# Cpp Plugin插件使用说明(2017)

目录

[Cpp Plugin插件使用说明(2017) 1](#_Toc493010522)

[一、配置环境 1](#_Toc493010523)

[二、VS下载与安装 2](#_Toc493010524)

[1. VS下载 2](#_Toc493010525)

[2. VS安装 2](#_Toc493010526)

[三、插件安装与卸载 4](#_Toc493010527)

[1. 插件安装 4](#_Toc493010528)

[2. 插件卸载 5](#_Toc493010529)

[四、插件功能说明 5](#_Toc493010530)

[登陆 6](#_Toc493010531)

[下载 6](#_Toc493010532)

[测试 7](#_Toc493010533)

[上传 8](#_Toc493010534)

[登出 8](#_Toc493010535)

[五、Visual Studio基本使用演示 8](#_Toc493010536)

[5.1 新建Solution、Project、File 8](#_Toc493010537)

[5.2 Visual Studio编码界面介绍 12](#_Toc493010538)

[5.3 解决方案层级结构 12](#_Toc493010539)

[5.4 编码 13](#_Toc493010540)

[5.5 编译与错误窗口 13](#_Toc493010541)

[5.6 调试操作与界面 14](#_Toc493010542)

[六、插件操作过程演示 16](#_Toc493010543)

[七、注意事项 20](#_Toc493010544)

## 一、配置环境

Visual Studio 2013 professional/ultimate/community

（VS其他版本可能可以用，但是不保证提供后期功能维护）

CppPlugin.vsix（插件安装包）

CppMonitor.vsix（插件安装包）

## 二、VS下载与安装

### 1. VS下载

1.1 紫荆站（推荐）

http://zijingbt.njuftp.org/stats.html?id=85743

* 1. Microsoft官网

https://www.visualstudio.com/zh-hans/vs/older-downloads/

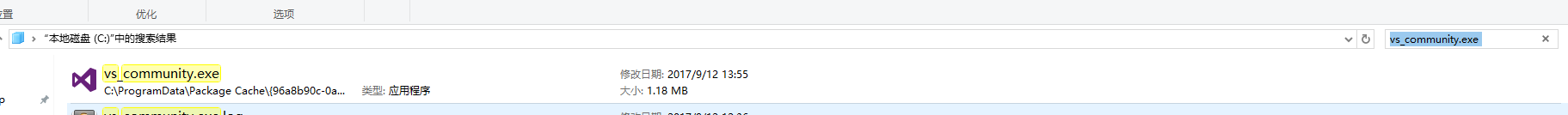
登录微软账号，选择” join the VS Dev Essential”

选择Download，搜索Visual Studio 2013，选择合适版本下载

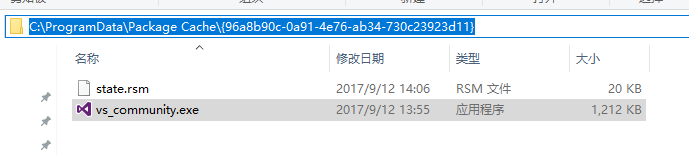
### 2. VS安装

* 1. 彻底卸载旧版本

1. 注意，在卸载Visual Studio主体程序之前，请将手动安装的语言包或扩展包在控制面板卸载。
2. 在C盘搜索vs\_community.exe（专业版/企业版，vs\_professional.exe /vs\_enterprise.exe ）



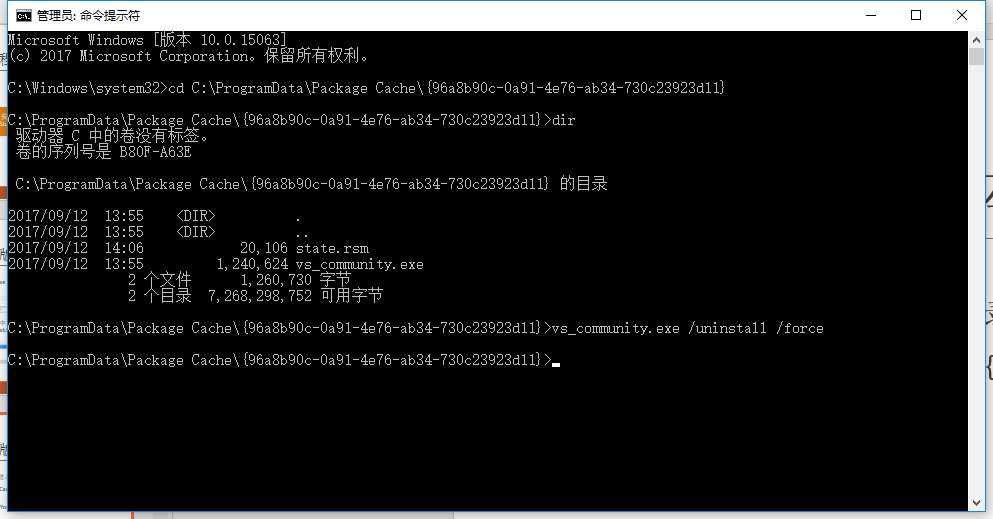
1. 复制\*.exe文件的路径（图中为 C:\ProgramData\Package Cache\{96a8b90c-0a91-4e76-ab34-730c23923d11} ）



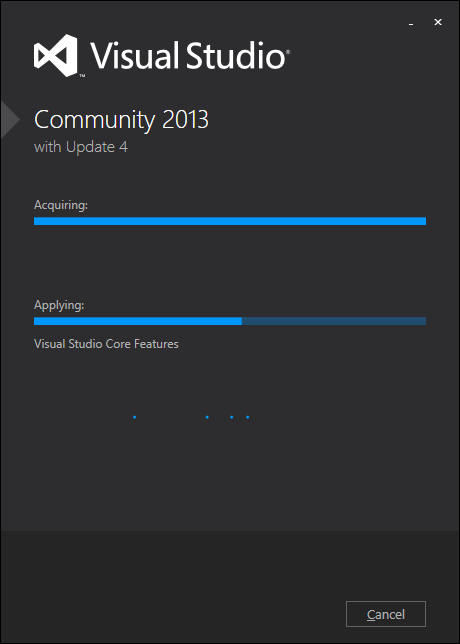
1. 以管理员身份启动cmd，进入目录，输入卸载命令。

Cd C:\ProgramData\Package Cache\{96a8b90c-0a91-4e76-ab34-730c23923d11}

vs\_community.exe /uninstall /force



1. 等待卸载完成。



* 1. 安装Visual Studio

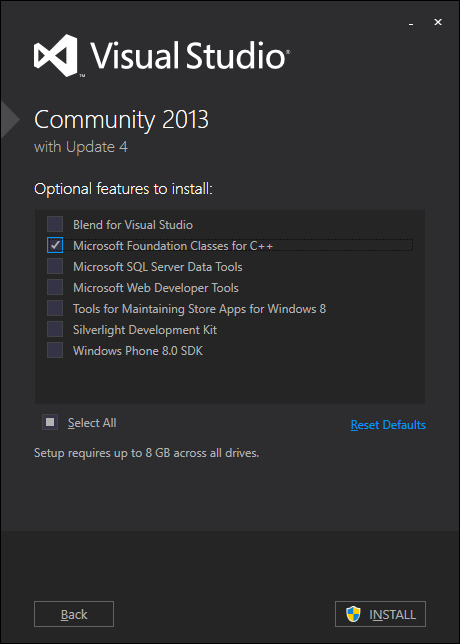
（1）挂载安装镜像vs2013.4\_ce\_enu.iso。

（2）运行vs\_community.exe启动安装。

（3）安装位置自选。

（4）可选组件只需勾选Microsoft Foundation Classes for C++

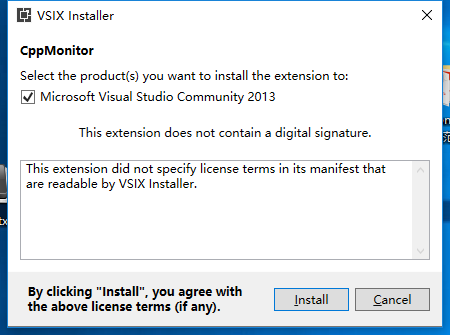
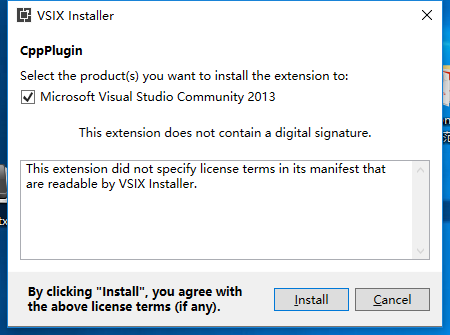
（5）选择install，等待安装完成。



## 三、插件安装与卸载

### 1. 插件安装

双击CppPlugin.vsix（CppMonitor.vsix）,根据提示点下一步即可



### 2. 插件卸载

1 点击菜单[工具（T）]

2 选择 [扩展与更新（U）]

3 在列表中找到 CppPlugin（CppMonitor），选择卸载即可

## 四、插件功能说明

插件正常使用的流程图如图1所示。

### 登陆

1 输入账号密码

2 点击登陆

前置条件

1未登录

其他

1如果出现错误，会有弹窗给出提示信息。

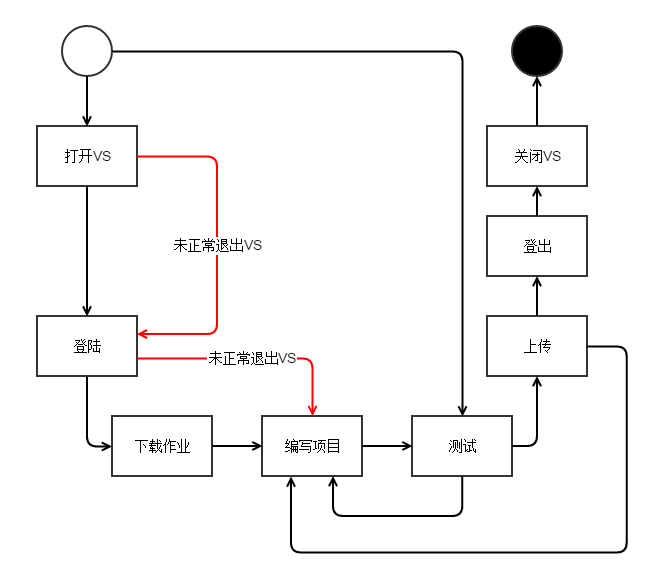


图1 业务流程图

### 下载

1 选择待下载的考试

2 选择存放考试描述信息的文件夹

3 点击下载

前置条件

1 已登录

2 用户未处于考试练习状态

其他说明

1如果出现错误，会有弹窗给出错误提示信息。

2 用户选择的存放考试描述信息的文件夹下不能有任何文件或者文件夹。

### 测试

（上传前必须先测试，用户的考试成绩由本地测试结果的历史最高成绩决定，不测试就没有成绩）

1 选择需要测试的题目

2 开始测试

3 查看测试反馈信息

前置条件

1 用户已登录

2 用户处于考试练习状态

其他说明

1. 测试要求使用Debug配置，请勿选择Release。且工程名需要与测试问题ID相同。
2. 对相应的题目点击Run运行测试之后，按钮Run将变为Cancel，此时可以单击Cancel按钮以取消当前测试。
3. 在Detail一栏中给出的测试状态码，释义如下表所示：

|  |  |  |  |
| --- | --- | --- | --- |
| **状态码** | **全称** | **解释** | **备注** |
| AC | ACCEPT | 答案正确 |  |
| WA | WRONG\_ANSWER | 答案错误 | 程序运行输出结果错误。若申请了过大的内存，测试程序中断了运行，导致输出结果为空，也将导致WA |
| TIE | TIME\_EXCEEDED | 运行超时 |  |
| RTE | RUNTIME\_ERROR | 运行时错误 | 如指针使用错误导致程序崩溃，但由于程序崩溃系统反应时间不同，该类型的错误可能被记录为TIE |

注：

在题目没有特别说明情况下，测试程序默认参数为：

1. 10MB内存限制
2. 1000ms运行时间限制
3. 普通任务调度优先级
4. CPU单核计算

原则上测试程序允许同时对多个题目运行测试，但是该操作将可能加重系统任务调度及计算负担，导致运行超时。因此不建议同时对多个题目进行测试。

### 上传

(编写并测试完练习题目后必须上传，不上传文件，服务器会没有用户完成考试/练习的记录，也就没有成绩)

1 选择上传

前置条件

1 已登录

2 用户处于考试练习状态

其他说明

1 如果出现错误，会有弹窗给出错误提示信息。

### 登出

1 点击登出

前置条件

1 用户已登录

其他说明

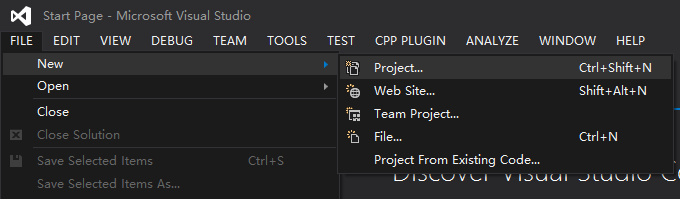
1如果存在错误，会有弹窗给出对应的错误提示。

## 五、Visual Studio基本使用演示

### 5.1 新建Solution、Project、File

5.1.1新建解决方案

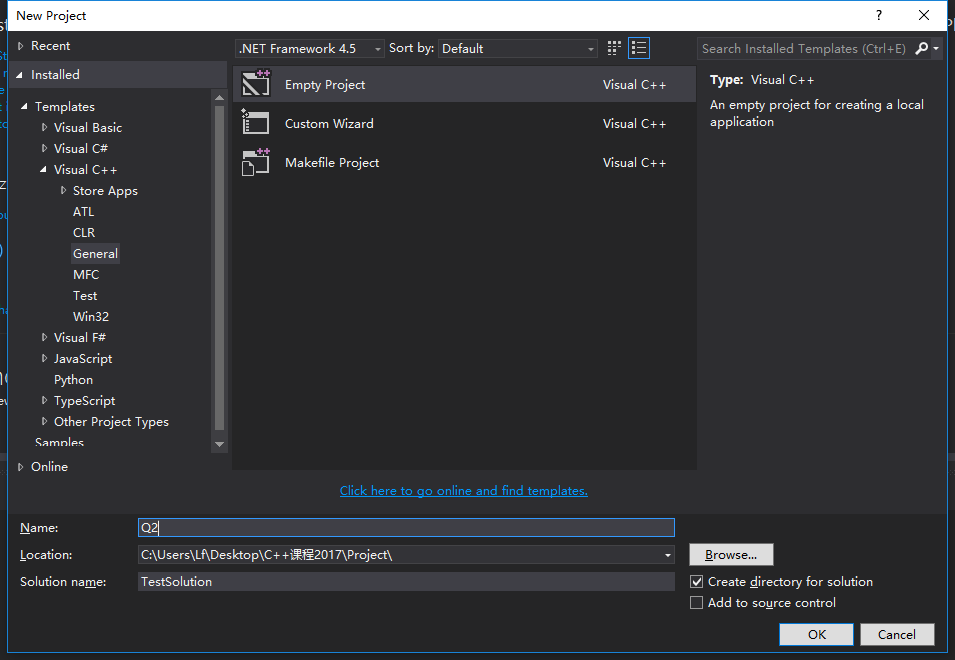
（1）选择菜单栏的File项，选择New->Project，弹出新建项目窗口。



（2）在弹出的窗口中，左侧为项目模板类型，右侧为具体模板名称，下方分别表示新建Project的名称、Solution的位置、Project所属的Solution、Solution的名称。

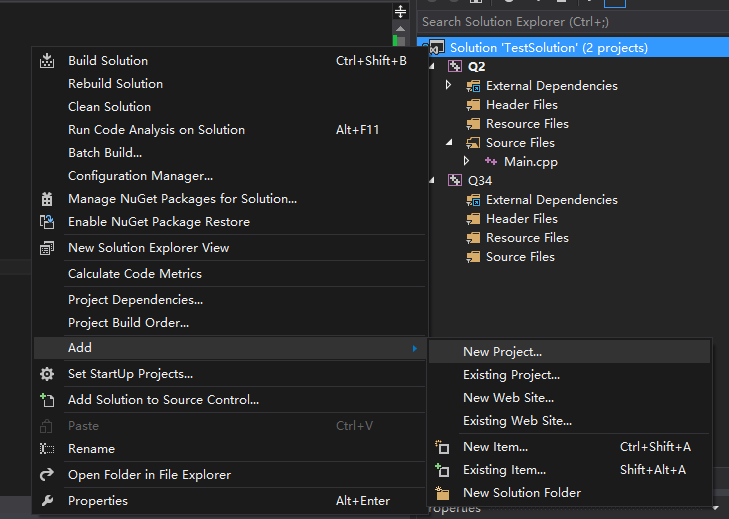
（3）左侧选择Installed->Templates->Visual C++->General，右侧选择Empty Project（Visual C++）。下方填写项目名称与解决方案名称，位置选择解决方案所在位置。Solution名称没有要求，Project名称应与对应题目的名称（Q\*）相同。

（4）点击OK完成新建任务。

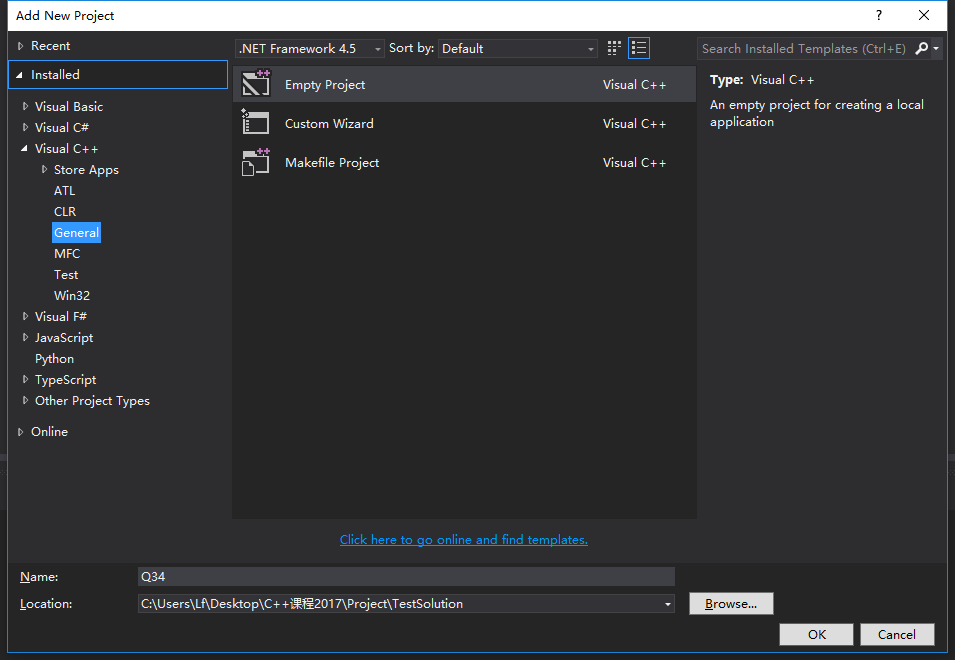


5.1.2 在解决方案中新建项目

（1）右键点击解决方案，选择Add->New Project，弹出新建项目窗口。



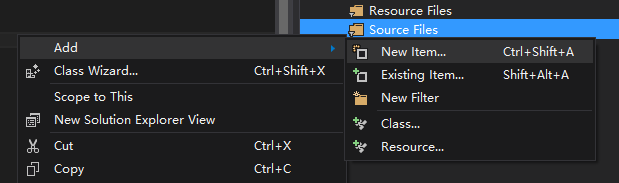
（2）左侧选择Installed -> Visual C++ -> General，右侧选择Empty Project。下方填写项目名称，项目位置采用默认位置。项目名称需要与题目的名称相同（格式为Q\*）



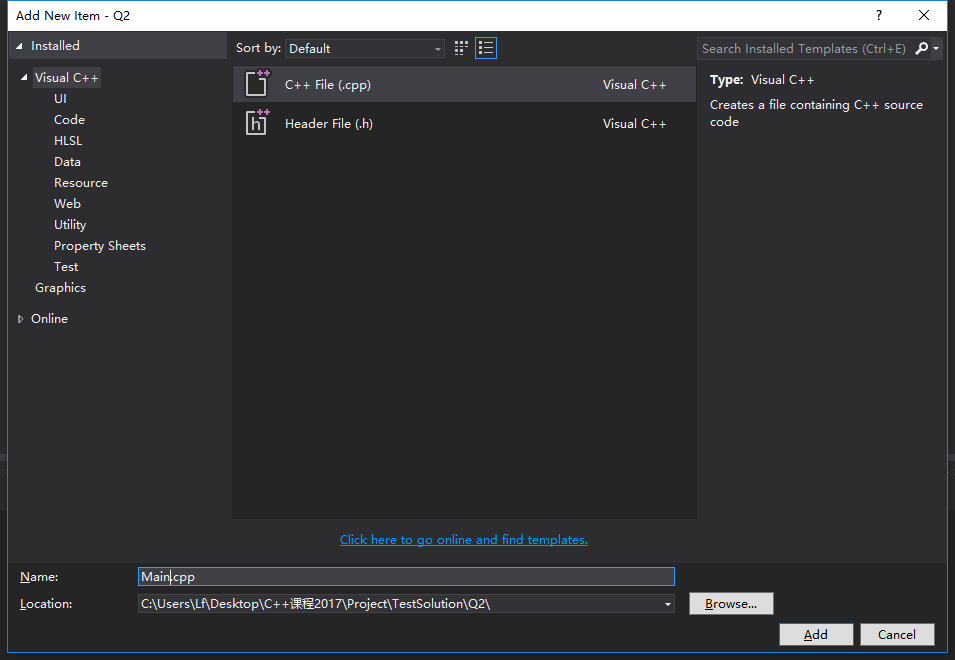
（3）选择OK完成新建项目。

5.1.3 在项目中添加文件

（1）在项目中的Source Files目录（若是.h文件，则为Header Files）右键，选择Add -> New Item，弹出新建文件窗口

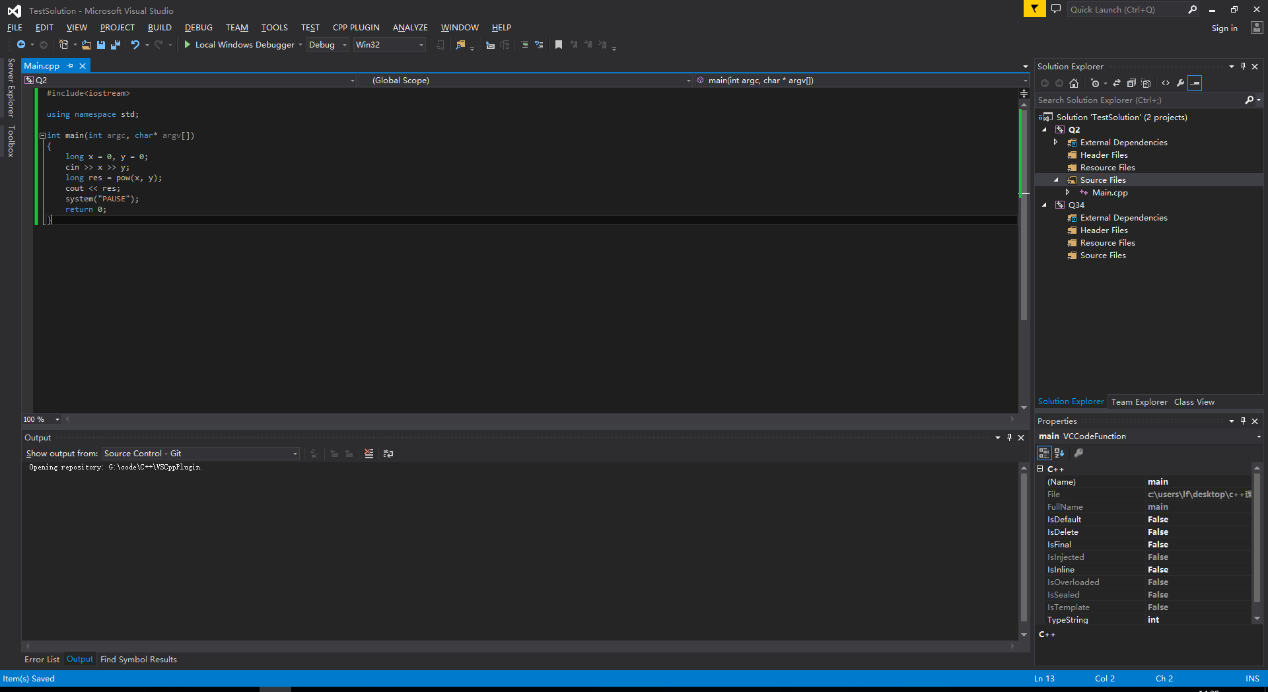


（2）窗口左侧为文件分类，右侧为具体文件类型，下方为文件名称。左侧选择Installed -> Visual C++。右边选择C++ File（若为头文件，则选择Header File）。下方填写文件名，位置采用默认。



（3）选择Add完成新建文件。

### 5.2 Visual Studio编码界面介绍



图中显示为基本的Visual Studio编码界面。左侧从上至下分别为菜单栏、工具栏、文件编辑框、输出等工具框，右侧从上至下分别是解决方案浏览器，文件属性框。编码过程与调试过程，主要使用文件编辑框、解决方案浏览器与输出等工具框。

菜单栏上包括VS的选项菜单，包括VS中所有的功能。主要使用的是File、Build、Debug、Tool、Window。具体使用可以网上查询

工具栏包括目前可以进行的操作的快捷方式。工具栏分为2种状态，分别为编辑模式与调试模式。编辑模式下，工具栏提供搜索保存等文件编辑的工具。调试模式下，工具栏提供继续、停止、单步调试等功能。

文件编辑框用于显示打开的文件列表和正在编辑的文件，在此框中进行编码。

输出等工具框用于显示输出、错误列表、查找结果、局部变量、调用栈等信息框。输出框显示编译的输出结果。错误列表用于显示编译时的错误信息。局部变量显示调试时所处函数的局部变量。

解决方案浏览器用于显示当前解决方案的文件的层级结构。在浏览器中主要进行对项目与文件的操作。

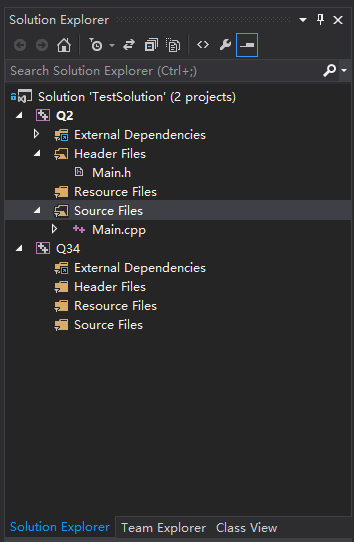
文件属性框用于显示当前文件的配置属性。

### 5.3 解决方案层级结构

在解决方案浏览器中，显示了当前解决方案的层级结构。解决方案类似于Eclipse的工作空间。在一个解决方案下可以存在多个相同或不同类型的项目。如图中解决方案为TestSolution，在这个解决方案下存在2个项目，名字分别为Q2与Q34。

通常一道题目是一个项目，项目中分为4个目录，External Dependencies、Header Files、Resource Files、Source File。cpp代码文件放于Source Files目录中，h头文件放于Header Files目录中，资源文件放于Resource Files中。所有目录中都可以再嵌套自定的目录结构。

下图展示了一个题目的解决方案层级结构。



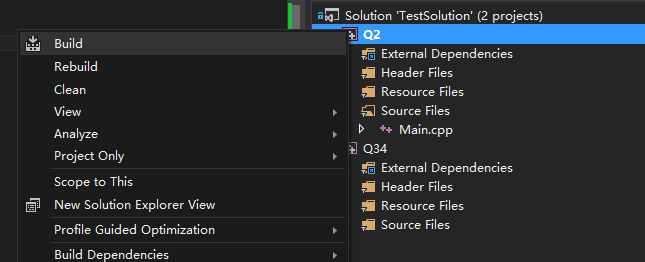
### 5.4 编码

新建Cpp文件，双击文件后，在左侧的编码框直接编写。为了调试方便，可以在main函数的结尾括号处打上断点，让程序执行完成后停止在断点，查看程序的输出。点击工具栏的continue选项，让程序正常退出。

### 5.5 编译与错误窗口

（1）编译

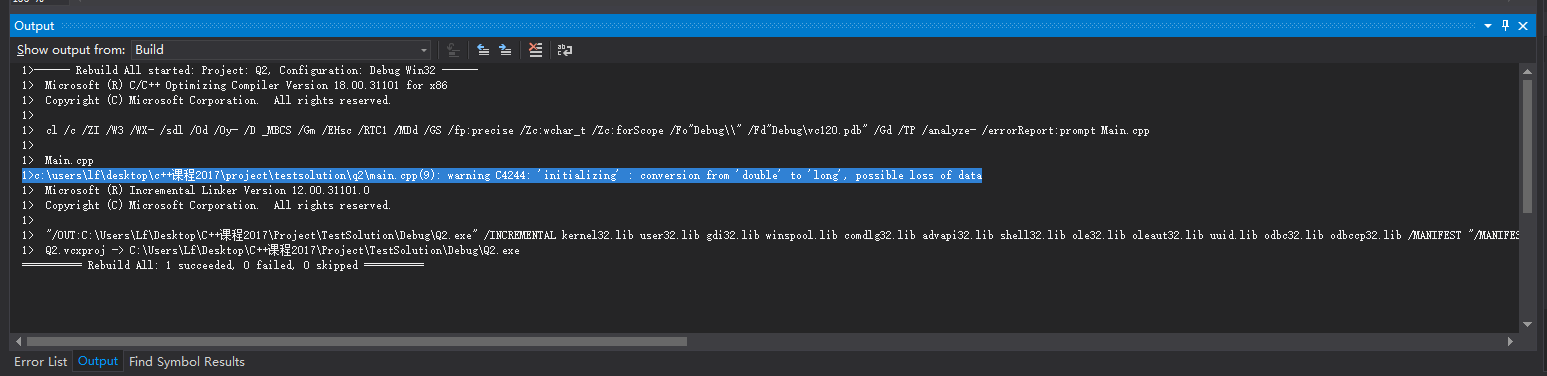
在需要编译的项目上右键，选择Build或Rebuild进行编译。在解决方案上右键选择Build与Rebuild将编译解决方案下全部的项目。编译的输出将显示在下方的Output框中，如果发生错误，会将错误列在Error List框中。



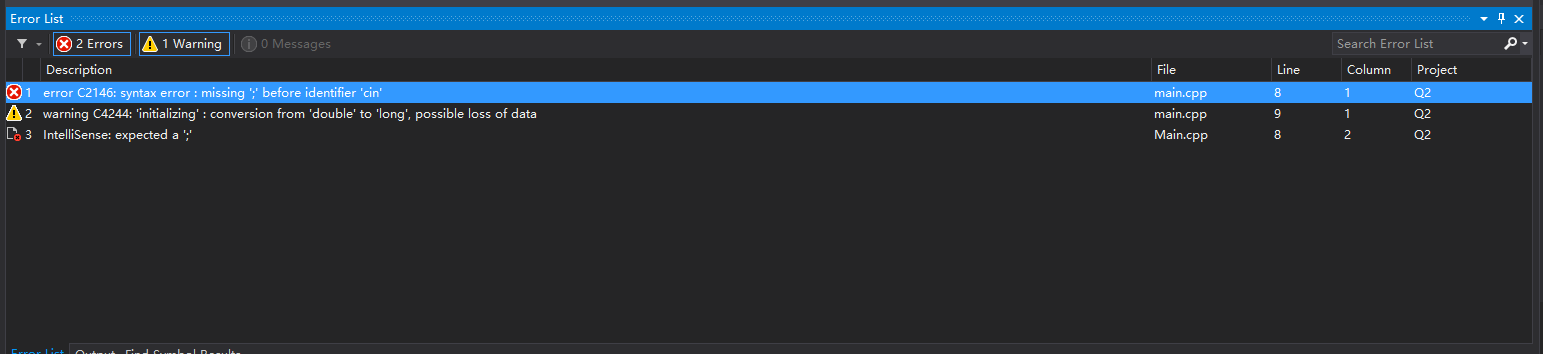
（2）编译输出

编译的输出将显示在下方的Output框中，如果发生错误，会将错误列在Error List框中。

若编译成功，将显示编译的Output，上面列举了编译的详细信息。



若编译发生错误，下方将显示Error List框，上面以红色列举了出现错误的位置。双击错误项可以直接跳转到对应代码行。



对错误进行修改之后，重新进行编译即可。

### 5.6 调试操作与界面

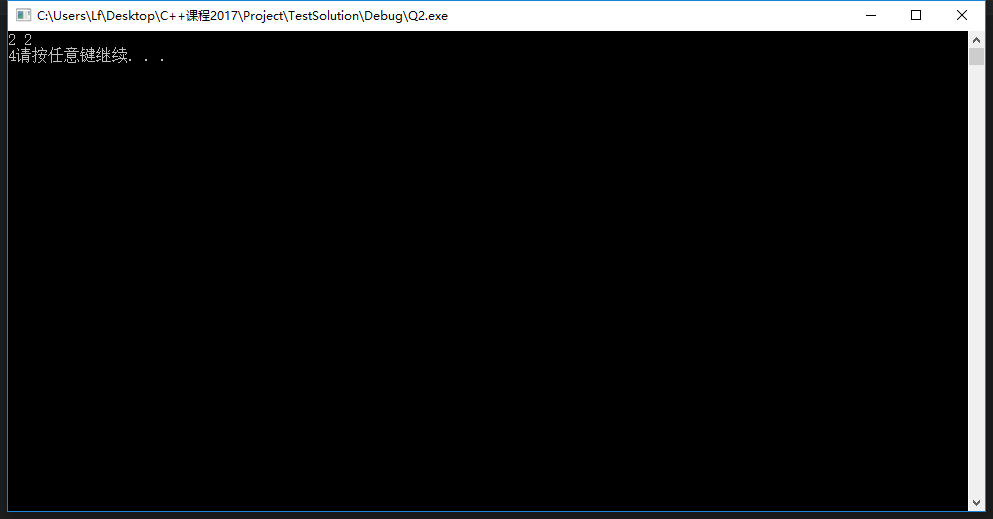
（1）直接运行

与其他IDE一致，在代码行的前方单击后，将出现红点表示设置为断点。若不设置断点，点击工具栏中的Local Windows Debugger则以Debug模式直接运行。

为了调试方便，可以在main函数的结尾括号处打上断点，让程序执行完成后暂停而不直接退出。之后选择工具栏的continue选项，让程序正常退出。

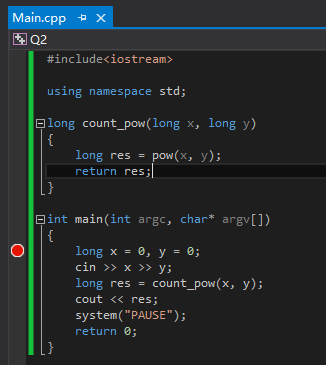


运行时，将弹出控制台窗口，在窗口中进行对应的输入，控制台将显示对应的输出。



（2）单步调试

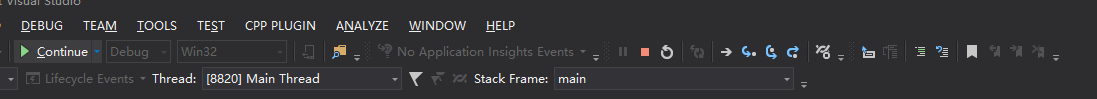
在编译完成后，在代码行的前方单击后，将出现红点表示设置为断点。代码如下。为了调试方便，可以在main函数的结尾的括号处打上断点，让程序执行完成后暂时而不直接退出。之后选择工具栏的continue选项，让程序正常退出。

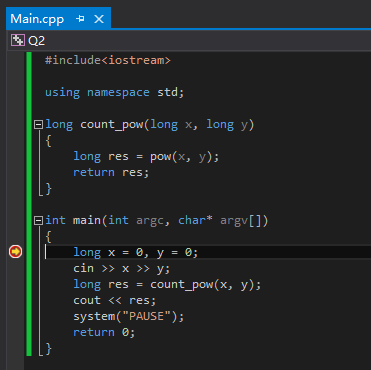


设置完成断点，点击工具栏中的Local Windows Debugger开始进行调试。



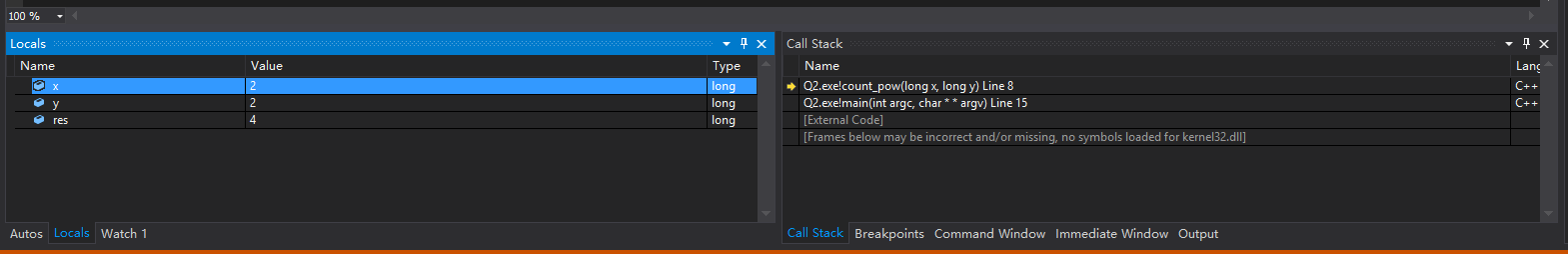
开始调试之后，程序将会在断点停下来，工具栏将变成调试模式，增加了调试的操作。同时，代码的前端出现一个箭头表示下一步将要执行的语句。





工具栏提供了逐语句调试（Step Over）、进入函数（Step Into）、离开函数（Step Out）、终止（Stop）、继续（continue）等操作。

在调试过程的暂停时间，下方将显示当前函数的局部变量的值。也可以通过Watch，自己输入变量名搜索。将光标移动到变量上，也可以显示变量的值。



调试过程中，在设置的断点后可以进行单步调试，可以选择继续（Continue），让程序继续执行，也可以点击终止（Stop）提前终止调试。

## 六、插件操作过程演示

1 安装插件，双击安装包，根据提示下一步即可。

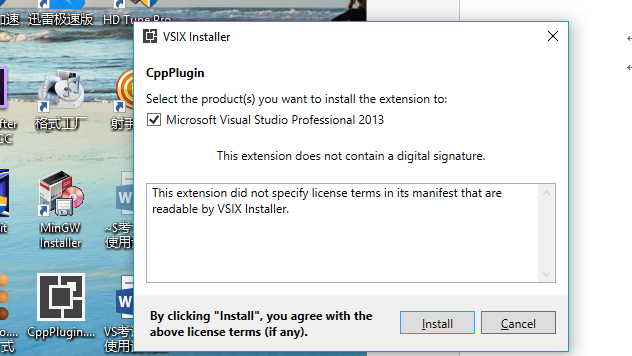


图2 操作演示-安装

2 点击[CPP PLUGIN] > [login]按钮，输入账号密码登陆。登录成功如图3。

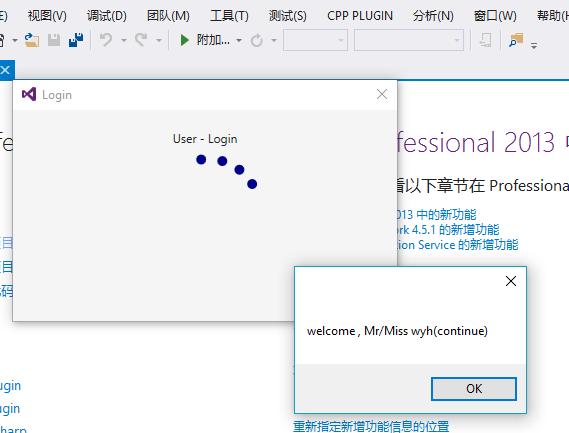


图3 操作演示-登录

3 点击点击[CPP PLUGIN] > [download]按钮按钮，选择需要下载的考试和存放考试说明文件的文件夹。下载完成后，将弹出下载的文件夹。

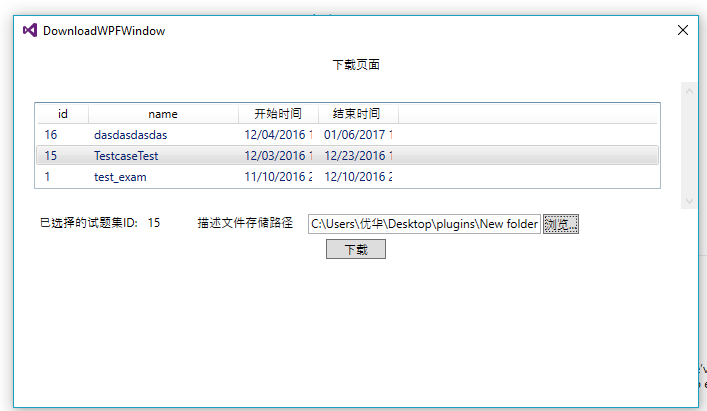


图4 操作演示-下载

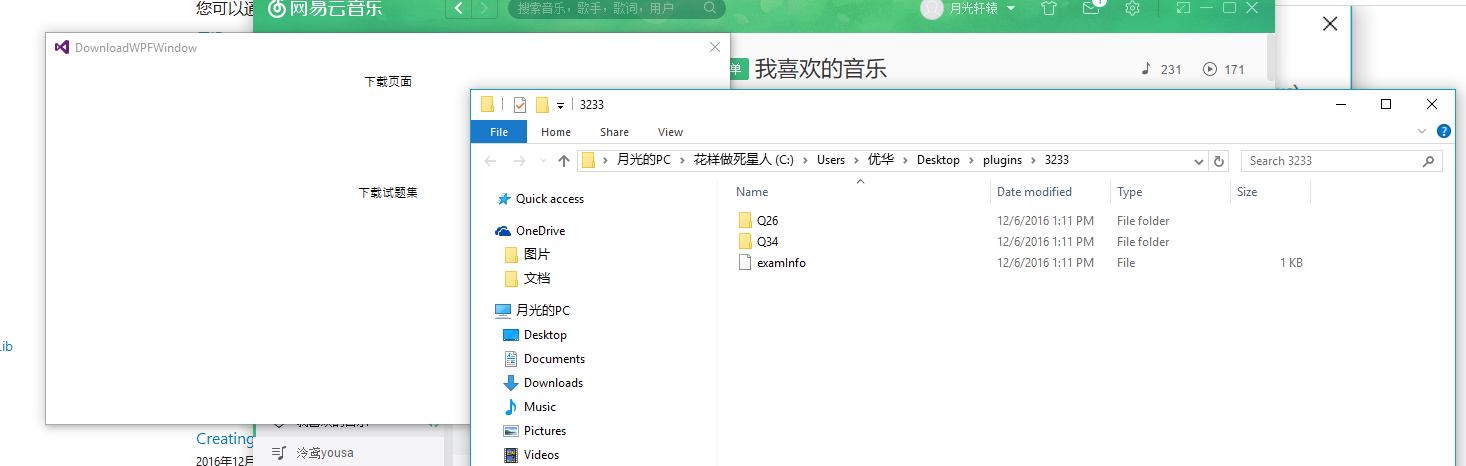


图5 操作演示-下载结束弹出文件夹

3 按要求编写程序(同一次练习/考试相关的项目放入同一个解决方案中)

* 1. 新建解决方案
  2. 添加项目

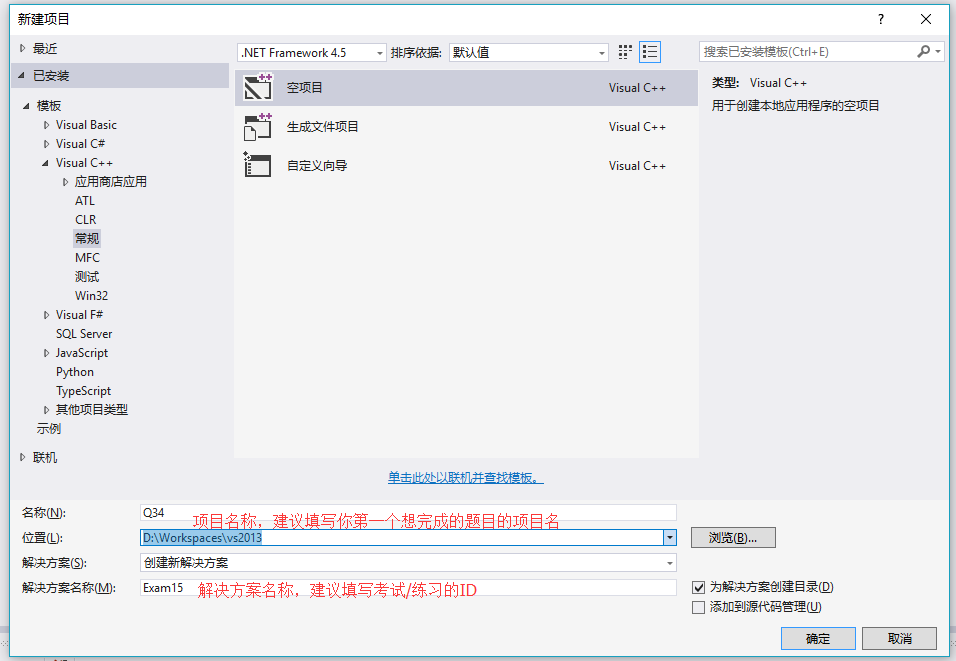


图6 操作演示-新建解决方案

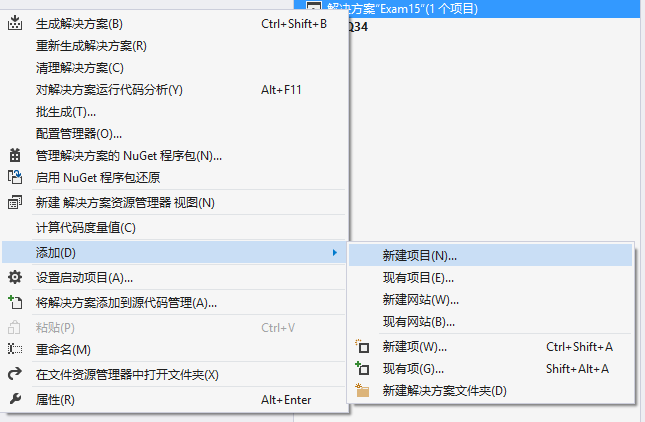


图7 操作演示-添加项目

* 1. 新建文件，并编写的项目内容
  2. 设置想要运行的项目为启动项目

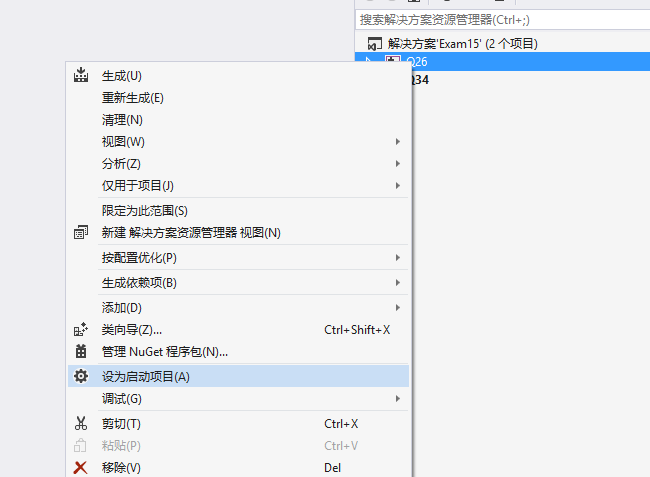


图8 操作演示-设置启动项目

* 1. 选择编译Debug ，点击[本地Windows调试器]，运行项目。调试完成后进行Test和Upload。

4点击[CPP PLUGIN] > [test]按钮按钮，选择需要测试的题目。测试结果将实时显示。

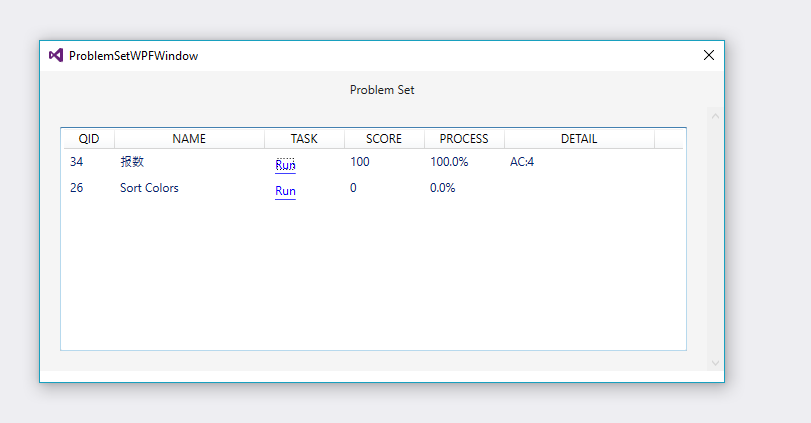


图9 操作演示-测试

5点击[CPP PLUGIN] > [upload]按钮。获得上传成功的提示后即完成上传

6 点击[CPP PLUGIN] > [logout]按钮，登出账户，结束考试/练习。

## 七、注意事项

1. 下载考试题目的位置要求为一个空目录，位置无要求
2. 一道题目建立一个C/C++项目，项目名称为题目编号
3. 同一次考试的所有题目写在同一个解决方案中，解决方案名称无要求
4. 一个解决方案中有多个项目，默认只启动其中一个，解决方案的选项中可以选择以某个项目启动。
5. 请将项目编译为Debug版本。插件的测试是基于Debug版本的，避免编译器优化
6. 请在编译完成后再进行测试，在测试完成后再提交结果，否则将无法记录成绩
7. 消息弹窗可能不在最上层。请切换窗口检查提示弹窗
8. 若在练习或考试过程中，VS出现错误而崩溃，请直接重启VS。请直接关闭VS并重新打开，重新登录，确认自己处在正确的练习中后，继续练习。插件将保存用户的练习与考试状态。
9. 请勿随意修改输入流。修改输入流可能造成测试结果不正确。