



C++ 环境设置

本地环境设置

如果您想要设置 C++ 语言环境，您需要确保电脑上有以下两款可用的软件，文本编辑器和 C++ 编译器。

文本编辑器

这将用于输入您的程序。文本编辑器包括 Windows Notepad、OS Edit command、Bri ef、Epsilon、EMACS 和 vim/vi。

文本编辑器的名称和版本在不同的操作系统上可能会有所不同。例如，Notepad 通常用于 Windows 操作系统上，vim/vi 可用于 Windows 和 Linux/UNIX 操作系统上。

通过编辑器创建的文件通常称为源文件，源文件包含程序源代码。C++ 程序的源文件通常使用扩展名 .cpp、.cp 或 .c。

在开始编程之前，请确保您有一个文本编辑器，且有足够的经验来编写一个计算机程序，然后把它保存在一个文件中，编译并执行它。

C++ 编译器

写在源文件中的源代码是人类可读的源。它需要"编译"，转为机器语言，这样 CPU 可以按给定指令执行程序。

C++ 编译器用于把源代码编译成最终的可执行程序。

大多数的 C++ 编译器并不在乎源文件的扩展名，但是如果您未指定扩展名，则默认使用 .cpp。

最常用的免费可用的编译器是 GNU 的 C/C++ 编译器，如果您使用的是 HP 或 Solaris，则可以使用各自操作系统上的编译器。

以下部分将指导您如何在不同的操作系统上安装 GNU 的 C/C++ 编译器。这里同时提到 C/C++，主要是因为 GNU 的 gcc 编译器适合于 C 和 C++ 编程语言。

安装 GNU 的 C/C++ 编译器

UNIX/Linux 上的安装

如果您使用的是 **Linux 或 UNIX**，请在命令行使用下面的命令来检查您的系统上是否安装了 GCC：

```
$ g++ -v
```

如果您的计算机上已经安装了 GNU 编译器，则会显示如下消息：

分类导航

HTML / CSS

JavaScript

服务端

数据库

移动端

XML 教程

ASP.NET

Web Service

开发工具

网站建设

Advertisement



C++ 接口 (抽象类)

C++ 高级教程

C++ 文件和流

C++ 异常处理

C++ 动态内存

C++ 命名空间

C++ 模板

C++ 预处理器

C++ 信号处理

C++ 多线程

C++ Web 编程

C++ 资源库

C++ STL 教程

C++ 标准库

C++ 有用的资源

C++ 实例

Using built-in specs.

Target: i386-redhat-linux

Configured with: ../configure --prefix=/usr

Thread model: posix

gcc version 4.1.2 20080704 (Red Hat 4.1.2-46)

如果未安装 GCC, 那么请按照 <http://gcc.gnu.org/install/> 上的详细说明安装 GCC。

Mac OS X 上的安装

如果您使用的是 Mac OS X, 最快捷的获取 GCC 的方法是从苹果的网站下载 Xcode 开发环境, 并按照安装说明进行安装。一旦安装上 Xcode, 您就能使用 GNU 编译器。

Xcode 目前可从 developer.apple.com/technologies/tools/ 上下载。

Windows 上的安装

为了在 Windows 上安装 GCC, 您需要安装 MinGW。为了安装 MinGW, 请访问 MinGW 的主页 www.mingw.org, 进入 MinGW 下载页面, 下载最新版本的 MinGW 安装程序, 命名格式为 MinGW-<version>.exe。

当安装 MinGW 时, 您至少要安装 gcc-core、gcc-g++、binutils 和 MinGW runtime, 但是一般情况下都会安装更多其他的项。

添加您安装的 MinGW 的 bin 子目录到您的 **PATH** 环境变量中, 这样您就可以在命令行中通过简单的名称来指定这些工具。

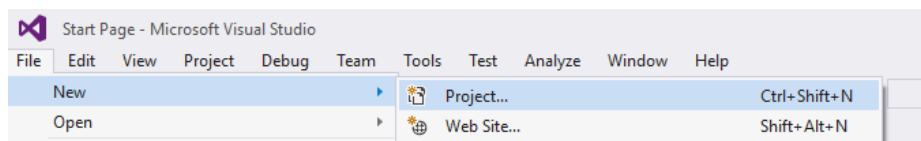
当完成安装时, 您可以从 Windows 命令行上运行 gcc、g++、ar、ranlib、dlltool 和其他一些 GNU 工具。

使用 Visual Studio (Graphical Interface) 编译

1、下载及安装 [Visual Studio Community 2015](#)。

2、打开 Visual Studio Community

3、点击 File -> New -> Project



4、左侧列表选择 Templates -> Visual C++ -> Win32 Console Application, 并设置项目名称为 MyFirstProgram。

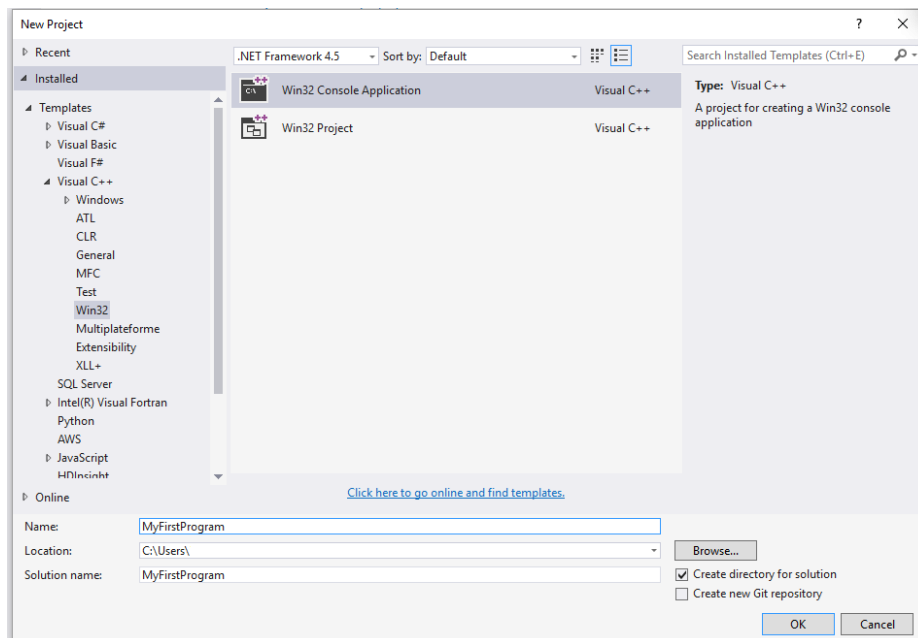
亿速

10大游戏公司都在用

29

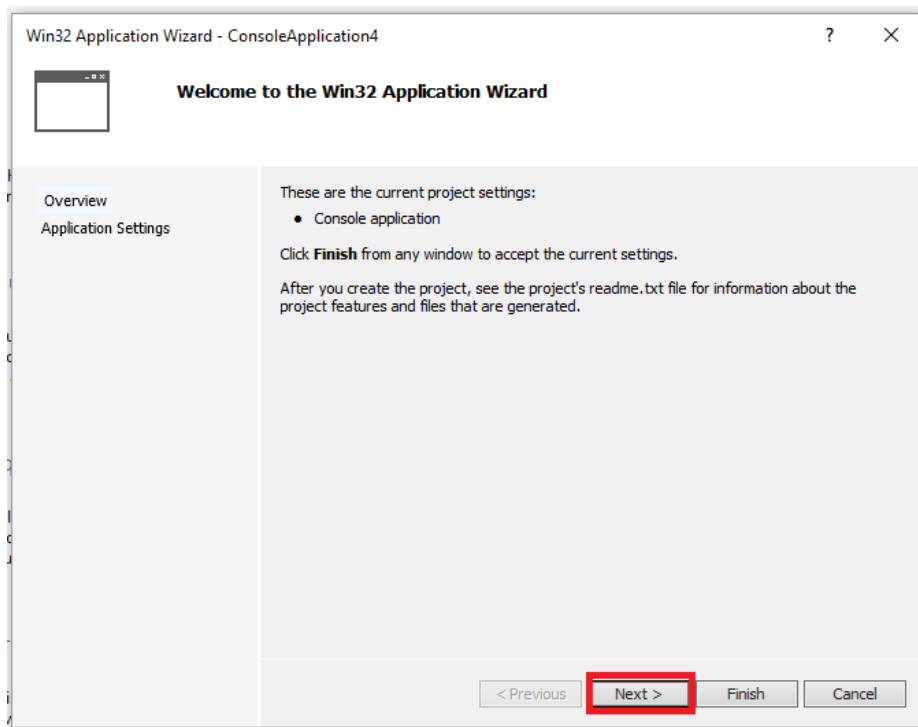


反馈/建议



5、点击 OK。

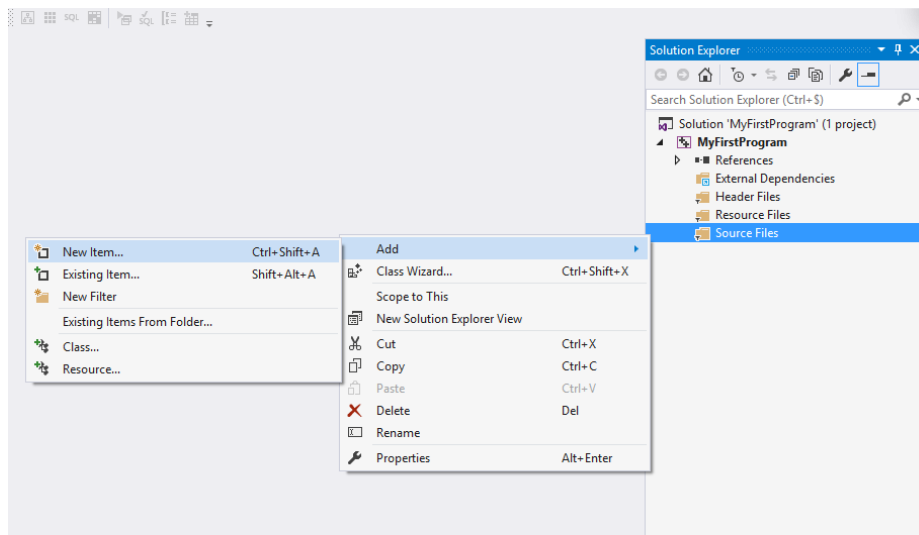
6、在以下窗口中点击 Next



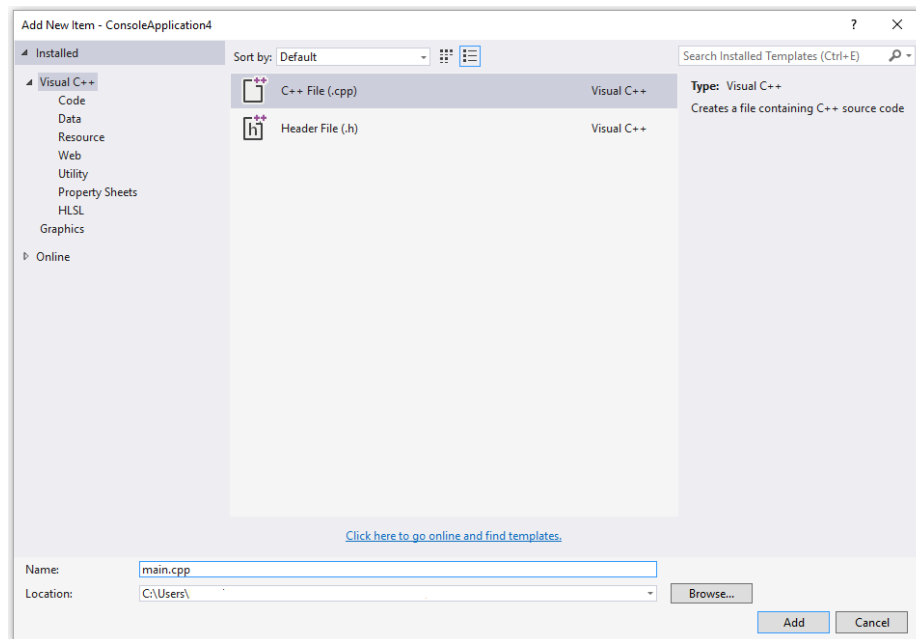
7、在弹出的窗口中选择 Empty project 选项后，点击 Finish 按钮：

8、右击文件夹 Source File 并点击 Add --> New Item...：





9、选择 C++ File 然后设置文件名为 main.cpp，然后点击 Add：

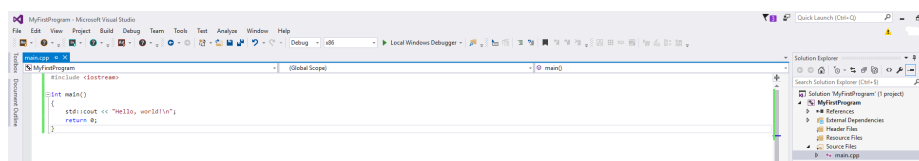


10、拷贝以下代码到 main.cpp 中：

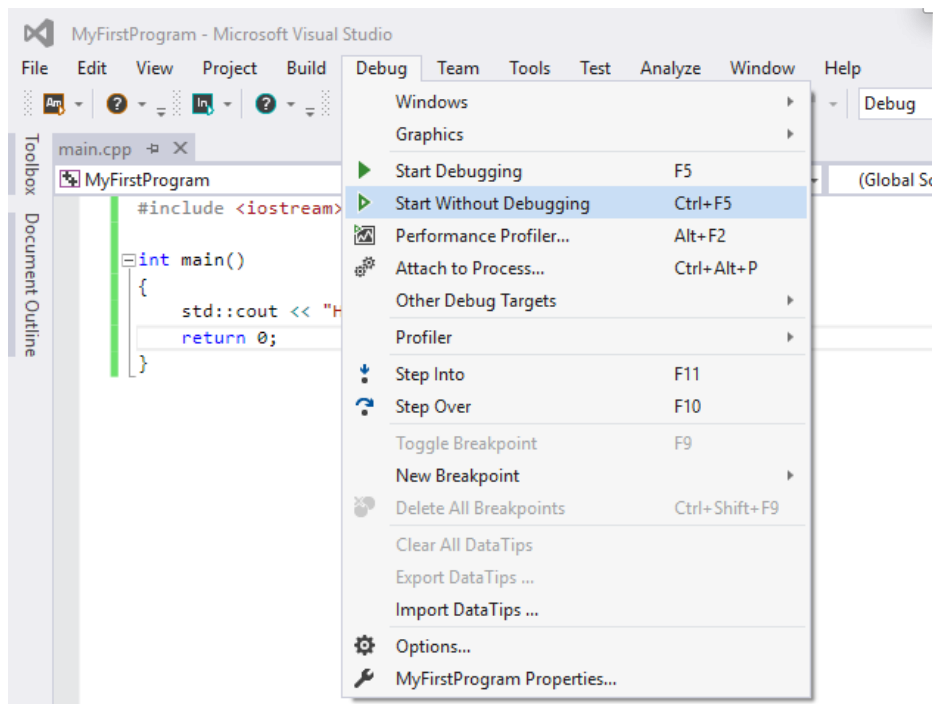
```
#include <iostream>

int main()
{
    std::cout << "Hello World!\n";
    return 0;
}
```

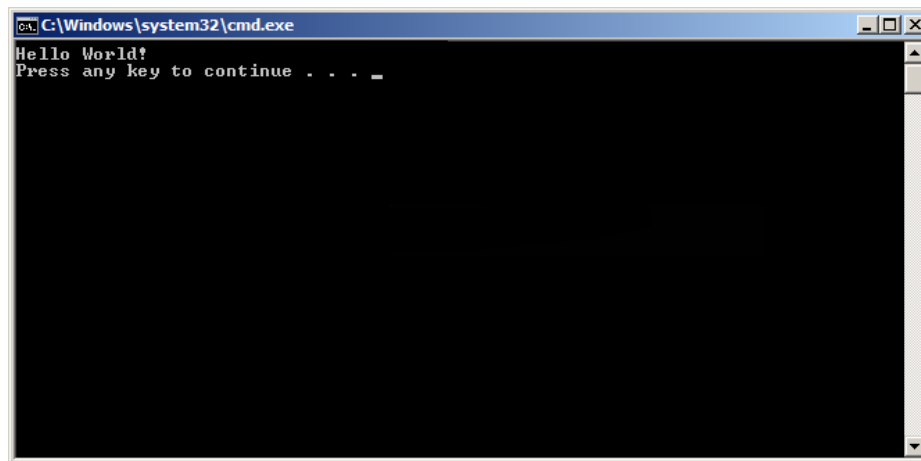
界面如下所示：



11、点击菜单上的 Debug -> Start Without Debugging (或按下 ctrl + F5)：



12、完成以上操作后，你可以看到以下输出：



g++ 应用说明

程序 g++ 是将 gcc 默认语言设为 C++ 的一个特殊的版本，链接时它自动使用 C++ 标准库而不用 C 标准库。通过遵循源码的命名规范并指定对应库的名字，用 gcc 来编译链接 C++ 程序是可行的，如下例所示：

```
$ gcc main.cpp -lstdc++ -o main
```

下面是一个保存在文件 helloworld.cpp 中一个简单的 C++ 程序的代码：

```
#include <iostream>
using namespace std;
int main()
{
    cout << "Hello, world!" << endl;
    return 0;
}
```

最简单的编译方式：

反馈/建议

```
$ g++ helloworld.cpp
```

由于命令行中未指定可执行程序的文件名，编译器采用默认的 a.out。程序可以这样来运行：

```
$ ./a.out
Hello, world!
```

通常我们使用 -o 选项指定可执行程序的文件名，以下实例生成一个 helloworld 的可执行文件：

```
$ g++ helloworld.cpp -o helloworld
```

执行 helloworld:

```
$ ./helloworld
Hello, world!
```

如果是多个 C++ 代码文件，如 runoob1.cpp、runoob2.cpp，编译命令如下：

```
$ g++ runoob1.cpp runoob2.cpp -o runoob
```

生成一个 runoob 可执行文件。

g++ 有些系统默认是使用 C++98，我们可以指定使用 C++11 来编译 main.cpp 文件：

```
g++ -g -Wall -std=c++11 main.cpp
```

g++ 常用命令选项

选项	解释
-ansi	只支持 ANSI 标准的 C 语法。这一选项将禁止 GNU C 的某些特色，例如 asm 或 typeof 关键词。
-c	只编译并生成目标文件。
-DMACRO	以字符串"1"定义 MACRO 宏。
-DMACRO=DEFN	以字符串"DEFN"定义 MACRO 宏。
-E	只运行 C 预编译器。
-g	生成调试信息。GNU 调试器可利用该信息。
-IDIRECTORY	指定额外的头文件搜索路径 DIRECTORY。
-LDIRECTORY	指定额外的函数库搜索路径 DIRECTORY。

反馈/建议

-lLIBRARY	连接时搜索指定的函数库LIBRARY。
-m486	针对 486 进行代码优化。
-o	FILE 生成指定的输出文件。用在生成可执行文件时。
-O0	不进行优化处理。
-O	或 -O1 优化生成代码。
-O2	进一步优化。
-O3	比 -O2 更进一步优化，包括 inline 函数。
-shared	生成共享目标文件。通常用在建立共享库时。
-static	禁止使用共享连接。
-UMACRO	取消对 MACRO 宏的定义。
-w	不生成任何警告信息。
-Wall	生成所有警告信息。

← C++ 简介

C++ 基本语法 →

 点我分享笔记

在线实例

- HTML 实例
- CSS 实例
- JavaScript 实例
- Ajax 实例
- jQuery 实例
- XML 实例
- Java 实例

字符集&工具

- HTML 字符集设置
- HTML ASCII 字符集
- HTML ISO-8859-1
- HTML 实体符号
- HTML 拾色器
- JSON 格式化工具

最新更新

- Python redis 使...
- Windows10 MYSQ...
- Docker 镜像加速
- Debian Docker 安装
- C 库函数
- Linux groupadd ...
- CSS var() 函数

站点信息

- 意见反馈
- 合作联系
- 免责声明
- 关于我们
- 文章归档

关注微信



Copyright © 2013-2019 菜鸟教程
runoob.com All Rights Reserved
备案号：闽ICP备15012804号

反馈/建议



反馈/建议