

实验课预备指南

1. 编程语言：Python

Python 是一种易学且功能强大的编程语言。它具有高效的高级数据结构，简单而有效的面向对象编程，以及优雅的语法。

安装 Python

1. 建议通过安装软件包Anaconda的形式安装（详见下面的第3点）

编写你的第一个 Python 程序

在安装好 Python 后，打开你的文本编辑器（如 Notepad，VS Code 等），然后输入以下代码：

```
print("Hello, World!")
```

在本次实验课代码演示的Python版本为: **Python 3.8.16**

2. 实验平台：Visual Studio Code

Visual Studio Code (VSCode) 是由微软开发的一款免费、开源的代码编辑器。它支持多种编程语言，如 JavaScript、Python、C++、Java 等，并且拥有许多强大的功能，例如代码自动完成、内置的 Git 集成、调试工具等。

如何安装 VSCode

1. 前往 VSCode 官方网站 <https://code.visualstudio.com/> 下载适合你操作系统的安装包。
2. 下载完成后，运行安装包并按照提示进行安装。
3. 安装完成后，打开 VSCode，你将看到一个欢迎界面，可以从这个界面开始使用 VSCode。

3. 软件包：Anaconda

Anaconda 是一个开源的 Python 和 R 的发行版，用于进行大规模数据处理、预测分析和科学计算。它旨在简化包管理和部署。Anaconda 的包使用软件包管理系统 Conda 进行管理。

如何安装 Anaconda

1. 前往 Anaconda 的官方网站 <https://www.anaconda.com/> 并下载适合你操作系统的 Anaconda 版本。
2. 下载后，运行安装程序并按照提示完成安装过程。

在 Windows 的 VSCode 中配置 Anaconda 环境

1. 首先确保 Anaconda 已经安装并配置好。
2. 在 VSCode 中打开终端 (`Terminal > New Terminal`) 。
3. 输入 `conda create -n myenv python=3.8` 创建一个新的 Anaconda 环境。这里的 "myenv" 是你的环境名称, "3.8" 是 Python 的版本, 你可以根据需要进行更改。
4. 使用命令 `conda activate myenv` 激活新创建的环境。
5. 在激活的环境中, 你可以使用 pip 或者 conda 安装 Python 包。例如: `pip install numpy` 或者 `conda install numpy`。

在 macOS 的 VSCode 中配置 Anaconda 环境

1. 首先确保 Anaconda 已经安装并配置好。
2. 在 VSCode 中打开终端 (`Terminal > New Terminal`) 。
3. 输入 `conda create -n myenv python=3.8` 创建一个新的 Anaconda 环境。这里的 "myenv" 是你的环境名称, "3.8" 是 Python 的版本, 你可以根据需要进行更改。
4. 使用命令 `source activate myenv` 激活新创建的环境。
5. 在激活的环境中, 你可以使用 pip 或者 conda 安装 Python 包。例如: `pip install numpy` 或者 `conda install numpy`。

在 VSCode 中配置 Python 解释器

1. 在 VSCode 中打开 Python 文件, 点击左下角的 Python 版本信息。
2. 在弹出的选项中选择 "Enter interpreter path"。
3. 选择你的 Anaconda 环境中的 Python 解释器。例如, 对于 Windows, 路径可能是 `./Anaconda3/envs/myenv/python.exe`; 对于 macOS, 路径可能是 `./anaconda3/envs/myenv/bin/python3`。

以上只是配置 Anaconda 和 VSCode 的基础步骤, Anaconda 和 VSCode 还有许多其他的配置选项和功能需要自行谷歌或者百度。

4. pip 指令

pip 简介

pip 是 Python 的包管理工具, 可以用来安装 Python 包。它可以从 Python 包索引 (Python Package Index, PyPI) 中搜索、下载和安装 Python 包。

以下是一些基本的 pip 命令, 这些指令都是在终端执行:

- 安装包: `pip install package-name`
- 卸载包: `pip uninstall package-name`

- 升级包: `pip install --upgrade package-name`
- 列出已安装的包: `pip list`
- 查找包: `pip search package-name`

课程需要导入的包或者函数

- `import numpy as np`
- `import pulp as pl`
- `import random`
- `import copy as cp`

上述的包如果缺少的话，需要用pip指令去安装

5. Python中的PuLP软件包

PuLP 是一个 Python 的线性规划库。线性规划（Linear Programming，简称LP）是运筹学的一个分支，它涉及到在一组线性不等式约束条件下，最大化或最小化一个线性目标函数。PuLP 可以用来创建优化问题，然后使用选择的求解器来解决这些问题。PuLP也可以用来求解整数规划问题。

如何安装 PuLP

安装 PuLP 非常简单，只需要使用 pip 安装即可。在命令行中运行以下命令：

```
pip install pulp
```

代码中如果包缺少的话，需要用pip指令去安装

6. 检查实验环境是否配置完成

打开暑期学校的网址，在网站上下载[本次课程四个模型的Python代码](#)，若全部运行成功，则表示上述软件安装成功！