



## 实验UNIT 01

# C++上机基本操作指导

《程序设计》课程组



# C++编程集成开发环境

建议使用的C++编程环境：

- Visual Studio 2015（或更新的版本）
  - <https://visualstudio.microsoft.com/zh-hans/vs/older-downloads/>
- Eclipse IDE for C/C++
  - <https://www.eclipse.org/downloads/packages/>
- 华为鲲鹏云平台（自愿选择）



# 第一部分

## Visual Studio



# Visual Studio 2015

## ◆ 简介

Visual Studio 2015是美国微软公司出品的一套完整的大型软件开发工具集，它包括了整个软件生命周期中所需要的许多工具，如集成开发环境(IDE)、UML软件建模工具、代码管控工具、测试工具等。其中的Visual C++ 组件用于支持C++语言编程开发

Visual Studio开发的目标代码适用于微软支持的所有平台，2017版可全面支持面向Android、iOS、Windows、Web 和云平台的软件系统开发

目前Visual Studio针对不同用户类型有多个产品版本（Community、Professional、Enterprise），Visual Studio Community版可供学生或个人免费使用，下载地址：

<https://visualstudio.microsoft.com/zh-hans/downloads/>

[http://download.microsoft.com/download/B/4/8/B4870509-05CB-447C-878F-2F80E4CB464C/vs2015.com\\_chs.iso](http://download.microsoft.com/download/B/4/8/B4870509-05CB-447C-878F-2F80E4CB464C/vs2015.com_chs.iso)





# Visual Studio 2015

## ◆ 安装环境

### 1. 支持的操作系统:

Windows 7 Service Pack 1, Windows 8.1, Windows 8, Windows Server 2008 R2 SP1, Windows Server 2012, Windows Server 2012 R2, Windows 10

### 2. 硬件要求:

最低配置: 1.6 GHz 或更快的处理器, 1 GB RAM (如果在虚拟机上运行, 则为 1.5 GB), 4 GB 可用硬盘空间

### 3. 安装参考:

<https://docs.microsoft.com/zh-cn/visualstudio/install/install-visual-studio-2015?view=vs-2015>

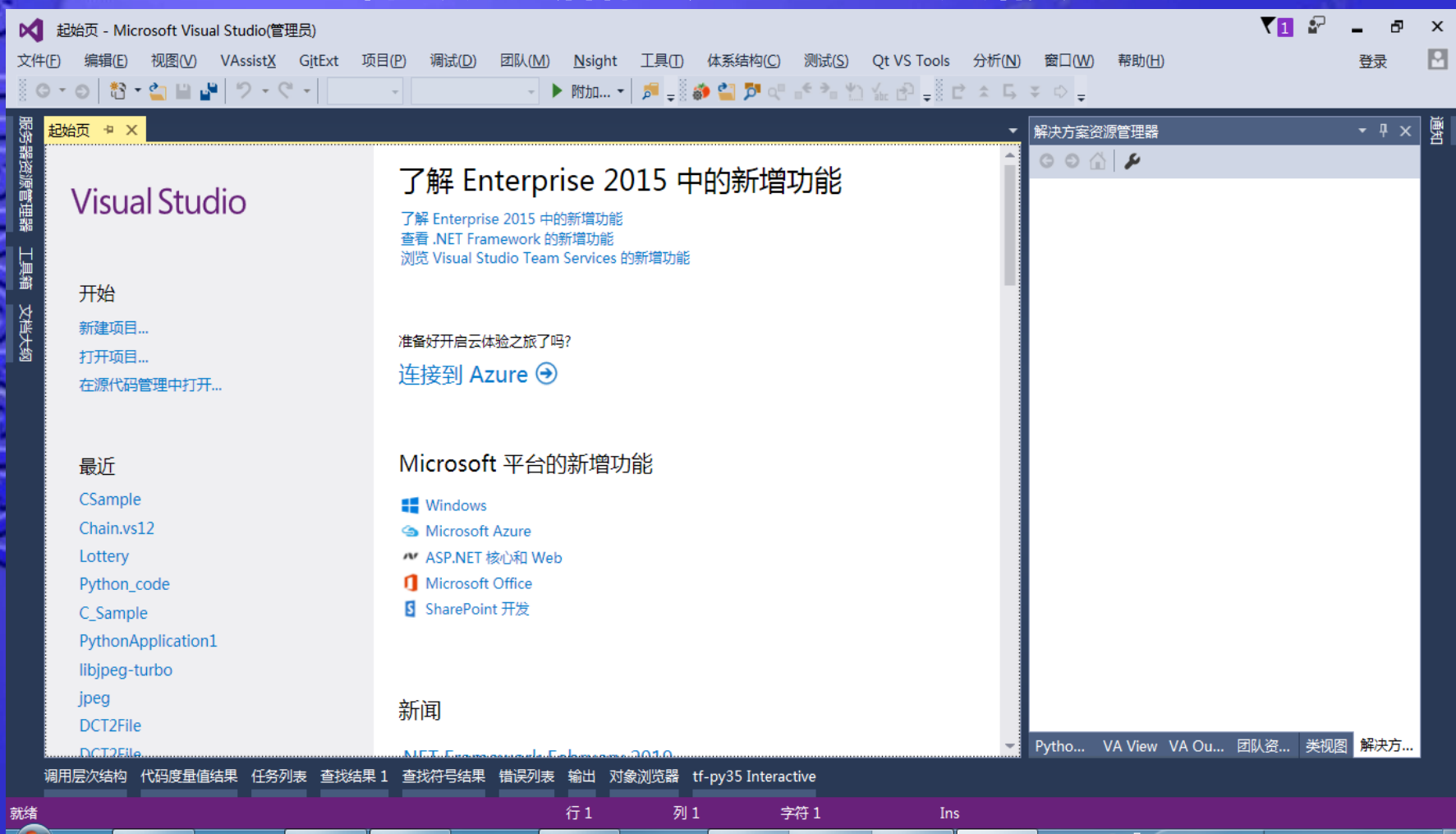
<https://docs.microsoft.com/zh-cn/visualstudio/install/install-visual-studio?view=vs-2017>



# Visual Studio 2015 使用方法

## 1.启动VS2015

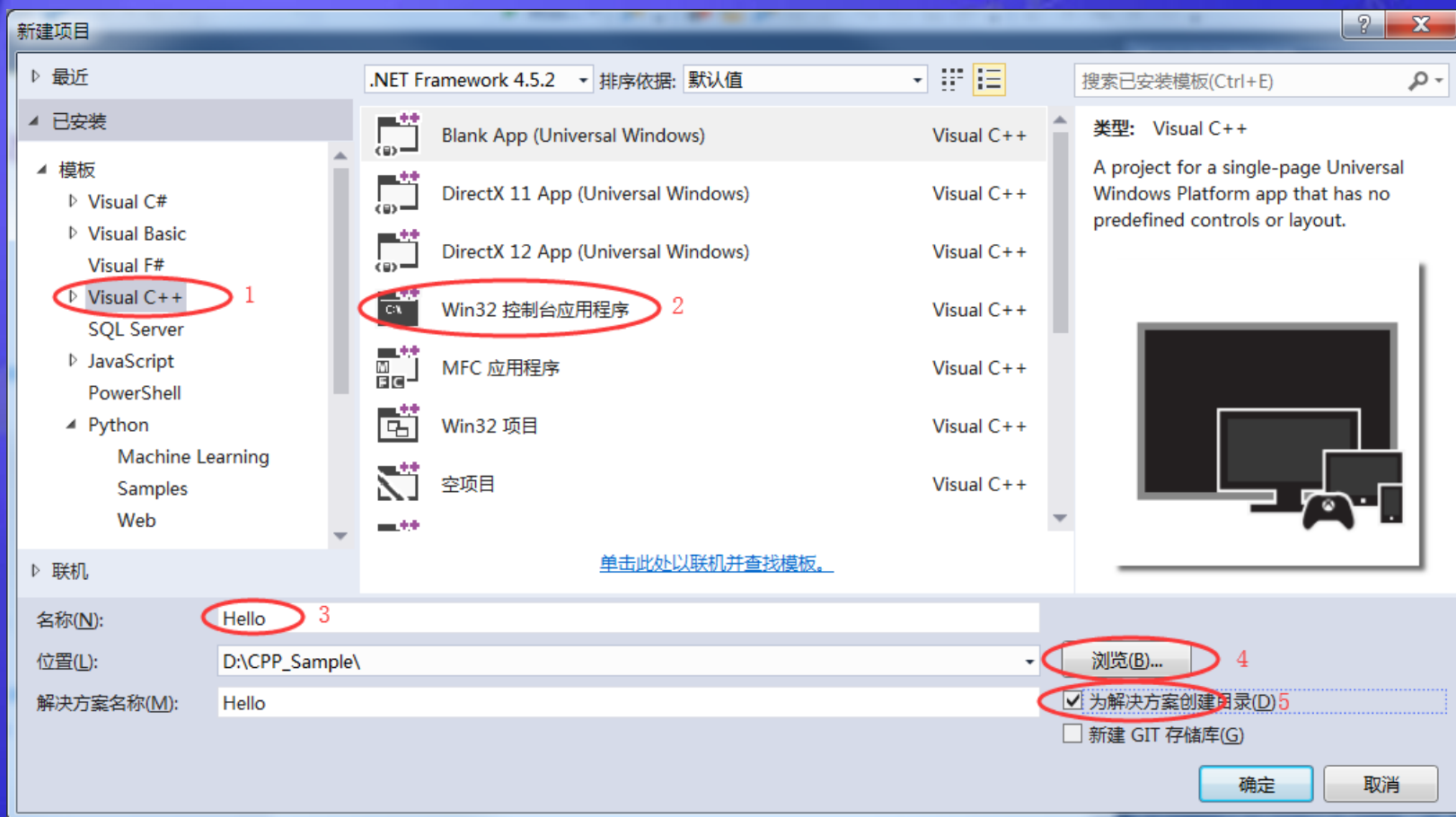
从Windows桌面的“开始”按钮中的菜单组里选择“Visual studio 2015”,启动运行后将打开VS2015的起始页



# Visual Studio 2015 使用方法

## 2. 创建一个控制台项目

(1) 点击起始页的“创建项目”或“文件”菜单的“新建” | “项目...”，将打开新建项目对话框



# Visual Studio 2015 使用方法

## 2. 创建一个控制台项目

(1) 点击起始页的“创建项目”或“文件”菜单的“新建” | “项目...”，将打开新建项目对话框

操作：

- 1) 项目类型模板选中“Visual C++”
- 2) 项目模板选中“Win32 控制台应用程序”
- 3) “名称”输入框中输入新建项目的名称
- 4) 点击“浏览...”按钮，在打开的“项目位置”对话框中指定项目保存位置
- 5) 可选中“为解决方案创建目录”选项
- 6) 上述操作完成后，点击“确定”按钮，将打开项目向导对话框





# Visual Studio 2015 使用方法

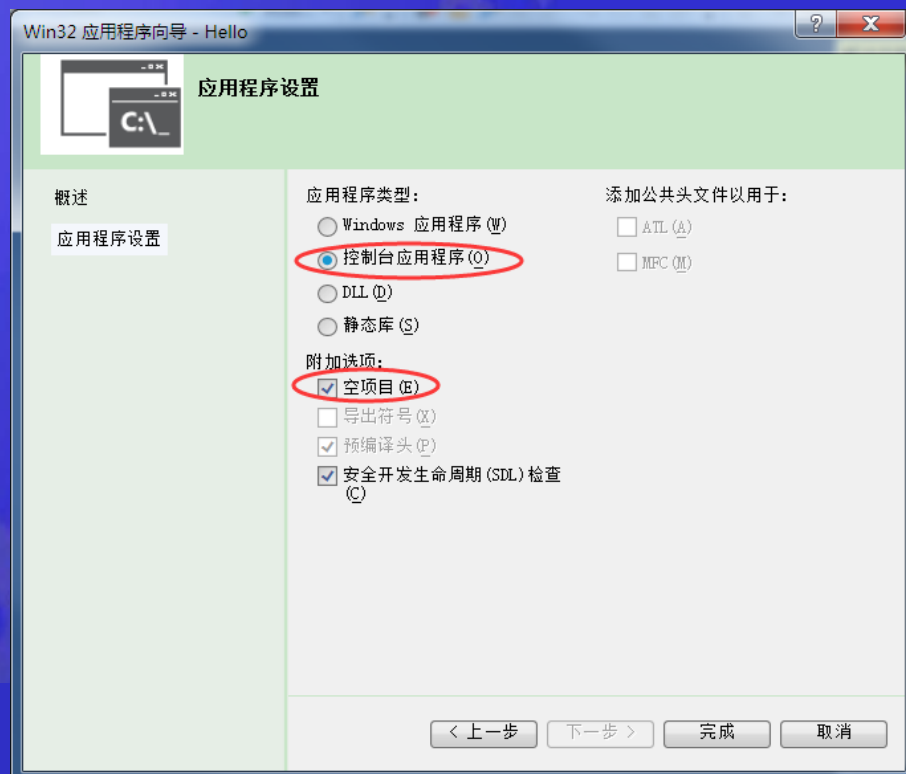
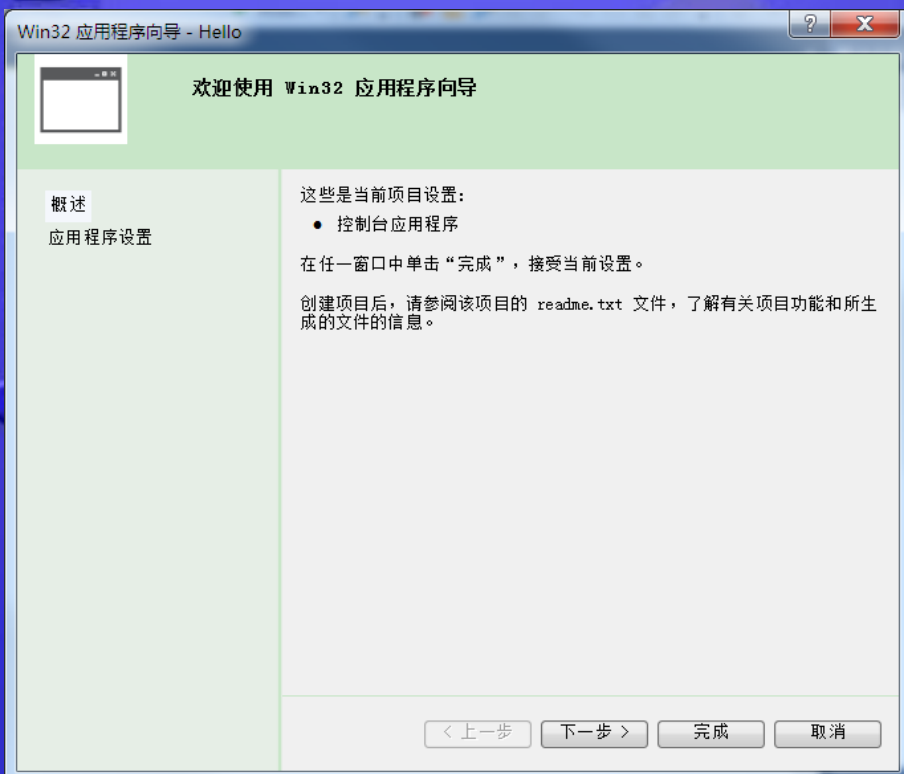
## 2. 创建一个控制台项目

(2) 在弹出的欢迎向导对话框中，直接点击“下一步”按钮，将进入“应用程序设置”对话框。

操作：1) 选中“控制台应用程序” (Console application)

2) 附加选项可选中“空项目” (Empty project)

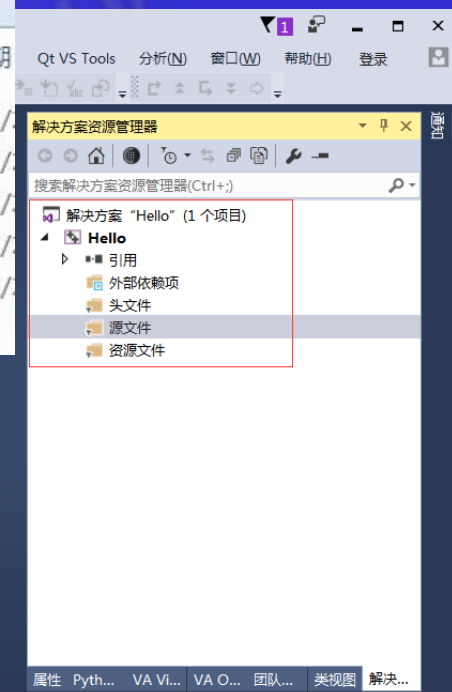
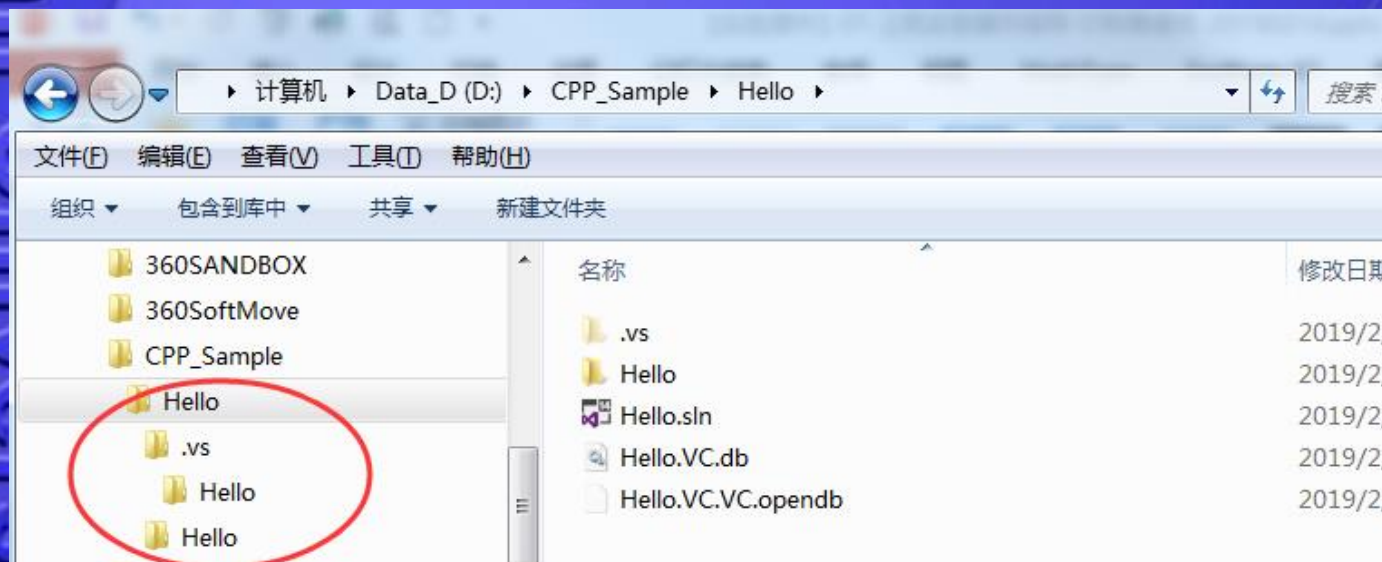
3) 点击“完成”按钮，将建立一个新的项目



# Visual Studio 2015 使用方法

## 2. 创建一个控制台项目

新创建的空项目结果



# Visual Studio 2015 使用方法

## 3. 创建一个C++源程序文件

### (1) 打开添加新项对话框

给项目中添加新的C++源程序，可以有几个不同操作入口打开新建源文件对话框

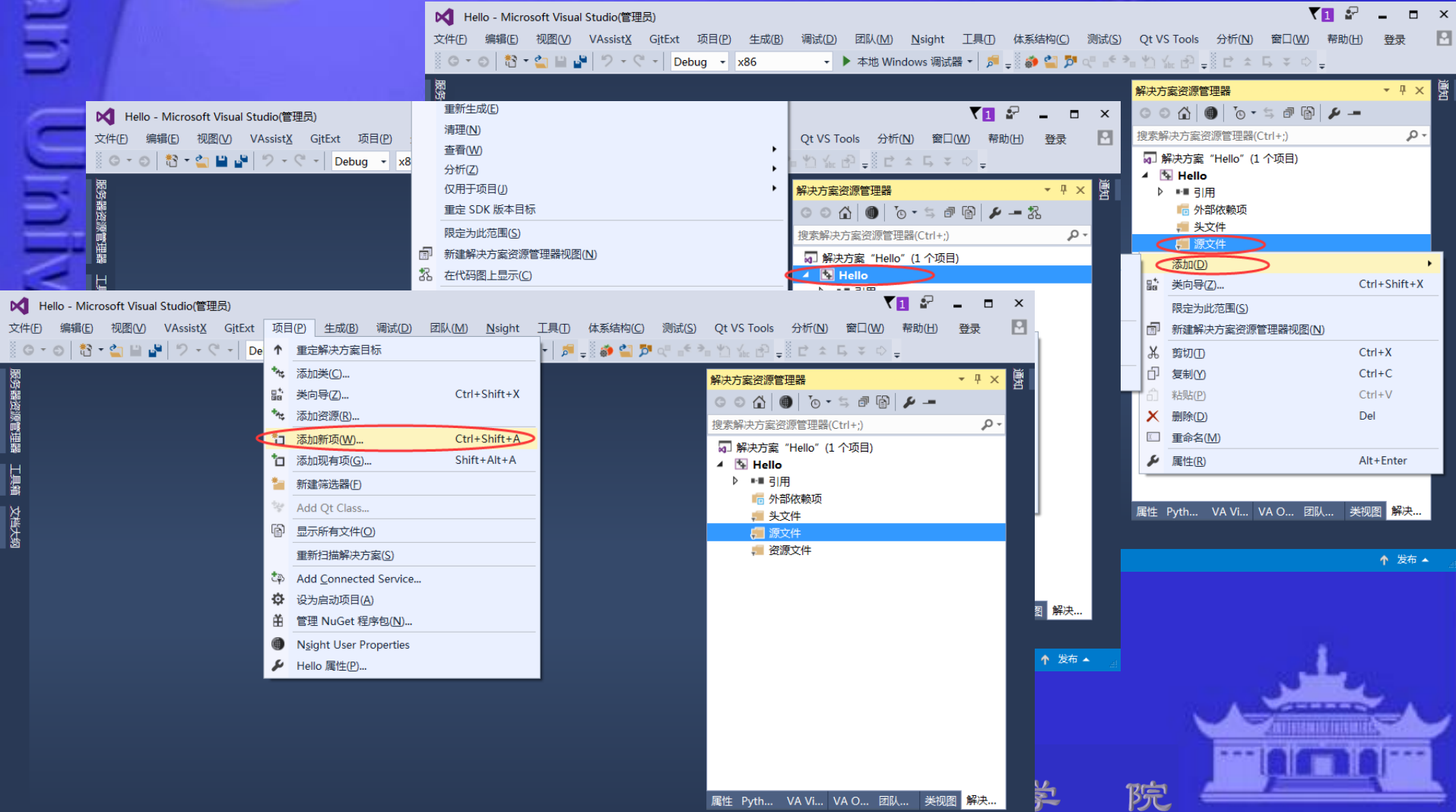
- 1) 选中命令菜单“项目” | “添加新项”
- 2) 在“解决方案”窗口，选中项目“Hello” 点击右键，在弹出菜单项中选中“添加” | “新建项”
- 3) 在“解决方案”窗口，选中“源文件”节点点击右键，在弹出菜单项中选中“添加” | “新建项”



# Visual Studio 2015 使用方法

## 3. 创建一个C++源程序文件

### (1) 打开添加新项对话框

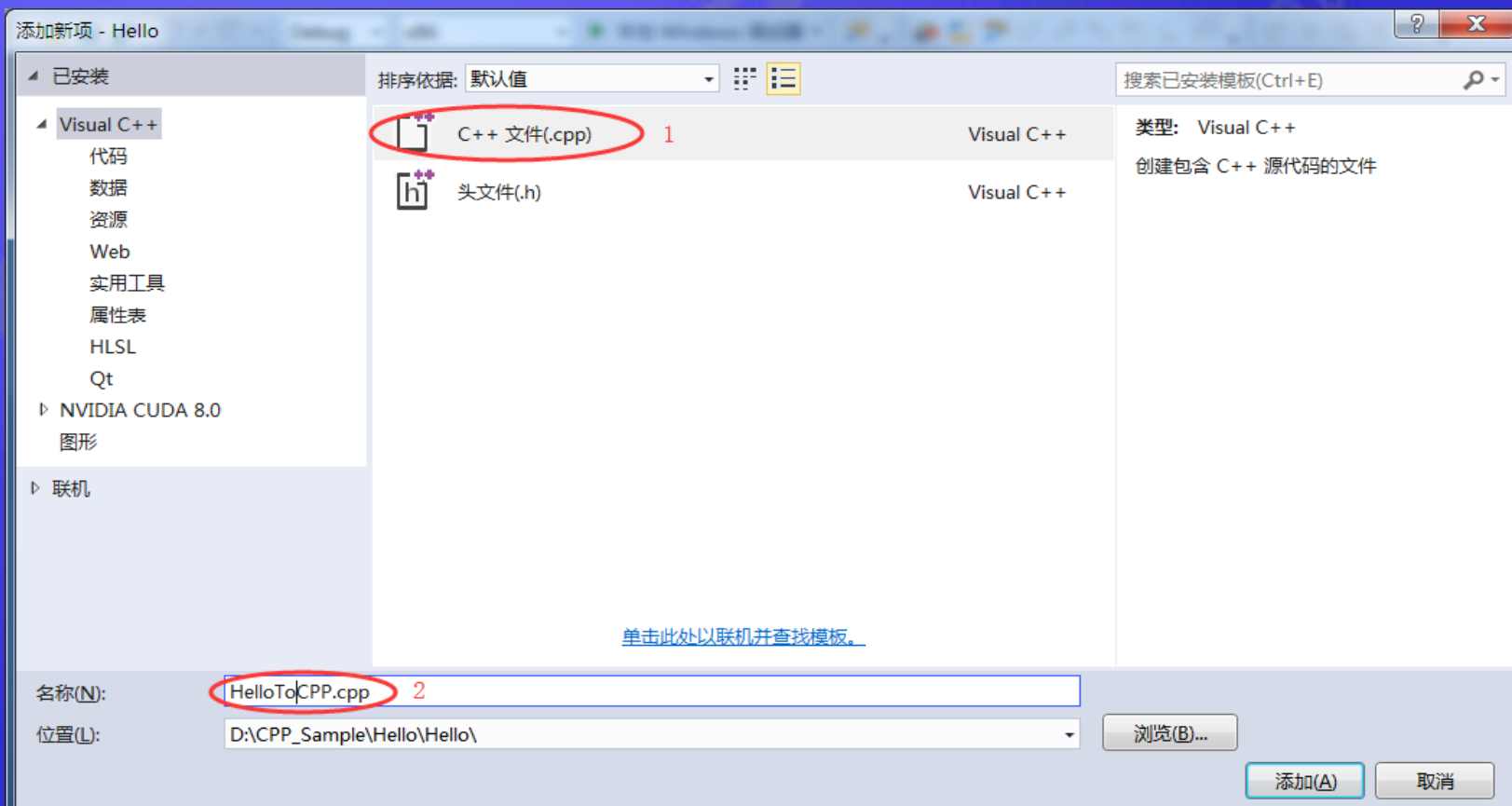


# Visual Studio 2015 使用方法

## 3. 创建一个C++源程序文件

### (2) 添加新项对话框

在添加新项对话框中，首先选中创建的文件类型为“C++文件”，然后在“名称”输入框中填写源程序的文件名称，点击“添加”按钮，完成C++源文件的新建

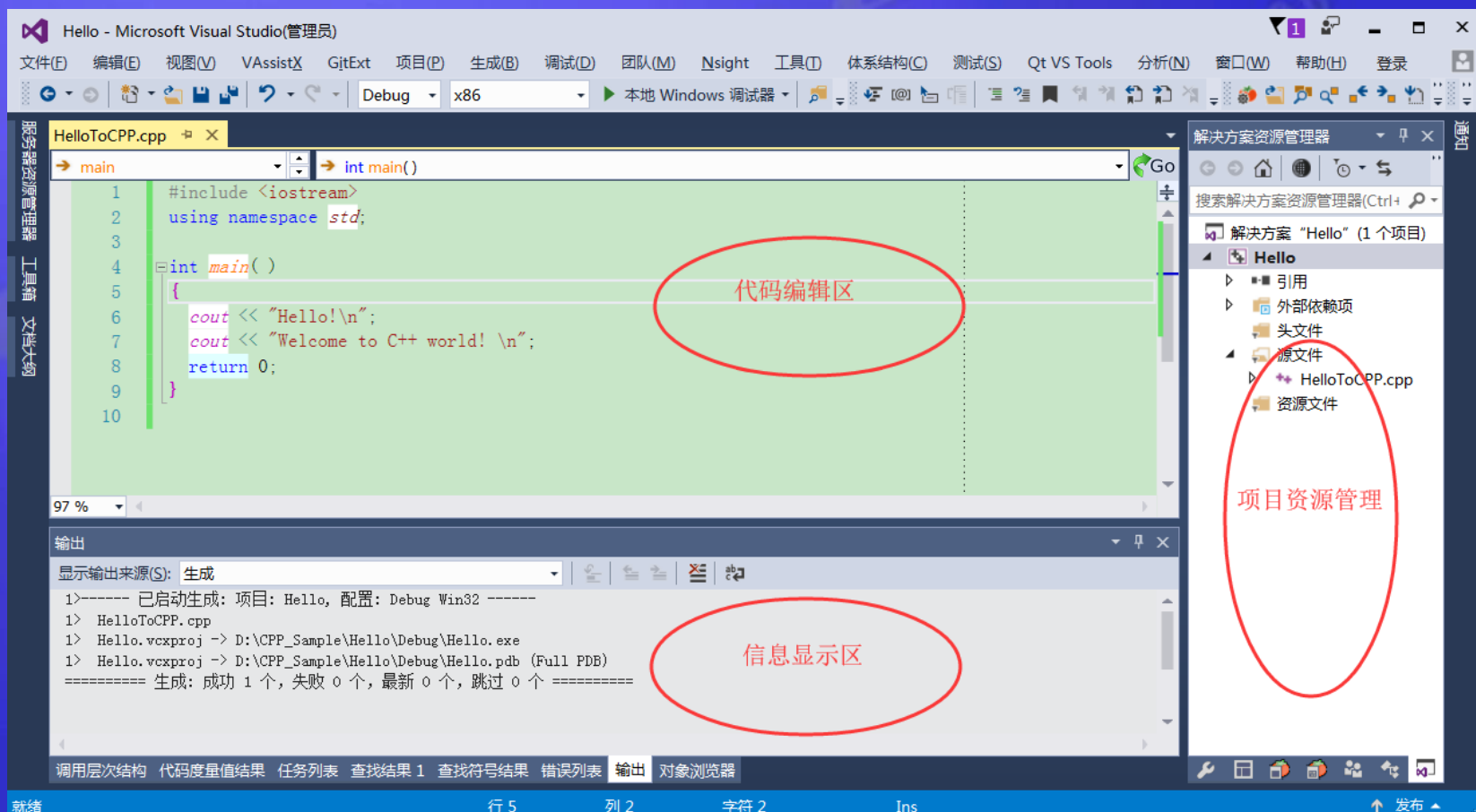




# Visual Studio 2015 使用方法

## 4. 编辑C++源程序文件

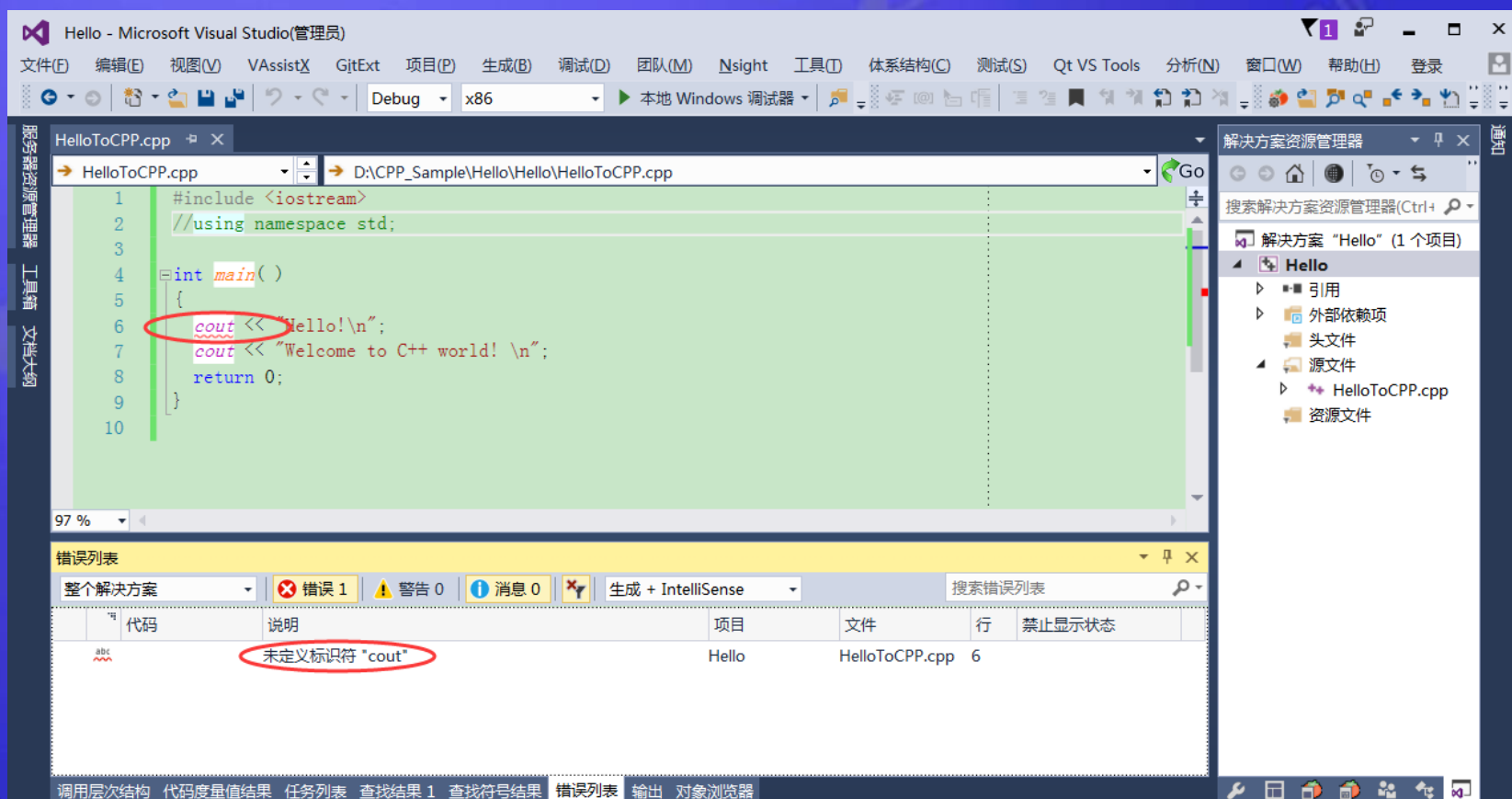
### (1) 在代码编辑器中输入程序代码



# Visual Studio 2015 使用方法

## 4. 编辑C++源程序文件

(2) 代码如果有输入错误，在编辑器中出错的语句处会显示红色波浪线，在“错误列表”信息窗口会显示错误的说明信息，应根据提示的内容和语句位置改正代码错误。

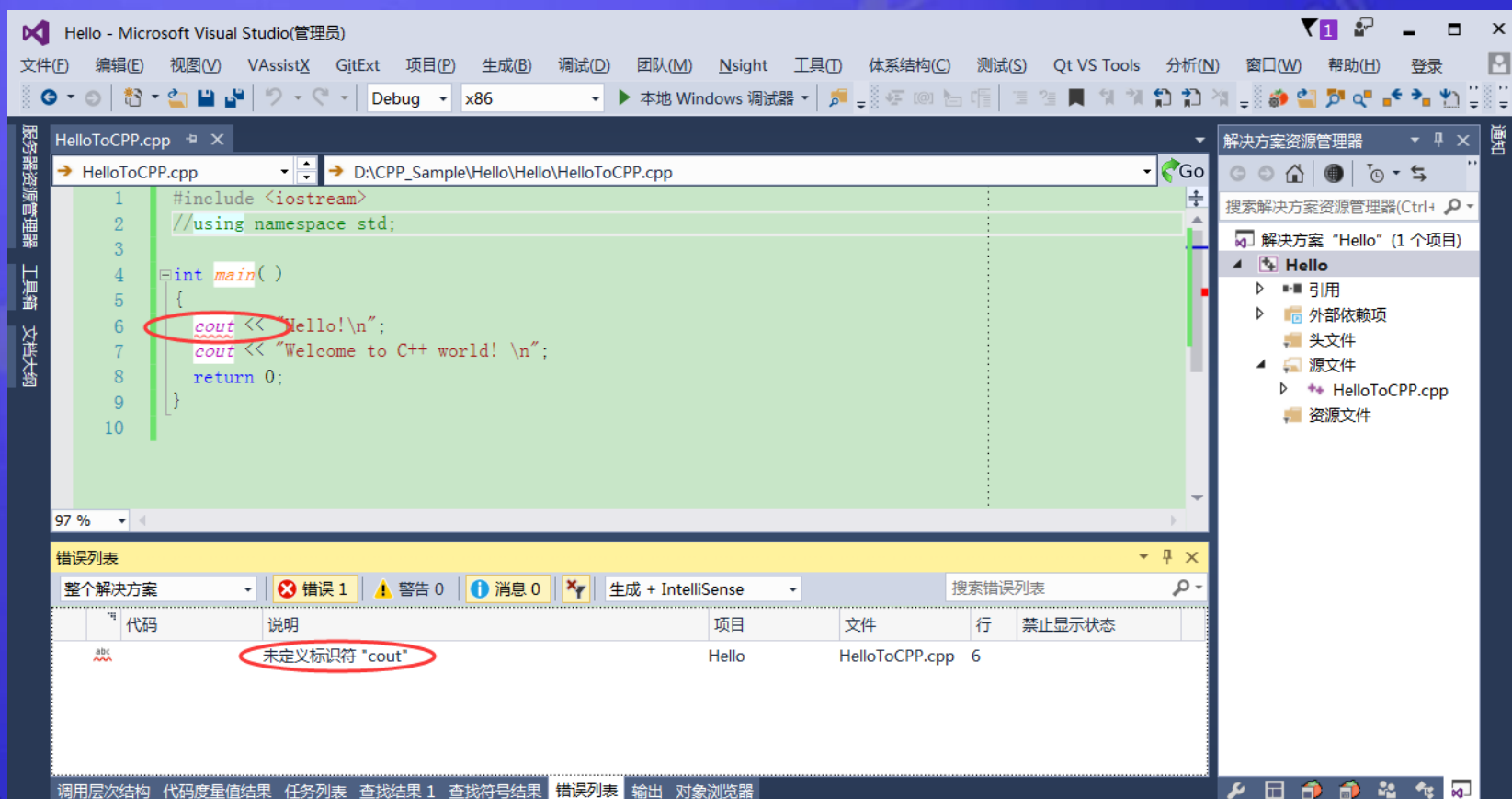


# Visual Studio 2015 使用方法

## 4. 编辑C++源程序文件

### (3) 保存代码

代码编辑完后，选择菜单“文件”|“保存 HelloToCPP.cpp”或按快捷操作键 Ctrl+S，保存所编辑的源程序文件

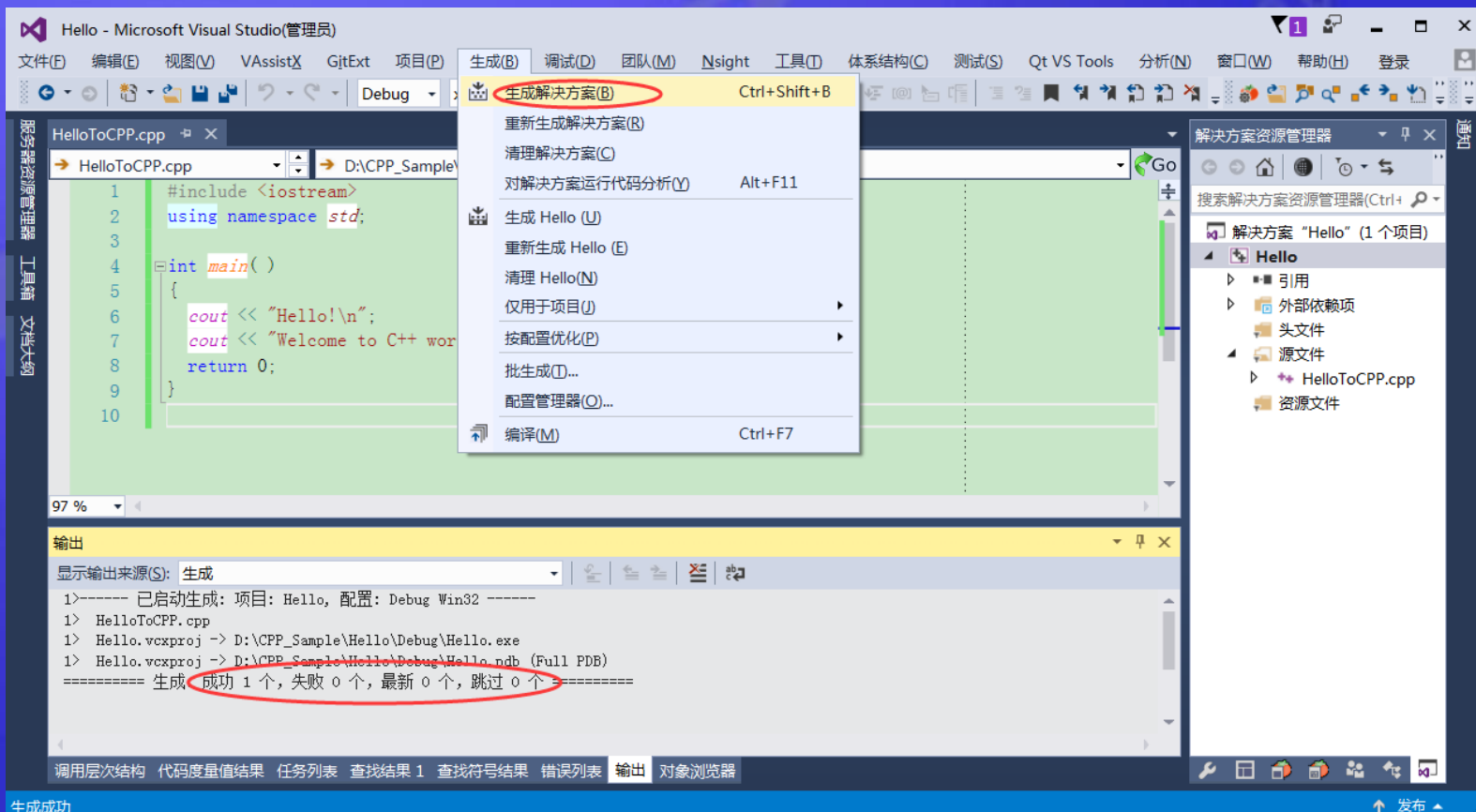


# Visual Studio 2015 使用方法

## 5.生成和运行程序

### (1) 生成可执行程序

选择菜单“生成” | “生成解决方案” 或按快捷键 Ctrl+Shift+B, 生成当前工程中的所有项目；选择菜单“生成” | “生成Hello” 则只生成当前项目的可执行程序。如有错误需修改代码后再重新生成。

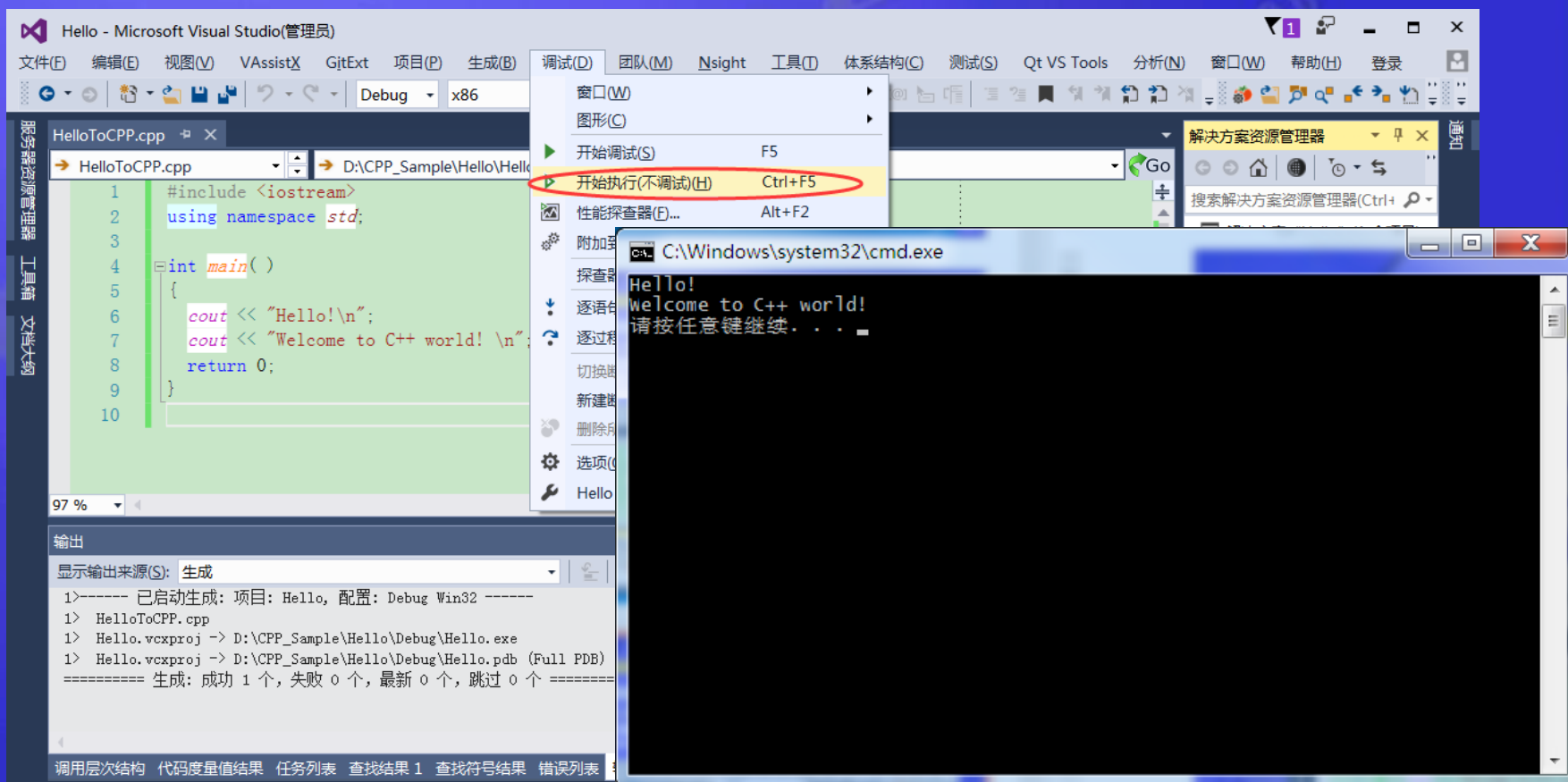


# Visual Studio 2015 使用方法

## 5.生成和运行程序

### (2) 运行程序

执行程序成功生成后，选择菜单“调试”|“开始执行”，或按快捷键 Ctrl+F5，运行程序。运行结果会显示在一个命令窗口中



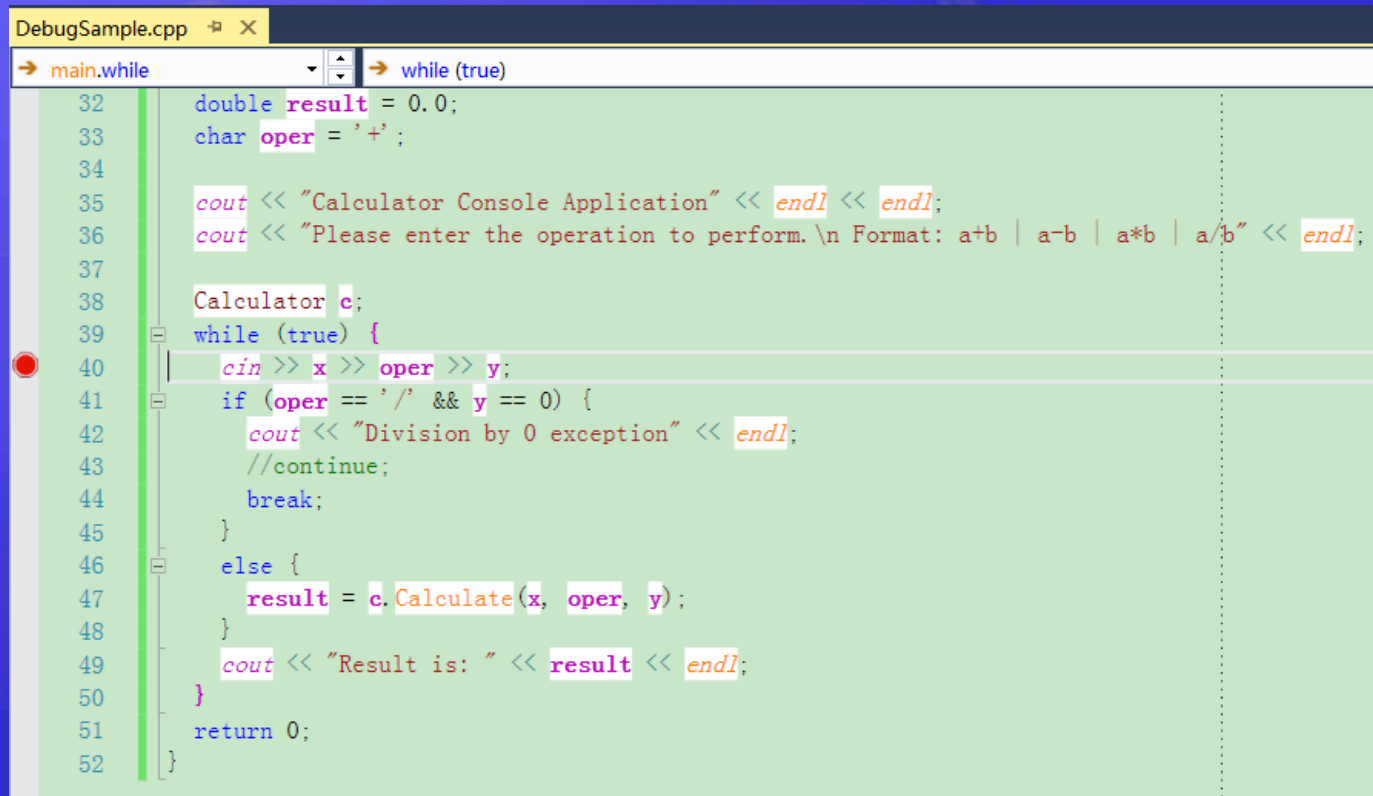


# Visual Studio 2015 使用方法

## 6. 调试程序

若程序执行的结果不正确，可以使用调试器（debugger）工具来辅助我们追踪和定位程序中的逻辑错误。

在调试器工具的主要功能包括：设定断点、逐行执行程序、监视变量的值。许多程序的错误可以利用调试技术迅速地找出来，因而减少排除错误所需消耗的时间和精力。



# Visual Studio 2015 使用方法

将如下代码输入到DebugSample.cpp文件中，演示调试器使用

```
#include <iostream>
#include <stdio.h>
using namespace std;

class Calculator {
public:
    double Calculate(double x, char oper, double y);
};

double Calculator::Calculate(double x, char oper, double y) {
    switch (oper) {
        case '+': return x + y;
        case '-': return x - y;
        case '*': return x * y;
        case '/': return x / y;
        default: return 0.0;
    }
}
```

# Visual Studio 2015 使用方法

```
int main( )
{
    double x = 0.0;    double y = 0.0;
    double result = 0.0; char oper = '+';

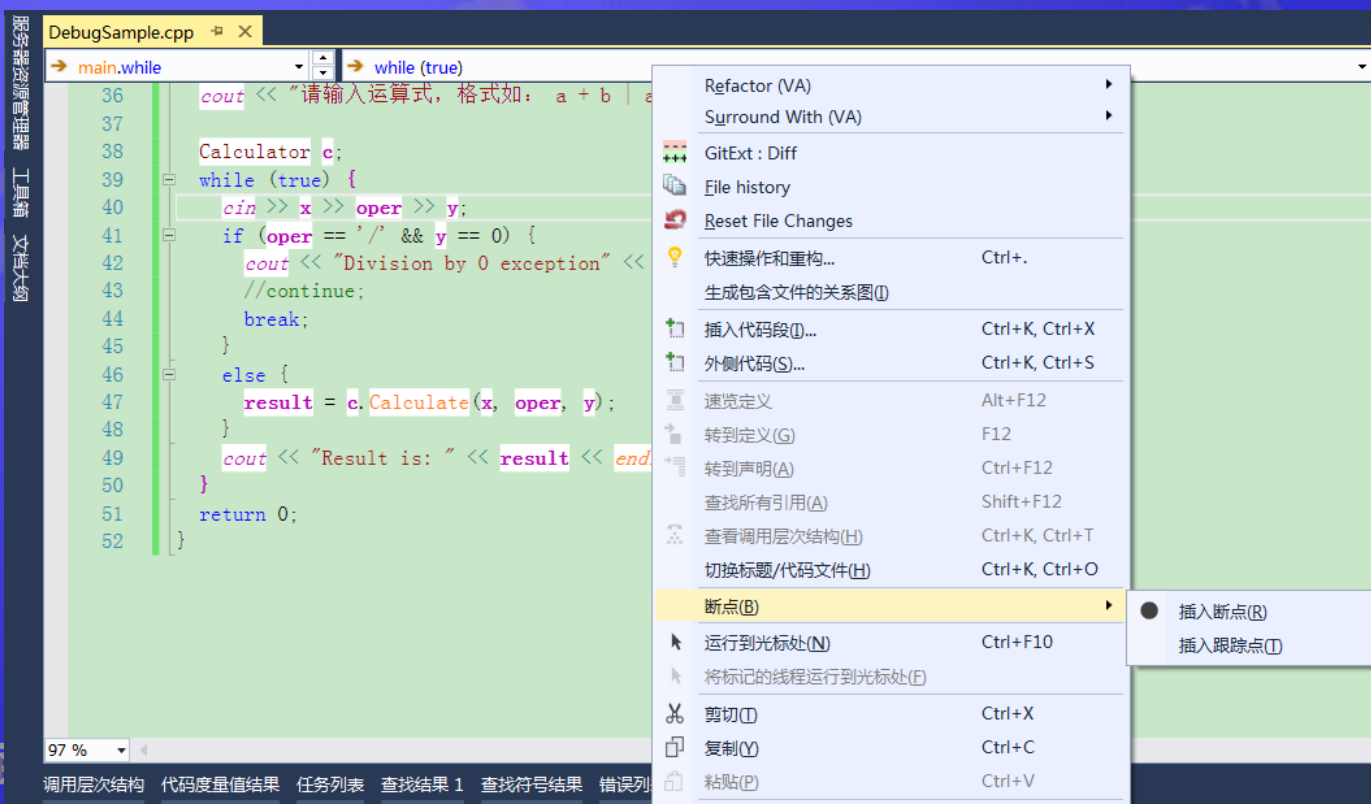
    cout << "算术运算示例程序" << endl << endl;
    Calculator c;
    while (oper != '$') {
        cout << "请输入运算式, 格式如: a + b | a - b | a*b | a / b" << endl;
        cin >> x >> oper >> y;
        if (oper == '/' && y == 0) {
            cout << "Division by 0 exception" << endl;
            continue;
        }
        else {
            result = c.Calculate(x, oper, y);
        }
        cout << "Result is: " << result << endl;
    }
    return 0;
}
```

Wuhan University

## (1) 设置断点

**断点设置的位置在怀疑的错误语句之前，程序运行到断点处将暂停，然后逐条语句依次执行，查看相关变量的值，判断出语句错误原因**

在选定代码行后，按快捷键**F9**，或者鼠标右键单击弹出菜单“断点”|“插入断点”，或者在该行行首空白处单击左键，或者选中菜单“调试”|“切换断点”



# Visual Studio 2015 使用方法

## (2) 启动调试

- ◆ 设置断点后要启动调试, 按快捷键**F5**, 或者菜单“**调试**”|“**开始调试**”, 或者工具栏按钮 

启动调试后, 程序运到断点所在的语句将暂停执行, 此时可以进行单步跟踪调试, 在变量观察窗口查看每条语句运行的结果是否和预期值相符, 从而定位出产生错误的具体语句。

变量观察窗口有 **自动窗口(Autos)**, **局部变量窗口 (Locals)**, **监视窗口(Watch 1-4)**

- ◆ 逐过程单步跟踪**F10** (Step Over), 单步调试时**不进入**被调用的函数内部跟踪,
- ◆ 逐语句单步跟踪**F11** (Step Into), 单步调试时将**进入**被调用值函数内部跟踪
- ◆ 跳出当前函数**Shite+F11**, 从当前跟踪的函数体快速跳出, 返回上层调用语句
- ◆ 快速运行到下一个断点处, 快捷键**F5**
- ◆ 停止调试, 快捷键**Shift+F5**





# Visual Studio 2015 使用方法

## (2) 启动调试

Visual Studio 2015 启动调试的界面截图。图中展示了代码编辑区、调试菜单、自动窗口、局部变量窗口、监视窗口和内存窗口。

**代码编辑区 (DebugSample.cpp):**

```
1  // 算术运算示例程序
2  #include <iostream>
3  using namespace std;
4
5  double x = 0.0;
6  double y = 0.0;
7  double result = 0.0;
8  char oper = '+';
9
10 cout << "算术运算示例程序" << endl;
11
12 Calculator c;
13 while (oper != 'q') {
14     cout << "请输入运算符，格式如：a + b | a - b | a * b | a / b\n";
15     cin >> x >> oper >> y;
16     if (oper == '/' && y == 0) {
17         cout << "Division by 0 exception" << endl;
18         continue;
19     }
20     result = c.calculate(x, y, oper);
21     cout << "结果: " << result << endl;
22 }
```

**调试菜单 (Debug):**

- 开始调试(S) F5
- 开始执行(不调试)(H) Ctrl+F5
- 性能探查器(E...) Alt+F2
- 附加到进程(P...) Ctrl+Alt+P
- 逐语句(S) F11
- 逐过程(O) F10
- 切换断点(S) F9
- 新建断点(B)
- 删除所有断点(D) Ctrl+Shift+F9
- 禁用所有断点(N)
- IntelliTrace(I)

**自动窗口 (Auto Window):**

名称	值	类型
返回 Calculator::calculate	4.0000000000000000	double
c	{...}	Calculator
oper	0x2b '+'	char
result	4.0000000000000000	double

**局部变量 (Local Variables):**

名称	值	类型
oper	0x2b '+'	char
result	4.0000000000000000	double
x	1.0000000000000000	double
y	3.0000000000000000	double

**监视 1 (Watch 1):**

名称	值	类型
oper	0x2b '+'	char
&oper	0x002dfcd7 '+'	char *

**内存 3 (Memory 3):**

地址	内容
0x002dfcd7	cc cc cc cc cc cc cc cc 00
0x002dfce1	00 00 00 00 10 40 cc cc cc
0x002dfceb	cc cc cc cc cc 00 00 00 00
0x002dfcf5	00 08 40 cc cc cc cc cc cc
0x002dfcfe	cc 00 00 00 00 00 f0 3f cc
0x002dfdf0	cc cc cc fd dc 33 98 24 fd 2d

**底部状态栏:**

行 36 列 3 字符 3 Ins 发布

## 第二部分

# Eclipse IDE for C++



# Eclipse IDE for C++

## ◆ 简介

Eclipse是基于Java的跨平台软件集成开发平台，支持多种编程语言，拥有非常丰富的扩展功能插件包。Eclipse IDE for C++提供了功能完整的C/C++集成开发环境，能够在Windows、Linux及Mac OS多种操作系统中完成C/C++软件的开发

- Eclipse 平台的安装和运行，需要首先安装Java 8以上的JRE或JDK，  
下载地址：[https://www.java.com/zh\\_CN/download/manual.jsp](https://www.java.com/zh_CN/download/manual.jsp)  
<https://www.oracle.com/technetwork/cn/java/javase/downloads/jre8-downloads-2133155-zhs.html>
- 还需要MinGW的GCC编译器,并将GCC的安装目录设置到系统环境变量的PATH参数中  
下载地址：<https://mingw-w64.org/doku.php>



# Eclipse IDE for C++

## ◆ 安装

Eclipse IDE for C++ 2018版在Windows下的安装有两种方式:

- 一种是下载离线压缩包, 解压到磁盘中适当目录就可以直接使用  
<https://www.eclipse.org/downloads/packages/release/2018-12/r/eclipse-ide-cc-developers>
- 另一种是下载Eclipse平台在线安装包, 执行安装管理器选择所需的软件包安装  
<https://www.eclipse.org/downloads/>  
在线安装说明 <https://www.eclipse.org/downloads/packages/installer>

The screenshot displays the Eclipse Foundation website. On the left, the 'Eclipse IDE for C/C++ Developers' page is shown, with the 'Download Links' section circled in red, listing Windows 64-bit, Mac OS X (Cocoa) 64-bit, and Linux 64-bit. Below this, the 'Downloaded 104,340 Times' and 'Checksums...' links are visible. On the right, the 'Download Eclipse Technology that is right for you' page is shown, featuring a 'Download 64 bit' button circled in red. The bottom of the page includes links for 'Get Eclipse IDE 2018-12', 'Eclipse Che', and 'Orion'.



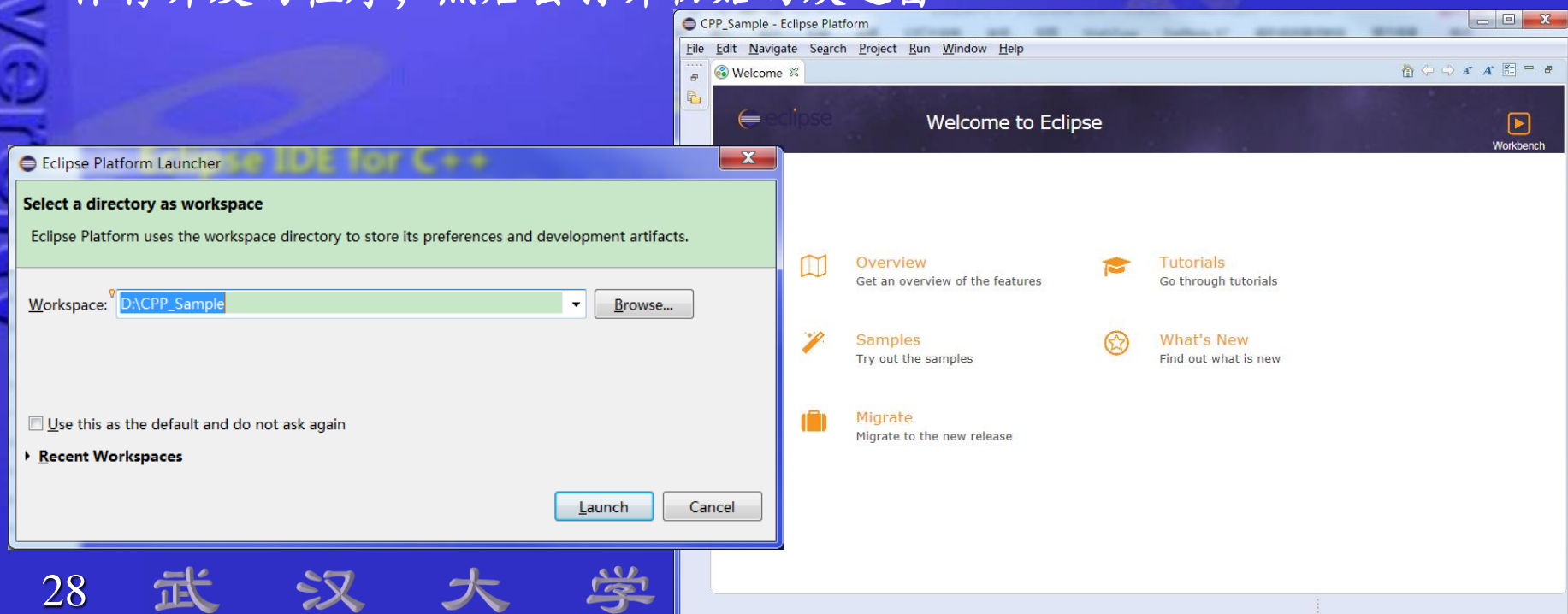
# Eclipse IDE for C++

## ◆ 1. 启动

离线解压方式安装的 Eclipse IDE for C++，找到所在的目录直接点击 Eclipse.exe 即可启动。

安装管理器在线方式 安装的2 Eclipse IDE for C++，通过创建的快捷图标可以启动

第一次启动Eclipse，需要指定工作空间（Workspace）的位置，用于保存开发的程序，然后会打开初始的欢迎窗口



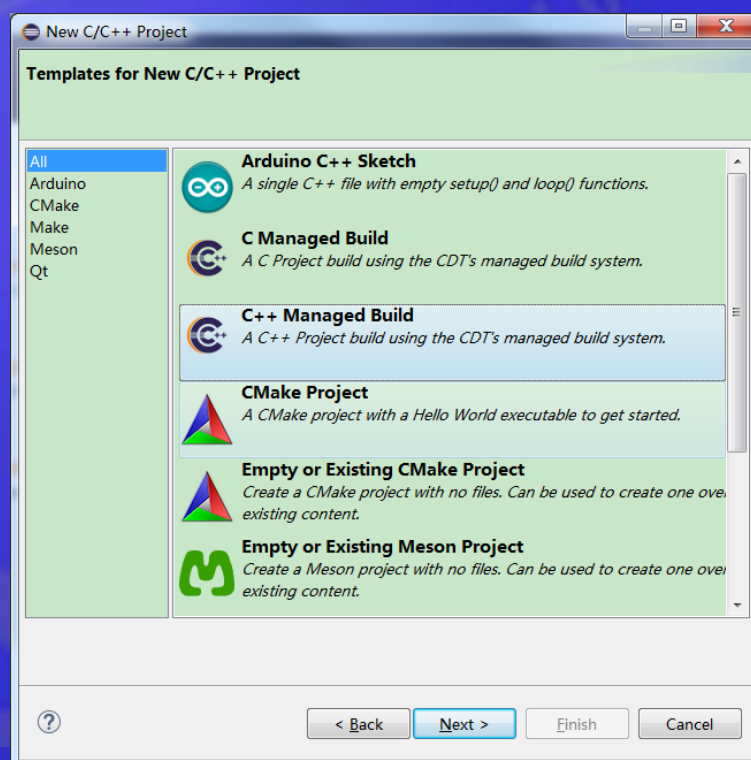
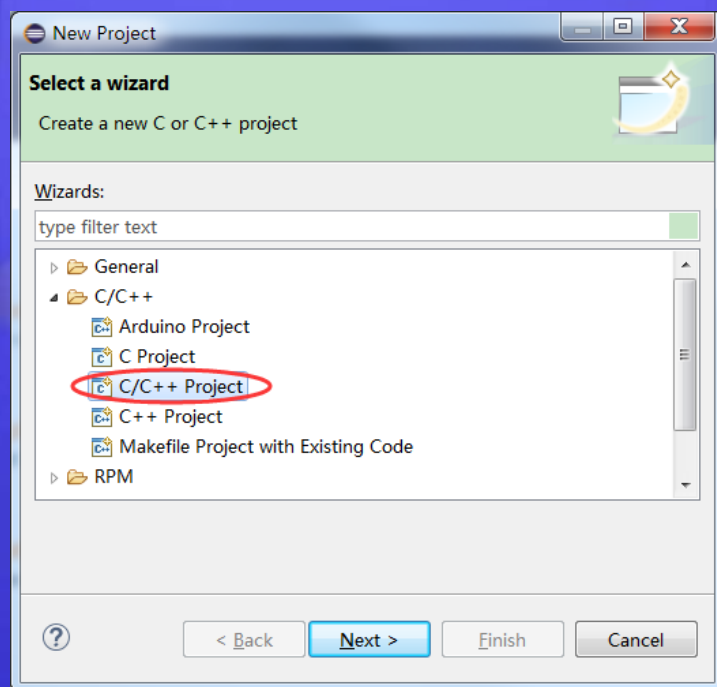


# Eclipse IDE for C++

## 2. 创建新项目

首先从菜单“File|New|Project...”可以打开新建项目向导，选择工程类型树中的“C/C++ Project”。

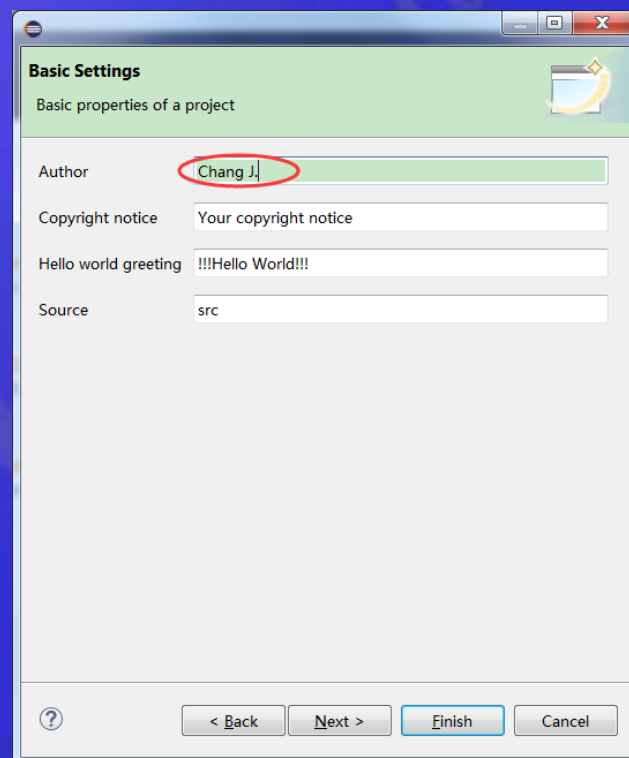
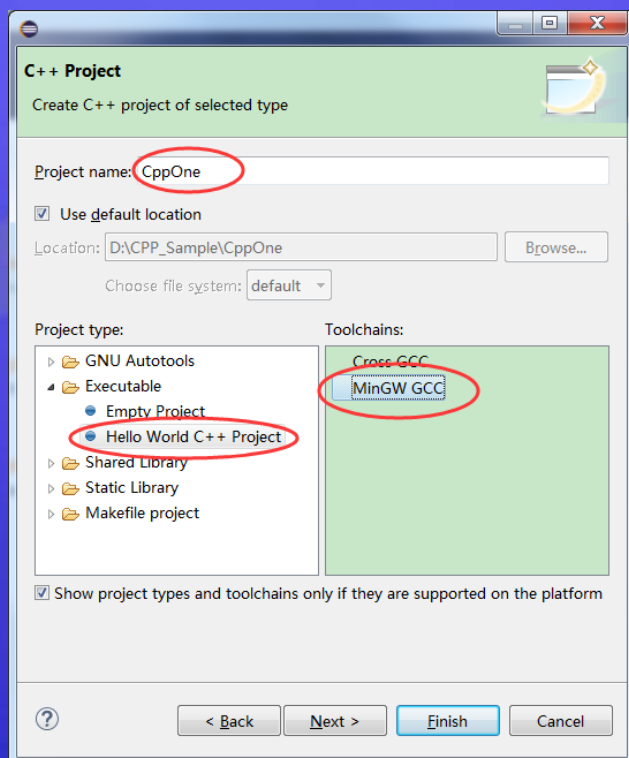
然后点击“Next”按钮，进入工程模板窗口，选择“C++ Managed Build”模板，点击“Next”



# Eclipse IDE for C++

## 2. 创建新项目

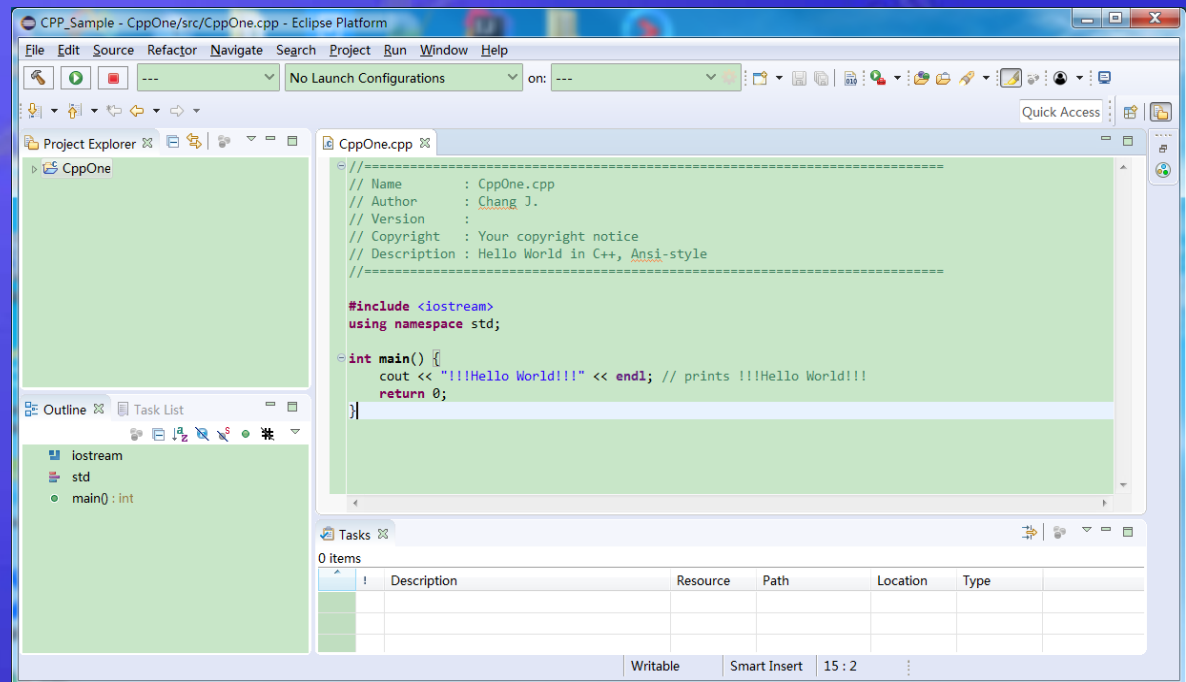
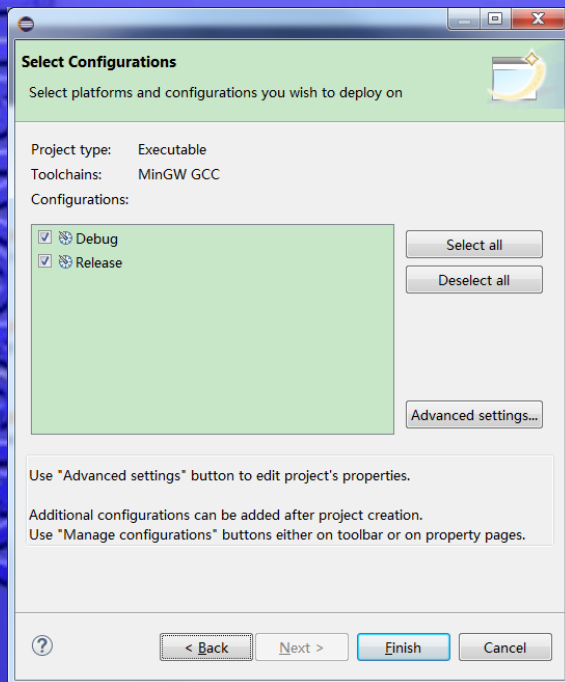
在“C++工程”窗口，输入工程名称（Project Name），选择工程的类型为“Hello World C++ Project”，选择工具链使用“MinGW GCC”；  
按“Next”进入项目信息设置窗口，输入项目基本设置信息



# Eclipse IDE for C++

## 2. 创建新项目

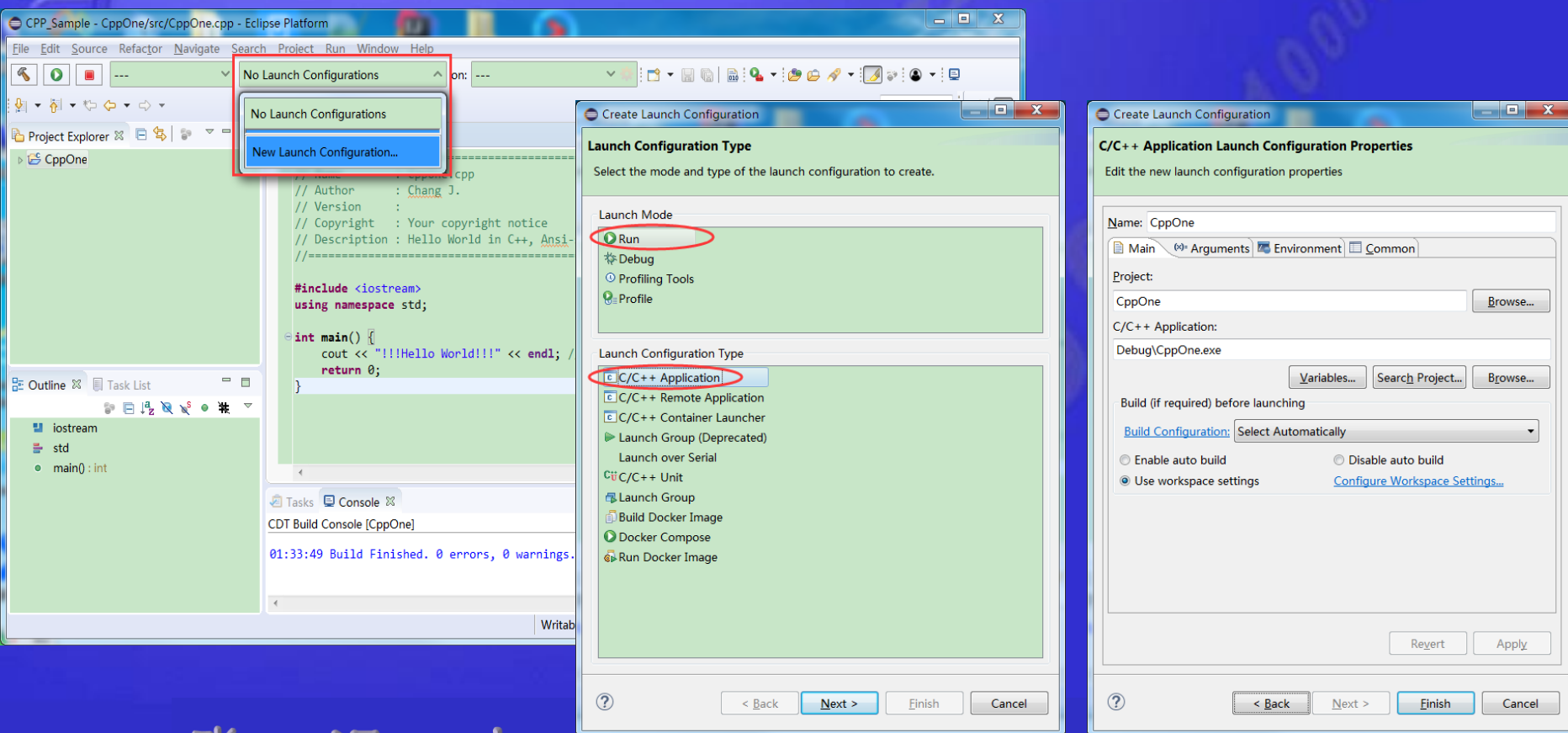
在“选择配置”窗口，保持缺省状态即可，点击“Finish”完成创建，生成了一个能够显示“Hello world”的简单程序



# Eclipse IDE for C++

## 3. 生成和运行项目

首先从菜单项“Project” | “Build All”或快捷键Ctrl+B 编译生成执行程序。然后设置创建程序启动配置项（New Launch Configuration），准备运行程序，设置完“应用启动配置属性”后，点击“Finish”



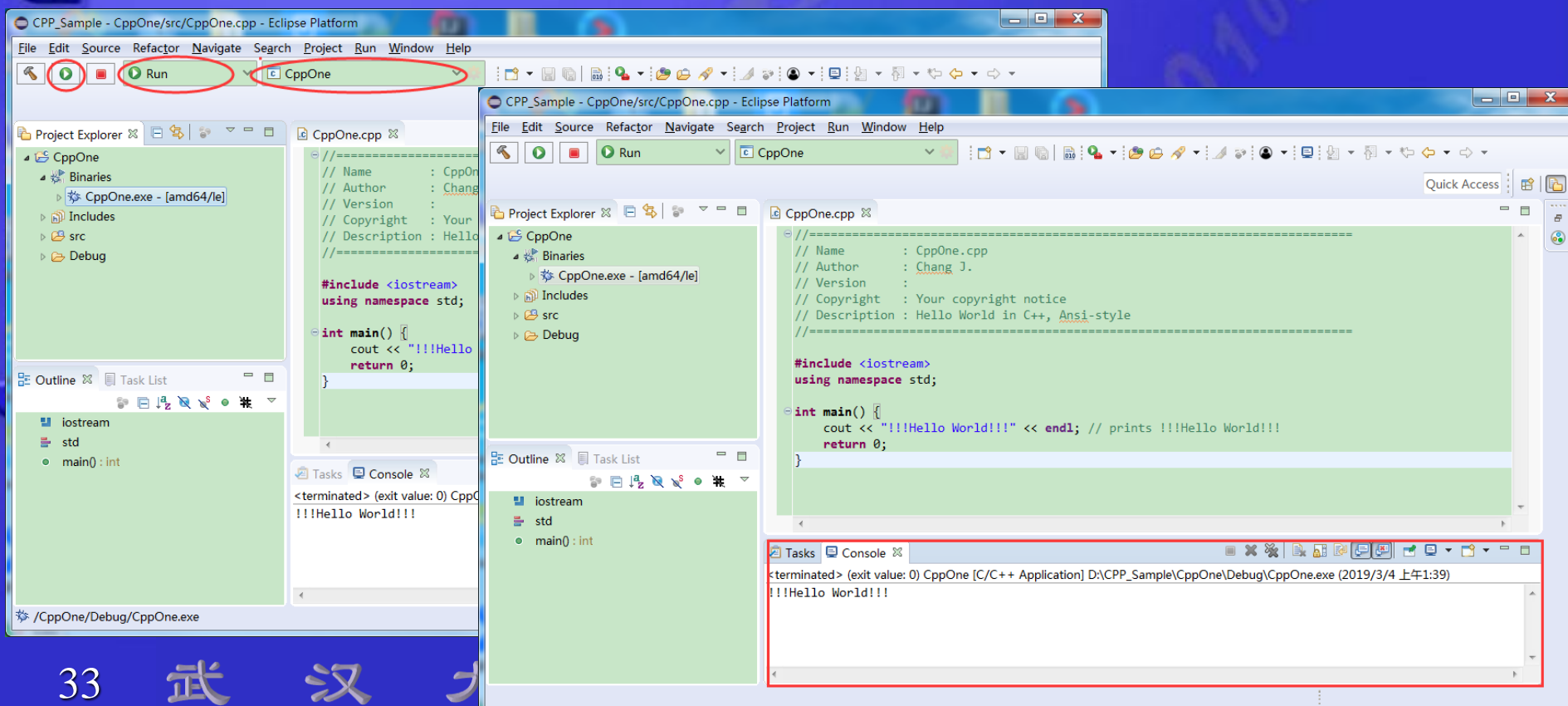


# Eclipse IDE for C++

## 3. 生成和运行项目

选择一个启动配置项(图中CppOne), 点击工具条“Run”图标, 运行程序。执行结果将显示在控制台区域中

可以另外配置Debug的启动配置项, 用Debug方式启动程序进行错误调试。





## 第三部分

## 关于DEV C++

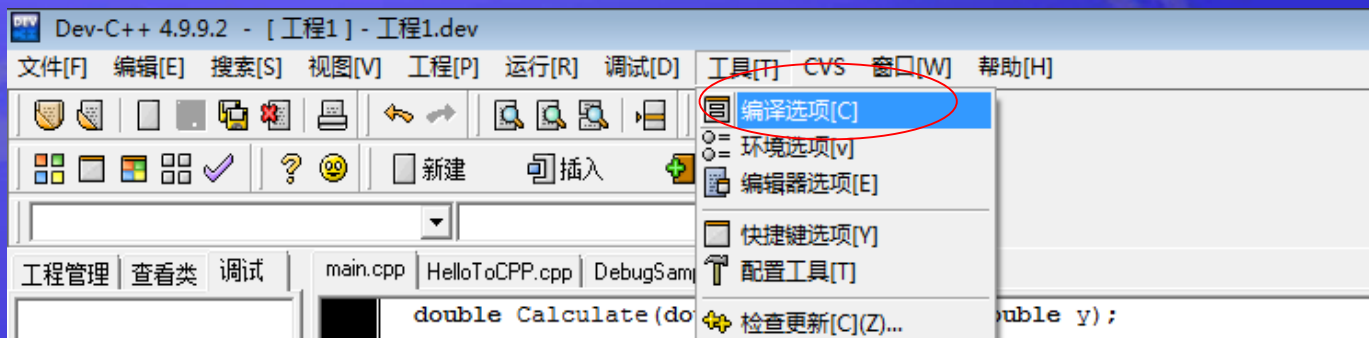
<https://sourceforge.net/projects/orwelldvcpp/>



# DEV C++ 调试功能不能使用怎么办

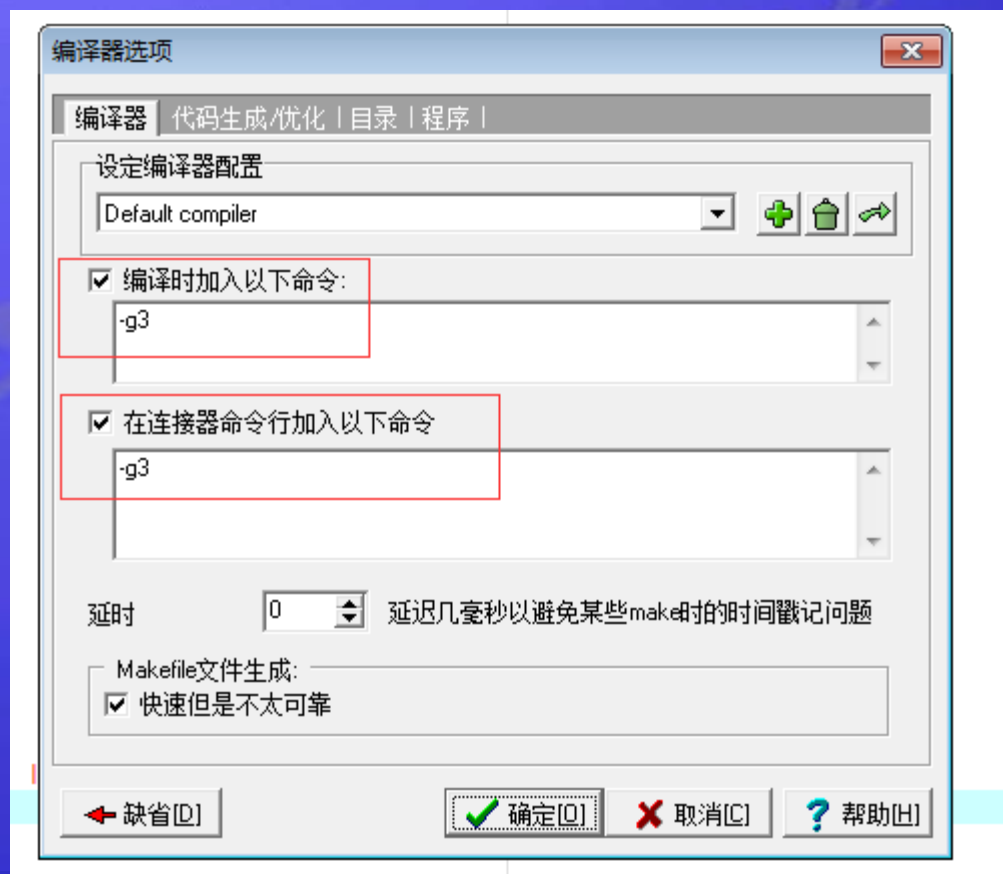
DEV C++：调试功能相对较弱，如果用Dev C++提示“项目没有调试信息，您想打开项目调试选项并重新生成吗？”，点击yes后退出没反应，请修改“编译选项”如下所示：

## 1、点击“工具”-》“编译选项”



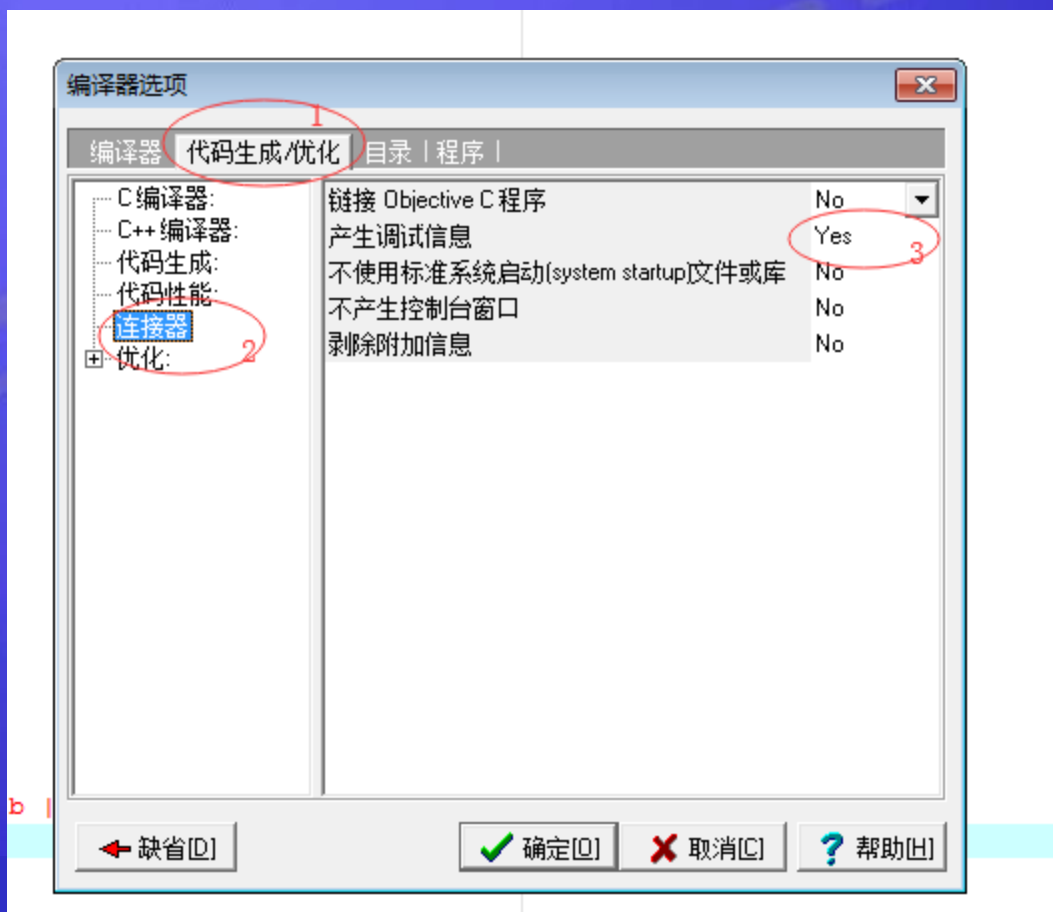
# DEV C++ 调试功能不能使用怎么办

2、点击“工具”-》“编译选项”，选择“编译器”，按下图配置



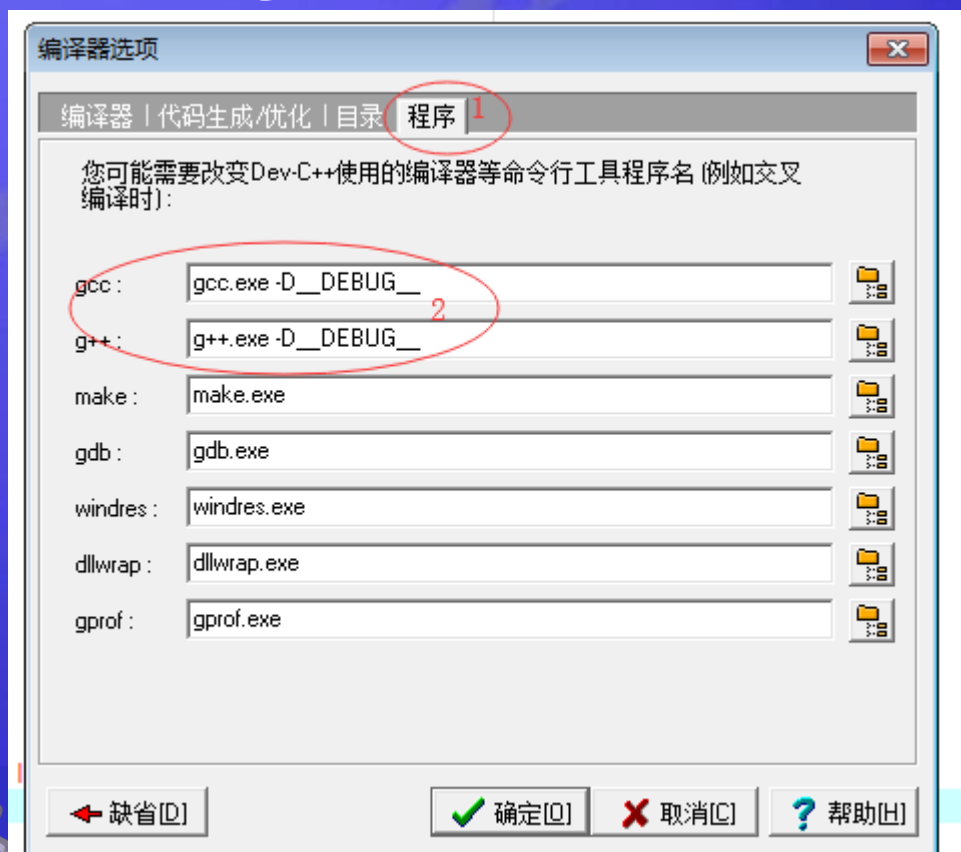
# DEV C++ 调试功能不能使用怎么办

3、点击“工具”-》“编译选项”，选择“编译器”，按下图配置



# DEV C++ 调试功能不能使用怎么办

- 4、点击“工具”-》“编译选项”，选择“编译器”，点击“程序”页，把gcc行修改为：gcc.exe -D\_\_DEBUG\_\_  
把g++行修改为g++.exe -D\_\_DEBUG\_\_





# 上机练习

学生用书：实验1

1. Visual Studio开发环境熟悉，练习本讲中的两个范例程序
2. 创建新工程，实现“Hello world”显示输出
3. 练习使用Debug排除程序错误，掌握断点、单步跟踪、观察变量值等基本方法



# 本讲结束



## 有问题吗?

