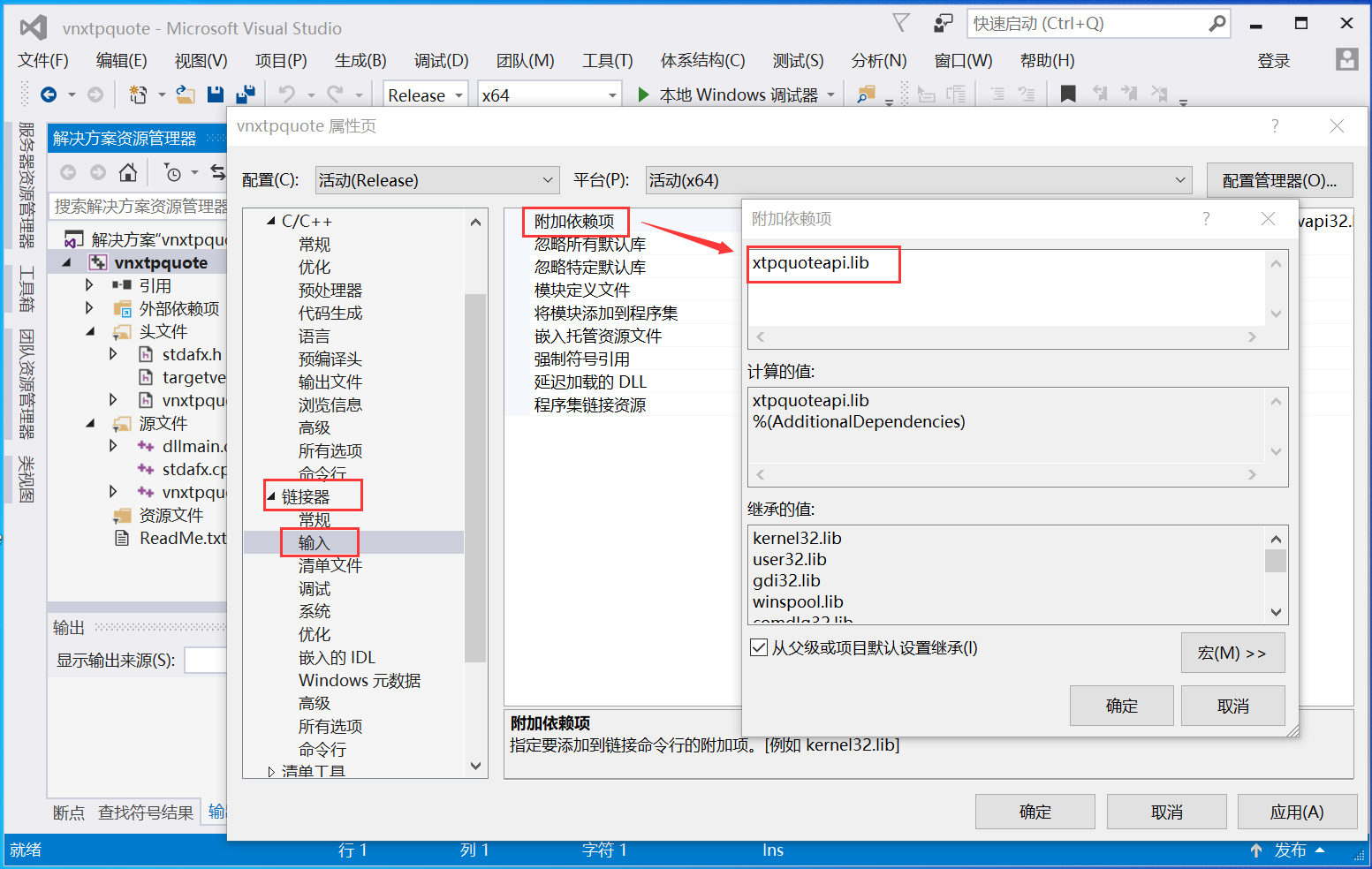
1. 点击【C/C++】展开，选中【预编译头】，设置为“不使用预编译头”



1. 点击【链接器】展开，选中【常规】，设置附加库目录



1. 选中【输入】，设置【附加依赖项】，添加依赖库



1. 编译项目

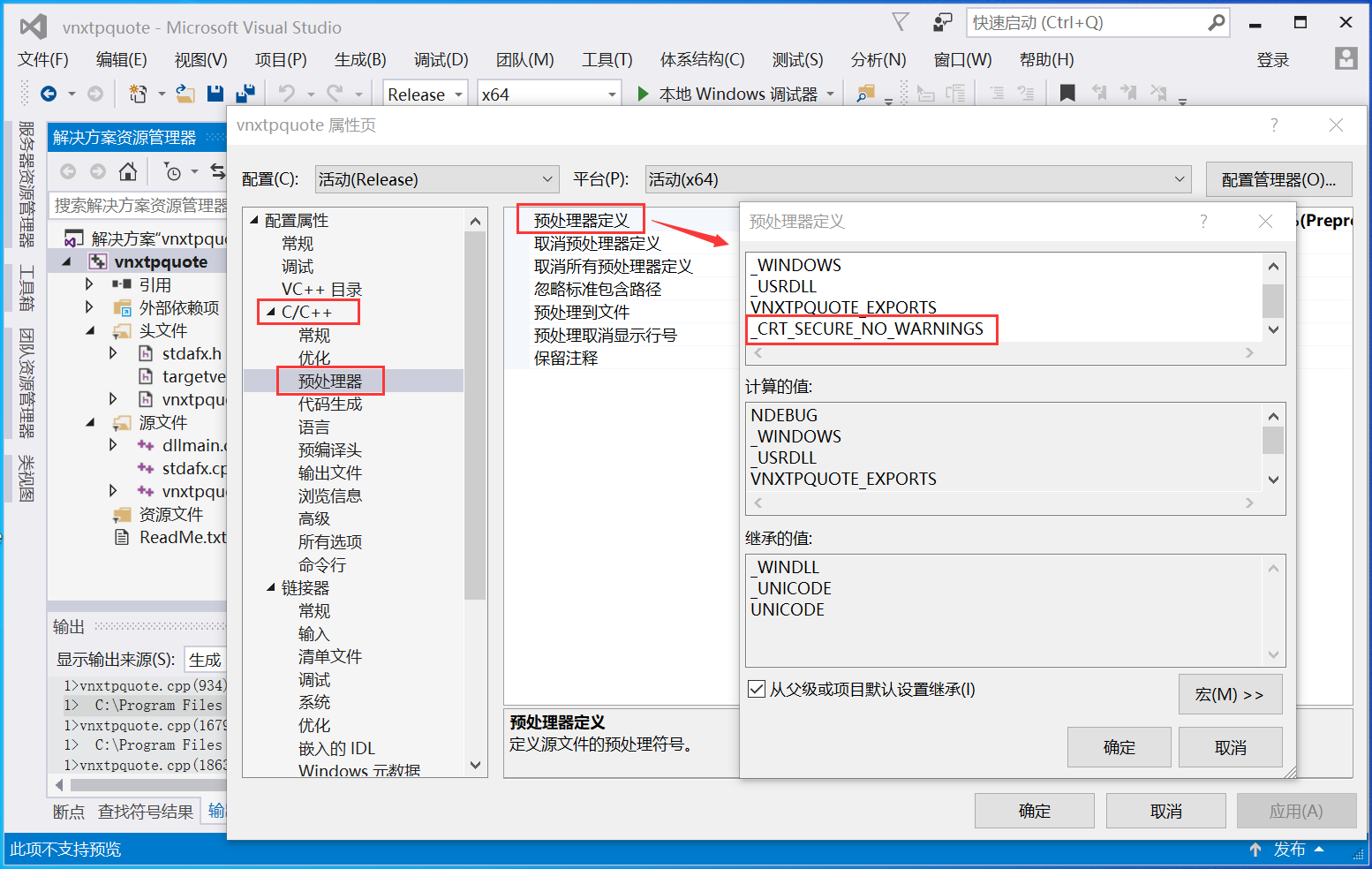
如果编译时出现如下错误：

1>vnxtpquote.cpp(934): error C4996: 'strncpy': This function or variable may be unsafe. Consider using strncpy\_s instead. To disable deprecation, use \_CRT\_SECURE\_NO\_WARNINGS. See online help for details.

1>C:\Program Files (x86)\Windows Kits\10\Include\10.0.10240.0\ucrt\string.h(346): note: 参见“strncpy”的声明

则是因为[strcpy函数](https://so.csdn.net/so/search?q=strcpy%E5%87%BD%E6%95%B0&spm=1001.2101.3001.7020" \t "https://blog.csdn.net/nyist_yangguang/article/details/_blank)不安全造成的溢出。

**解决方法：**对项目右键打开【属性】，点击【C/C++】里的【预处理器】，对【预处理器定义】进行编辑，加入一段代码：\_CRT\_SECURE\_NO\_WARNINGS。



1. 编译成功，生成.dll文件。

