

软件工程基础及实践

**课程实践作业一**

**院 系：**  能源与环境学院

**专 业：** 能源动力工程

**姓 名：** 石婧

**学 号：** 03016405

**指导教师：** 程懋华

2019年 3月 13日

目 录

[一、GitHub简介与账户建立 1](#_Toc3464816)

[1.1 GitHub的简介 1](#_Toc3464817)

[1.2 GitHub的账户建立 2](#_Toc3464818)

[二、Python语言 3](#_Toc3464819)

[2.1 Python3.7.2的安装 3](#_Toc3464820)

[2.2 扩展包的安装 6](#_Toc3464821)

[2.2.1 更新pip至最新版本 6](#_Toc3464822)

[2.2.2 设置下载默认网址 6](#_Toc3464823)

[2.2.3安装规范格式扩展包 6](#_Toc3464824)

[2.2.4 安装科学计算包 7](#_Toc3464825)

[2.2.5 安装SEUIF97 7](#_Toc3464826)

[2.3 Python3.7.2程序示例 8](#_Toc3464827)

[三、Jupyter Notebook的安装与运行 10](#_Toc3464828)

[3.1 Jupyter Notebook的安装 10](#_Toc3464829)

[3.2 Jupyter Notebook的简单运行 10](#_Toc3464830)

[3.2.1 Jupyter Notebook的启动 10](#_Toc3464831)

[3.2.2 Jupyter Notebook的简单编程 11](#_Toc3464832)

[四、安装MinGW-W64 13](#_Toc3464833)

[4.1 下载MinGW-W64 13](#_Toc3464834)

[4.2设置MinGW-W64运行环境 13](#_Toc3464835)

[五、Visual Studio code的安装与使用 15](#_Toc3464836)

[5.1 Visual Studio code的安装 15](#_Toc3464837)

[5.2 Python，C/C++插件的安装 16](#_Toc3464838)

[5.3 VS Code简单编程 16](#_Toc3464839)

[5.3.1 简单的Python程序 16](#_Toc3464840)

[5.3.2 简单的C++程序 17](#_Toc3464841)

[六、遇到的问题及解决方法 18](#_Toc3464842)

[七、个人小结 21](#_Toc3464843)

[参考文献 22](#_Toc3464844)

# GitHub简介与账户建立

## GitHub的简介

GitHub是一个基于Web，利用Git作为版本格式控制的托管服务平台。它提供了Git的所有分布式版本控制和源代码管理（SCM）功能，以及添加了订阅、讨论组、错误跟踪、功能请求、任务管理、代码片段分享（Gist）等自己的功能。

GitHub一直为包括免费账户在内的所有账户提供无限的私人存储库。2018年6月，GitHub报告其拥有超过2800万用户和5700万个存储库（包括2800万个公共存储库）。GitHub上还拥有众多知名开源项目，如Google、Facebook、python、Apple等。开源项目会在GitHub上发布自己的代码并与其他开发者协作编码。

JavaScript，机器学习，移动应用程序开发和容器化等都是最近几年GitHub开源软件中一些最强大的趋势。美国以外的开源贡献者的比例也每年都在增长，中亚，中东和非洲的资源库创建已经有所增加。在尼日利亚，一个不断增长的开发者社区在2018年创造了比2017年多1.7倍的开源存储库。

GitHub的优点有很多。它适合分布式开发，强调个体，公共的服务器压力和数量都不会太大，并且速度快， 拥有成熟的架构，开发灵活。利用GitHub，我们还可以将项目存档，与其他人分享交流，并让其他开发者帮助你一起完成这个项目。它支持多人共同完成一个项目，因此我们可以在同一页面进行对话交流。我们还能下载别人源码进行学习，也能找到好多好玩的软件。但GitHub的缺点就是比较复杂，学习难度较大。

## GitHub的账户建立

首先，进入GitHub官网<https://github.com/>

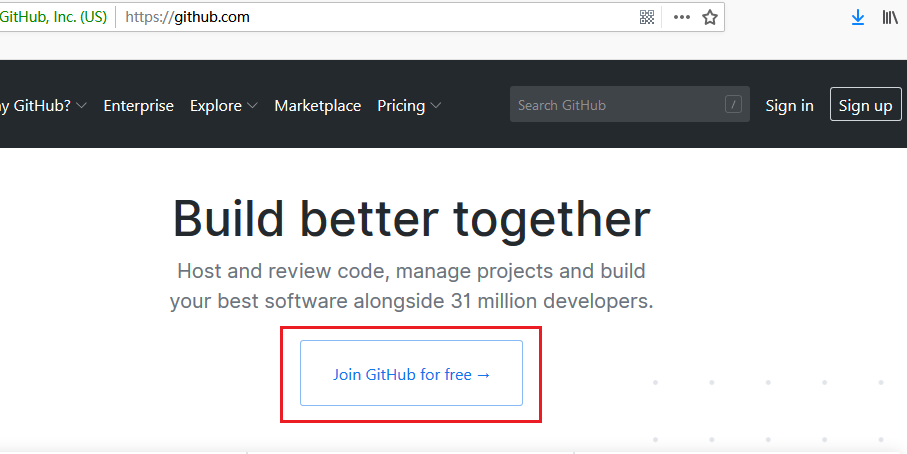


图1. GitHub官网

点击Join GitHub for free进行账户注册，填写好基本信息后进入个人的账户主页即可开始建立自己的仓库。

我所创建的账户名是：Shi-Jing1999 ，账户所使用电子邮箱是：shi101805@163.com

以下是我的Github账户主页截图：

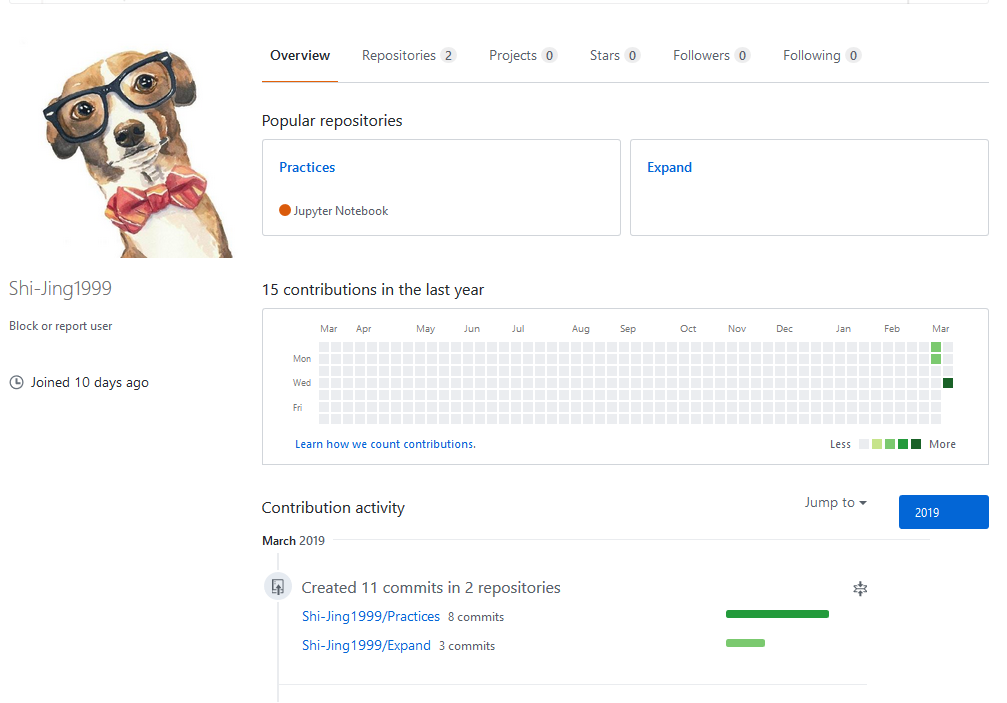


图2. Shi-Jing1999的账户主页

# Python语言

## 2.1 Python3.7.2的安装

首先进入Python的官网：<http://www.python.org> ，

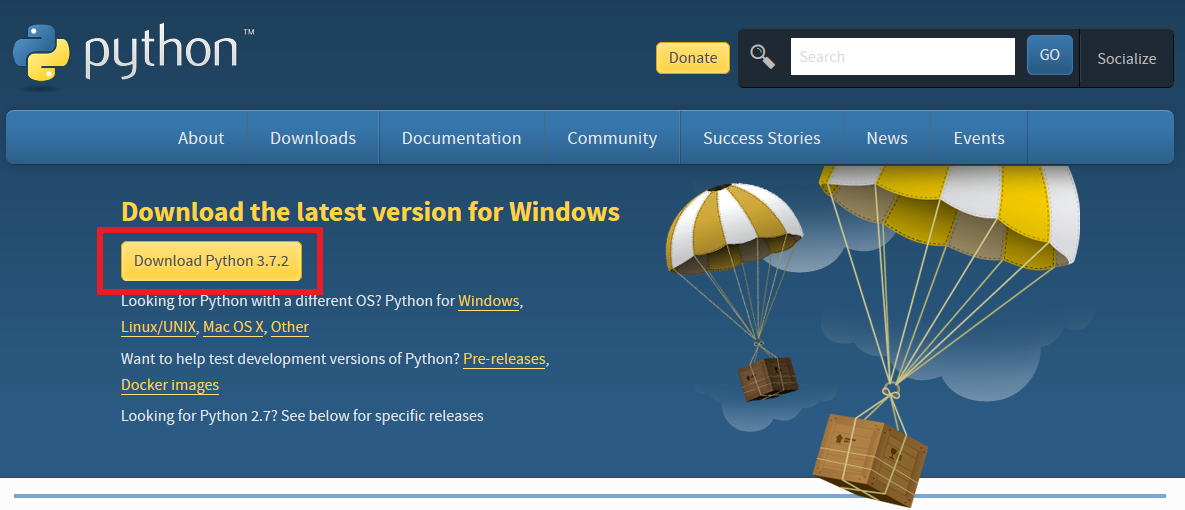


图3. Python官网部分界面

点击“Download Python3.7.2”，应注意不同的的操作系统有不同的软件版本，下载符合自己操作系统的软件版本。以我64位的Windows操作系统为例，我下载的版本是Windows x84-64 executable installer。我下载的软件版本已在下图中标注出。



图4. Python部分版本

软件安装包成功下载后，双击安装包，得到Python的安装界面，首先勾选Add Python 3.7 to PATH，然后点击Customize installation进入自定义安装界面。勾选Optional Features的所有选项后进入下一步。在Advanced Options界面额外勾选第一、五、六个选项，第七个选项是为了安装了VS 2015的用户准备用来调试程序的，我们没有安装所以不用理会。然后我们要注意自定义文件夹的安装位置，最好直接安装在C盘目录下，便于日后的软件使用。最后点击install进行软件安装。

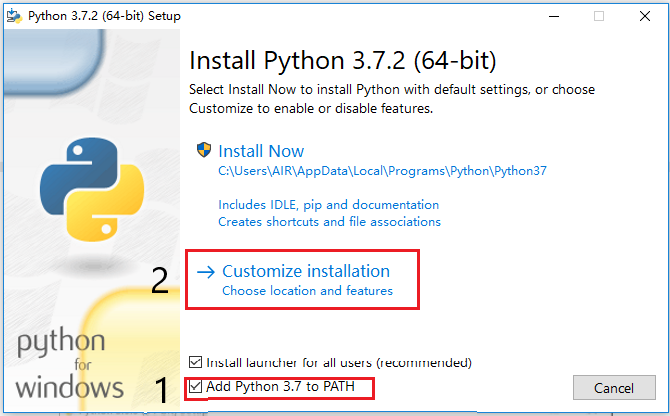


图5. Python安装界面

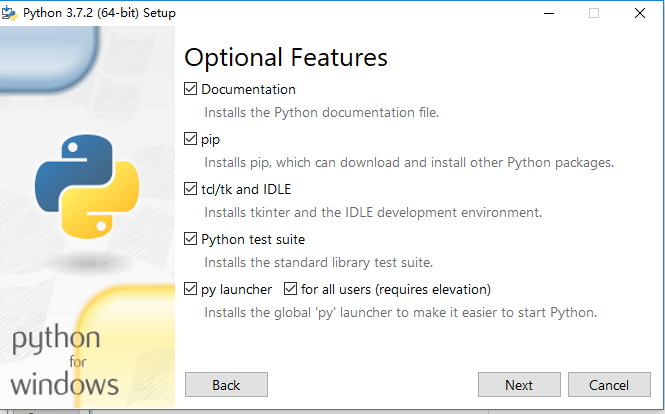


图6. Python安装自定义界面1

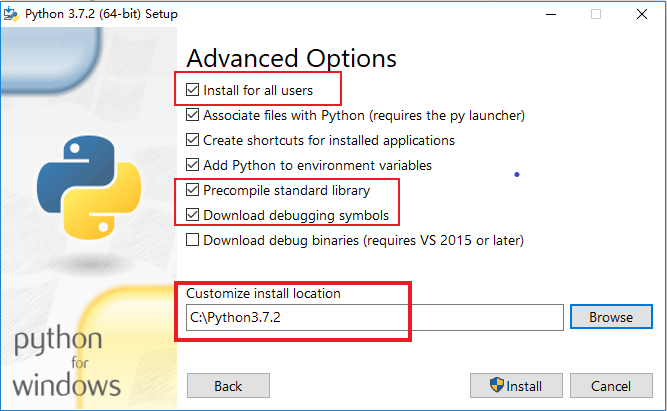


图7. Python安装自定义界面2

安装完成后的界面如下图：

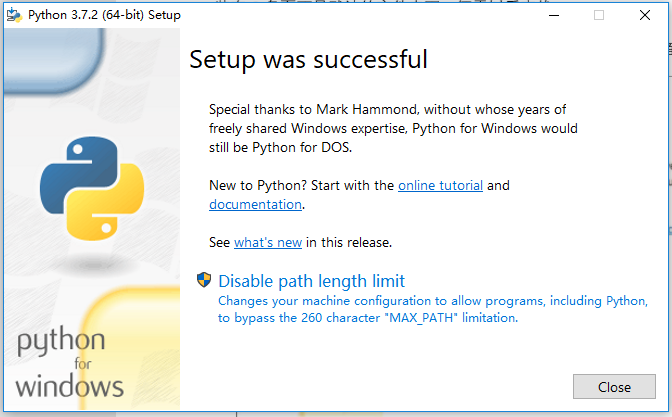


图8. Python成功安装界面

接着打开命令提示符窗口，可以输入C:\Users\AIR>python 指令查看Python的版本序列号。

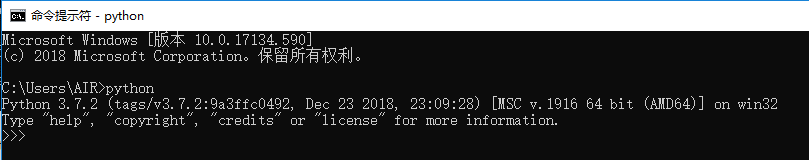


图9. 在cmd窗口查看Python版本

## 2.2 扩展包的安装

### 2.2.1 更新pip至最新版本

pip是在线安装Python软件包的软件，一般安装好Python会自带pip软件，但需要手动将其更新到最新版本。但是我的电脑不知为何没有安装好pip软件，因此，在命令提示符窗口输入 >python -m pip install -U pip 安装pip的最新版本。

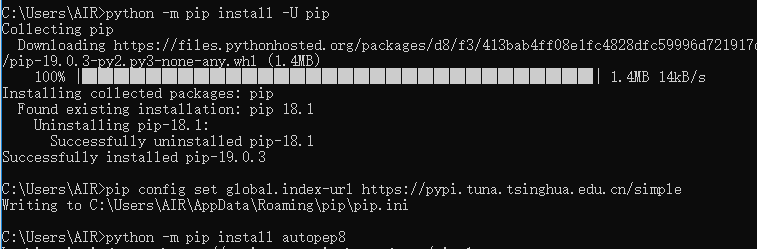


图10. 更新pip至最新版本

### 2.2.2 设置下载默认网址

由于在Python的官网上下载软件包的速度可能较慢，因此可以将清华大学的网址作为pip在线下载软件包的默认网址，加快下载速度。在命令提示符窗口输入 >pip config set global.index-url <https://pypi.tuna.tsinghua.edu.cn/simple> 进行设置。

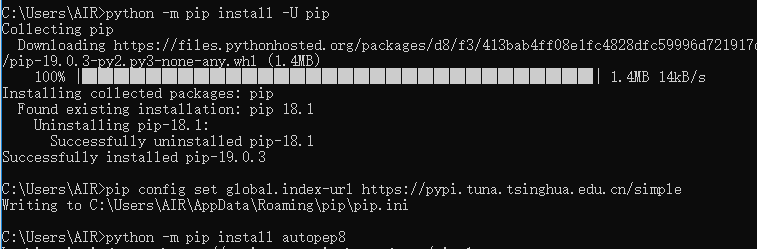


图11. 设置清华网址为下载默认网址

### 2.2.3安装规范格式扩展包

安装规范格式扩展包autopep8 和 pylint等利于编程环境的进一步建立，并且可直接利用pip进行。在命令提示符窗口分别输入 >python -m pip install autopep8 和 >python -m pip install pylint 进行安装。此处由于截图疏忽导致部分安装过程缺失。

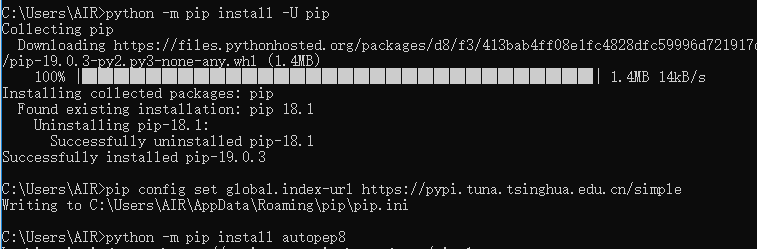


图12. 安装autopep8



图13. 安装pylint

### 2.2.4 安装科学计算包

安装Numpy, Scipy, Matplotlib需要先进入 <https://pypi.org/> 下载适合自己电脑版本与适合刚安装的Python版本的安装包，下面以Scipy为例。我下载的版本是scpy-1.2.1-cp37-cp37m-win\_amd64.whl，其中cp37表示适用于Python3.7即以上的软件，win\_amd64 表示适用于Windows的64位系统。

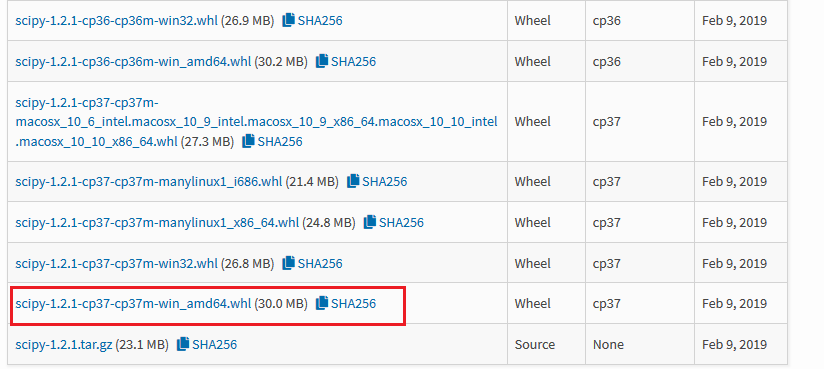


图14：Scipy的部分版本

软件包成功下载后，在命令提示符窗口输入pip install scpy-1.2.1-cp37-cp37m-win\_amd64.whl 进行安装。需要注意的是，安装的目录需要是下载的Scipy软件包存放的目录，否则将安装失败。

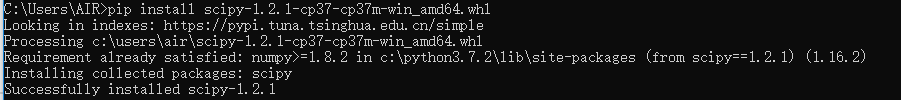


图15：安装Scipy软件包

### 2.2.5 安装SEUIF97

进入 <https://github.com/PySEE/SEUIF97> 下载仓库的 zip 文件并解压，将和操作系统对应版本的 libseuif97.dll拷贝到 c:\windows\system ，seuif97.py 拷贝到 c:\Python3.7.2\Lib 。

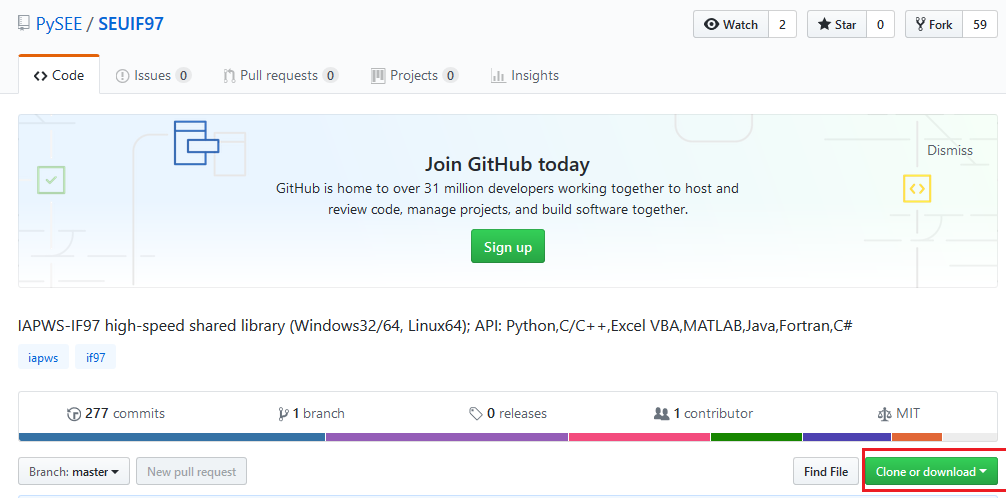


图16：下载仓库的zip文件

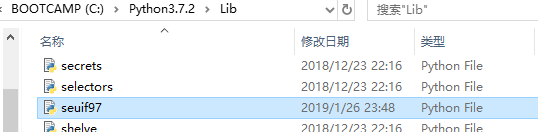


图17. seuif97拷贝到 C:\Python3.7.2\Lib

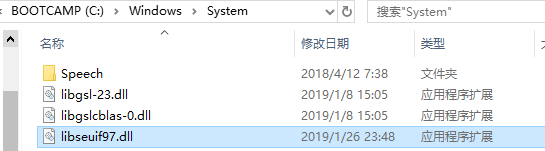


图18. Libseuif97.dll拷贝到C:\Windows\System

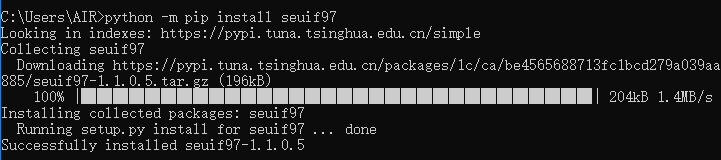
在命令提示符窗口输入> python -m pip install seuif97 进行安装。

图19.安装seuif97

## 2.3 Python3.7.2程序示例

我们可以直接在命令提示符窗口输入程序指令，按键 enter 执行每一句指令。首先需要在命令提示符窗口输入>python，然后就可以开始进行输入，但命令提示符窗口只能执行较简单的命令。以下为简单程序示例。

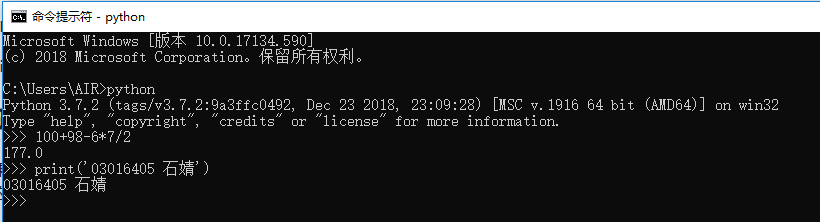
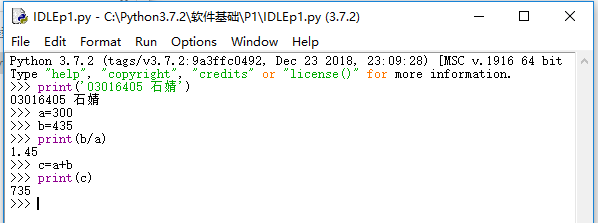


图20. 命令提示符窗口编写程序

我们还可以直接在IDLE窗口进行编程，若想要以文本形式给出编程的过程，则可将其保存为markdown格式的文档。



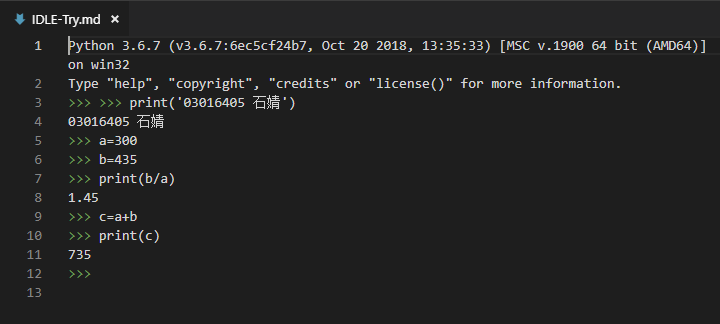
图21. IDLE窗口编程

图22. markdown格式的程序文档

# Jupyter Notebook的安装与运行

## 3.1 Jupyter Notebook的安装

Jupyter Notebook的本质是一个 Web 应用程序，便于创建和共享文学化程序文档，支持实时代码，数学方程，可视化和 markdown，其用途包括：数据清理和转换，数值模拟，统计建模，机器学习等等。

安装Jupyter Notebook时在命令提示符窗口输入 >python -m pip install jupyter

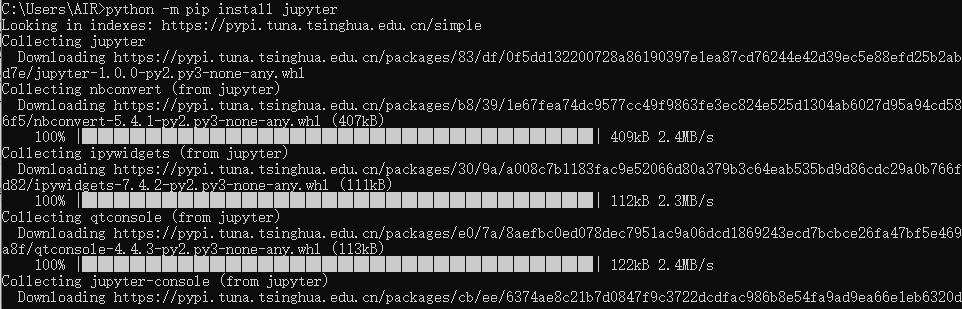


图23. 安装Jupyter Notebook

## 3.2 Jupyter Notebook的简单运行

### 3.2.1 Jupyter Notebook的启动

成功安装Jupyter Notebook后，直接在命令提示符窗口输入>jupyter notebook 即可打开Jupyter，此时打开的是默认的C:\Users\AIR下的Jupyter Notebook。



图24. C:\Users\AIR下的Jupyter Notebook

如果想在特定文件夹下打开Jupyter Notebook，可以先在特定文件夹下用记事本创建一个内容为“jupyter notebook”的名为start.bat的文件，然后另存为“所有文件”格式，这时双击得到的文件start.bat即可打开该文件所在目录下的Jupyter Notebook。

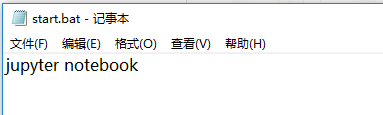


图25. 在记事本创建start.bat文件

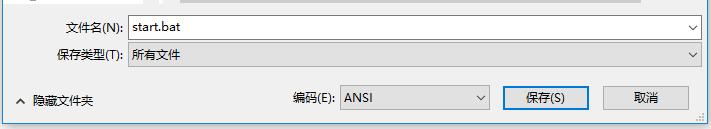


图26. 将start.bat另存为所有文件格式

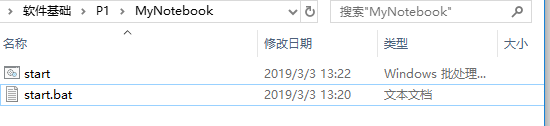


图27. 得到的start启动文件

### 3.2.2 Jupyter Notebook的简单编程

双击start.bat文件打开刚才在特定文件夹MyNotebook文件夹下的Jupyter Notebook，点击右上角的“new”，新建 Python3 格式的文件，其后缀名为.ipynb。



图28. 新建笔记本

打开后便可在每一个cell内编写程序或文本，可通过改变下图下拉框中的内容是“代码”或者“标记”，从而更改每一个cell里的内容。



图29. 更改cell中的文本类型

以下为我的Jupyter Notebook的简单编程：



图30. Jupyter Notebook编程示例

# 安装MinGW-W64

## 4.1 下载MinGW-W64

进入网址 <https://sourceforge.net/projects/mingw-w64/files/?source=navbar> 下载MinGW-W64安装包。



图31. 下载MinGW-W64安装包

## 4.2设置MinGW-W64运行环境

在C盘中创建新文件夹mingw64，将下载成功后解压压缩包后得到的文件MinGW-w64移动到C:\mmingw64。然后打开Windows系统设置，搜索“编辑系统环境变量”并点击它。



图32. 在设置中搜索“编辑系统环境变量”

在系统属性窗口中点击“环境变量”，

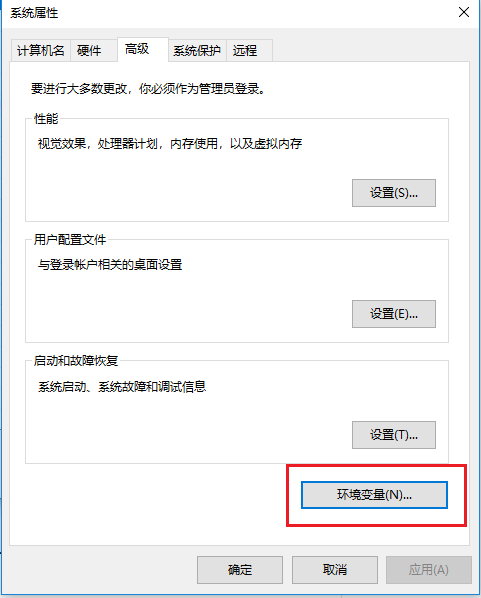


图33. 进入环境变量窗口

在环境变量窗口中，首先选中下方的“Path”，再点击编辑，进入编辑环境变量窗口。点击右方的“新建”按钮，输入“**C:\mingw64\bin**”作为新路径后再点击“确定”即可。

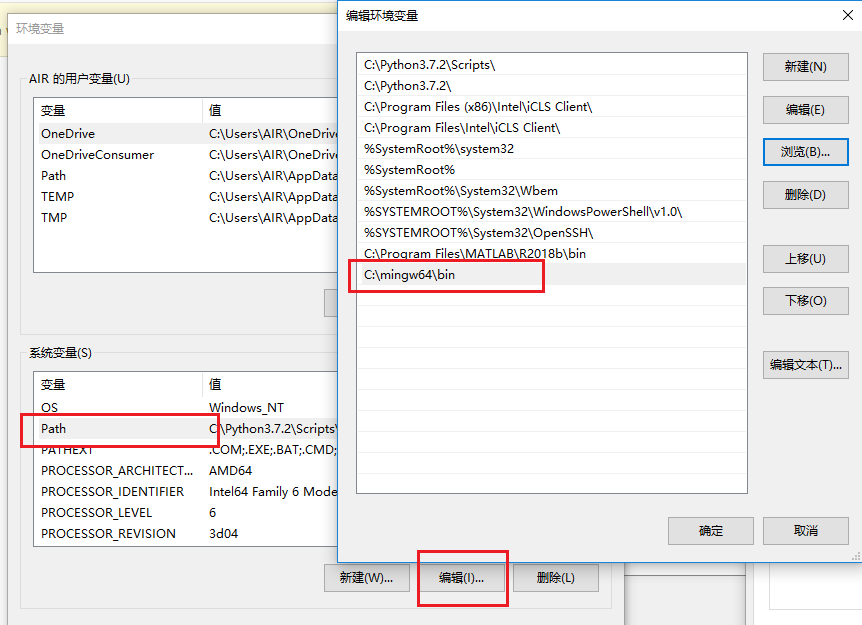


图34. 添加**C:\mingw64\bin作为新路径的具体操作**

**为验证是否已成功安装MinGW-W64，可打开命令提示符窗口输入> gcc --version，若成功输出的、gcc的版本号，则成功安装。**



图35. 成功安装**MinGW-W64**

# Visual Studio code的安装与使用

## 5.1 Visual Studio code的安装

Visual Studio Code是一个轻量但功能强大的源代码编辑器，可在桌面上运行，适用于Windows，macOS和Linux。它内置了对JavaScript，TypeScript和Node.js的支持，并为其他语言（如C，C++，Python，PHP，Go）和运行（如.NET和Unity）提供了丰富的扩展系统。

首先进入Visual Studio code官网：<https://code.visualstudio.com/> 下载软件安装包，解压后直接安装即可。

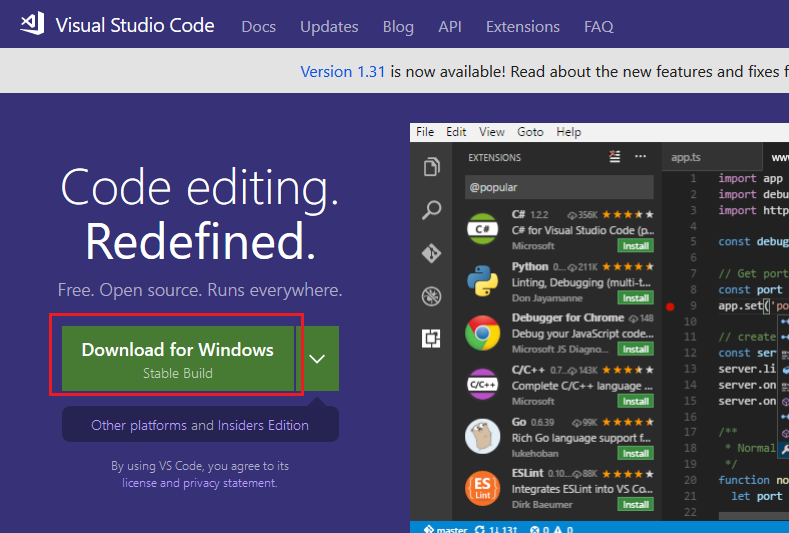


图36. VS Code官网

## 5.2 Python，C/C++插件的安装

安装成功后，打开VS Code，进入扩展功能区搜索Python与C/C++，点击“install”选项即可安装。



图37. 搜索安装Python插件

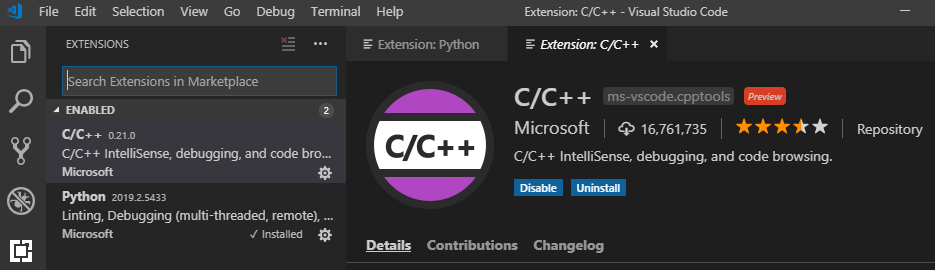


图38. 搜索安装C/C++插件

## 5.3 VS Code简单编程

### 5.3.1 简单的Python程序

在VS Code中将自己需要的文件夹打开在工作区，再在该目录下新建后缀名为.py的文件。以此次为例，我是在C:\Users\AIR\Desktop\软件基础\P1\VS Code的目录下新建了名为FirstTry.py的文件。输入好自己的程序之后，将FirstTry.py文件在终端打开，然后同时按下“shift”+“enter”执行程序。

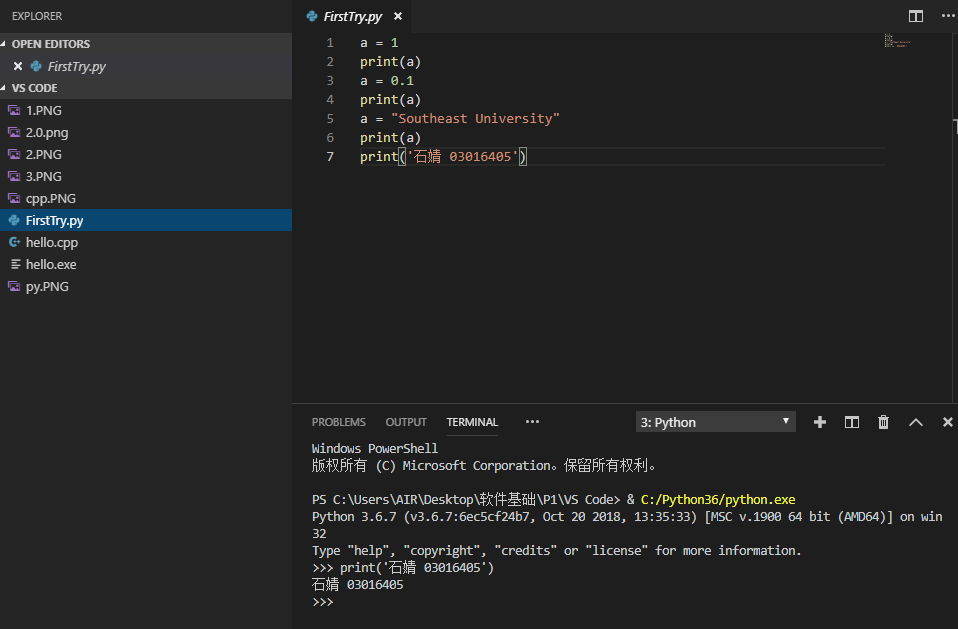


图39. 简单的Python程序

### 5.3.2 简单的C++程序

同样的，在VS Code中将自己需要的文件夹打开在工作区，再在该目录下新建后缀名为.cpp的文件。以此次为例，我是在C:\Users\AIR\Desktop\软件基础\P1\VS Code的目录下新建了名为hello.cpp的文件。输入C++代码后，将hello.cpp在终端中打开。由于VS Code无法直接编译.cpp格式的文件，并且编译姓名字符串时可能会出现乱码，因此需要先在终端输入>g++ -o hello hello.cpp -fexec-charset=GBK，再输入 > ./hello才能成功编译。C++程序如下图：

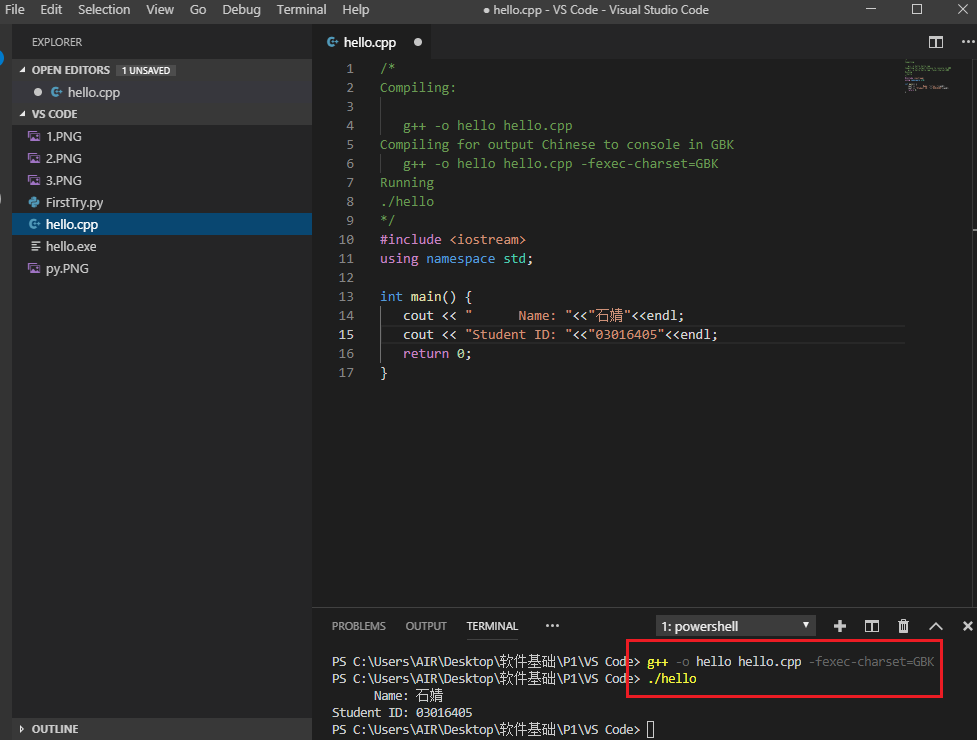


图40. 简单的C++程序

# 遇到的问题及解决方法

问题一：

使用Jupyter Notebook时始终无法连接到服务器，程序无法编译。

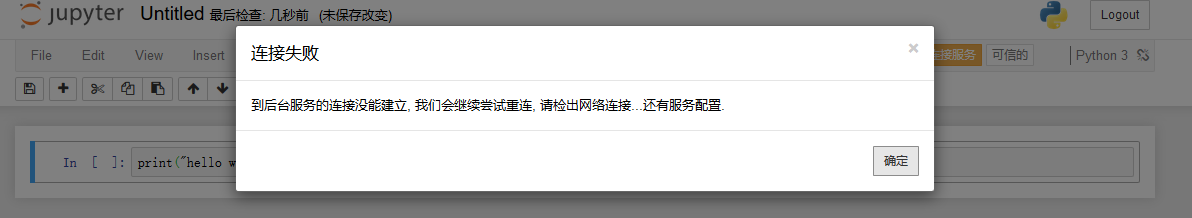


图41. Jupyter Notebook无法连接到服务器

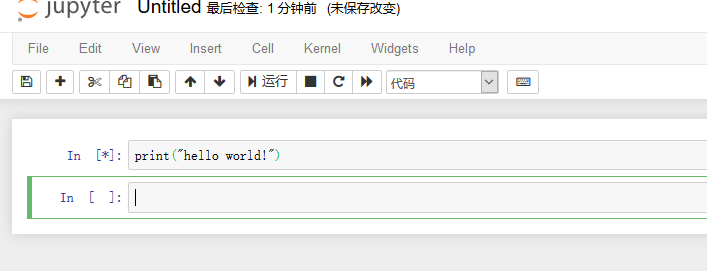


图42. Jupyter Notebook无法编译程序

解决方法：

发现问题后，我首先以“编译Jupyter Notebook时出现In[\*]”为关键词在网上搜索可行的解决方案。但经过多次尝试后仍然无法连接服务器。于是我在一节课下课后询问了老师。老师当时推断可能是Python下载的版本过新，可能会导致一些小问题，这时需要回退版本。在老师的帮助下，我下载了Python3.7.6。

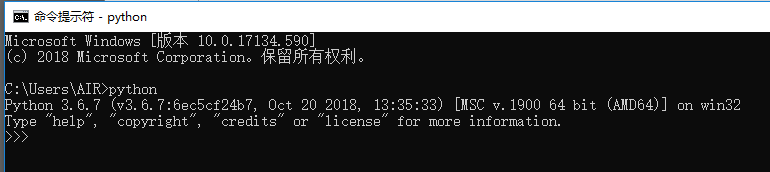


图43. 重新下载后的Python版本

但是Jupyter Notebook仍然无法连接上服务器。在下一次的课堂上老师为我们提出了解决方案。Tornado是一种 Web 服务器软件的开源版本。Tornado 和现在的主流 Web 服务器框架（包括大多数 Python 的框架）有着明显的区别：它是非阻塞式服务器，而且速度相当快。而Python也是基于Tormado的一个软件。3月初，Tornado更新了新版本，而我当时下载的Python还没有更新以适配Tornado，所以会出现安装的Jupyter连接Python3内核时异常的情况。解决问题可以通过卸载现在安装好的Tornado，然后安装旧版本的Tornado。具体操作时像命令提示符窗口输入“>python -m pip uninstall tornado”和“>python -m pip install tornado==5.1.1”。

问题二：

下载Git安装包时多次出现下载失败的情况。

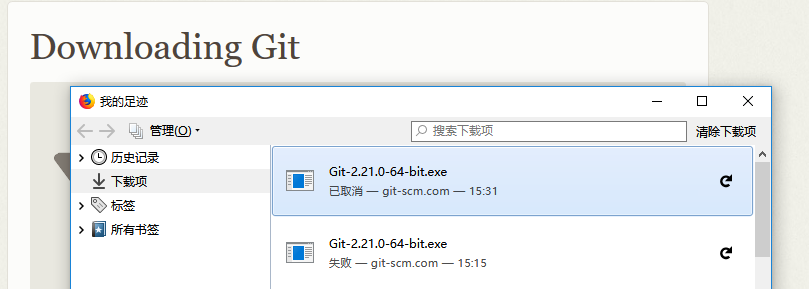


图44. Git下载失败

解决方法：

我首先在网上进行了搜索，发现可能是与git服务器的连接中断。网上的解决方法有好几种，但大部分解决方法的操作我看不懂，于是，我采用了网友提供的最简单的方案，即反复尝试。在经过一次又一次的重新下载乃至关机重启后，我最终成功下载好了Git安装包。

问题三：由C++程序文件编译生成的运行文件向终端输出时出现中文乱码。

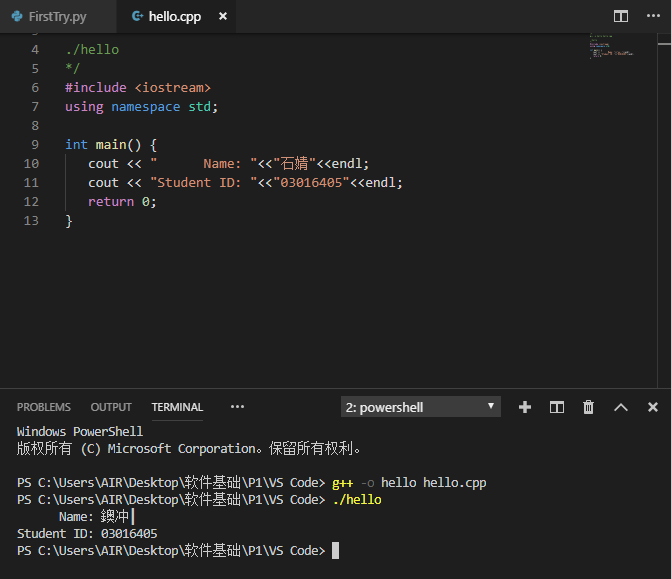


图45. 终端输出时出现中文乱码

解决方法：

发现问题后，我再次在网上进行搜索，但并没有找到有效信息，于是，我向老师发邮件提出了自己的疑问并很快的得到了老师的耐心回复。问题的原因是VS Code是跨平台软件，所有的纯文本类型文件的默认字符集编码都是UTF-8,但是Windows终端的默认字符集编码GBK。两者的默认字符集编码不同，C/C++程序输出UTF-8编码字符不能被GBK编码正确解析，就显示乱码.。解决方法是使用gcc编译选项，编译生成输出字符的中文编码为GBK的运行文件，即在终端输入“>g++ -o hello hello.cpp -fexec-charset=GBK”。

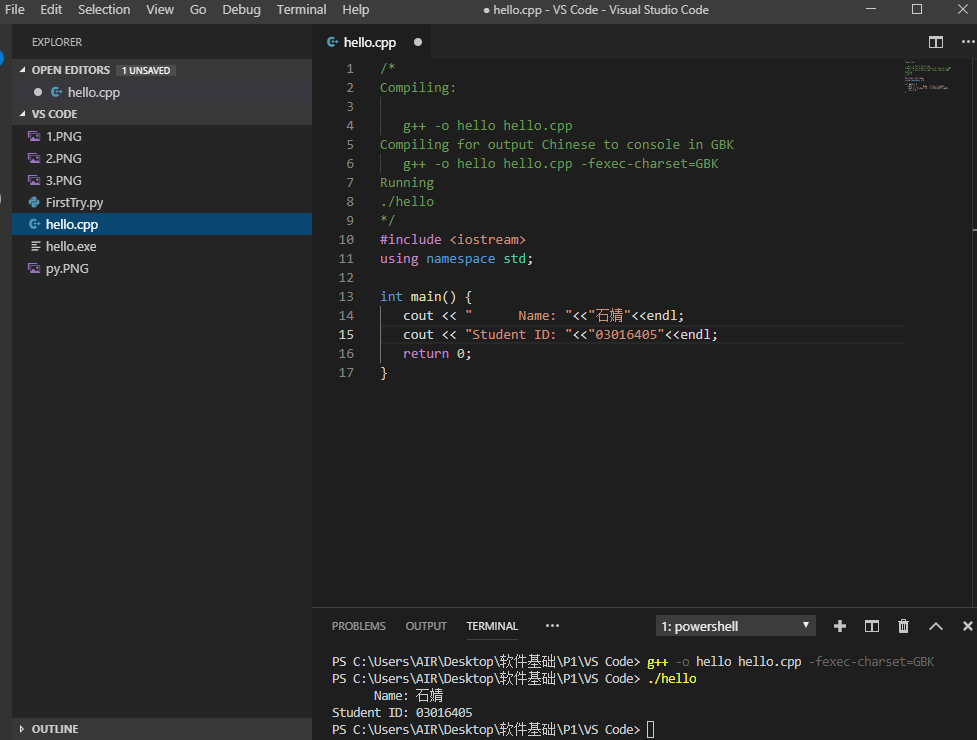


图46. 成功输出姓名字符串

# 个人小结

正确建立软件开发环境是学习这门课的一个前提，但由于这是我第一次接触到Python、VS Code这些软件，我在完成完成这篇报告时还是遇到的不少麻烦，如软件包下载失败、看不懂软件界面、程序无法编译或出现乱码等。

在安装软件的过程中遇到问题，我一般会先去网络上寻找解决方案，但我总是无法找到有效信息。这个问题不仅体现在软件工程基础及实践这门课程中，也体现在其他学科中。网上信息很多，但是否正确与有效仍需我们自己去甄别。如何高效且有效地进行网络信息检索是我继续提高的一个方面。在建立软件开发环境时，老师的“The Guide of Building Software Environment”讲义为我提供了很多帮助。根据老师在网页上列出的步骤一步一步地仔细进行，我便能很好的完成软件安装。此外，无论是课后或是发邮件向老师咨询问题，老师也能很耐心的为我解答。

另一方面，我也在完成报告的过程中发现了学习英语的重要性。各种程序都是英文界面，它们的网站也都是英文界面。如果我们想在官方网站查看软件的使用指导，那么我们必须具有良好的英语基础。GitHub上各开发者绝大多数也是用英文来编写代码和书写注释等。熟练的掌握英语在学习软件中起到了重要作用。

最后，尽管我已经成功安装了各个软件，但这仅仅是一个开始。我对于如何使用Python、VS Code和Jupyter Notebook进行代码编写仍然知之甚少，目前也仅掌握了这些软件最基本的使用功能。因此我仍需更进一步的学习才能更好的掌握这些语言。希望学期末的我能够对这些语言与软件有更深的理解。

# 参考文献

[1] 程懋华, “The Guide of Building Software Environment,” 3 2019. [联机]. Available: https://github.com/PySEE/home/blob/S2019/guide/BuildingSoftwareEnvironment.md.

[2] 程懋华, “Problem and Solution,” 3 2019. [联机]. Available: https://github.com/PySEE/home/blob/S2019/guide/Problem\_Solution.md.

[3] 姚依晨, “软件工程基础及实践课程实践作业二,” 南京, 2017.

[4] 吉珣碧, “软件工程基础及实践课程实践作业二,” 南京, 2017.

[5] Visual Studio Code, “C/C++ for Visual Studio Code (Preview),” Visual Studio Code, 22 8 2017. [联机]. Available: https://code.visualstudio.com/docs/languages/cpp. [访问日期: 13 3 2019].