

《Python 语言及应用》教学大纲

课程代码:

课程名称: Python 语言及应用

英文名称: Python Programming Language and Its Application

课程性质: 专业课程

学分/学时: 2 学分/54 学时

考核方式: 闭卷考试

开课学期: 第 5 学期

适用专业: 纳米科学与技术 (器件与化学方向)

先修课程: 无

后续课程: 计算科学初探、量子化学

开课单位: 纳米科学技术学院

课程负责人: 纪玉金

大纲执笔人: 纪玉金

大纲审核人:

选用教材: 《Python 语言及其应用》(著: [美]Bill Lubanovic, 译: 丁嘉瑞, 梁杰, 禹常隆, 人民邮电出版社, 2020 年);

一、课程目标

Python 语言及应用课程共 54 学时, 授课内容承接公共基础专业课《计算机应用》, 在其基础上系统掌握 Python 基本概念、编程思想及程序设计技术, 具备熟练使用 Python 程序语言编写程序解决现实生活中的问题, 最终提高程序设计水平和计算机应用能力, 从而胜任科研、教学过程中常见需求。

具体教学目标为:

1. 掌握 Python 语言程序设计的基础语法与知识, 掌握程序设计的基本方法, 掌握程序设计的基本理论、方法和应用;
2. 培养学生的计算思维能力, 能够识读和编写程序, 掌握利用计算机 Python 程序设计解决问题的能力, 并通过实验训练学生的动手能力;
3. 培养学生创新能力和发现问题、分析问题和解决问题的能力, 使用计算机基础知识与程序设计来进行方案设计、理论模型构建、实际运行、数据处理等步骤, 解决实际问题。

二、教学内容

序号	章节内容	具体内容
1	Python 编程简介	1. 程序语言的发展历程 2. Python 语言的背景 3. Python 解释器的安装
2	Python 中的基础数据结构	1. 数字类型的定义 (整型、浮点数数据类型) 2. 数字类型的操作与运算 3. 字符串类型的定义 4. 字符串类型的操作与运算
3	Python 中的组合数据结构	1. 序列类型数据的定义 (元组、列表) 2. 元组、列表数据类型的操作与运算 3. 映射类型数据的定义 (字典)

		4. 字典类型数据的操作与运算 1
4	Python 中的流程控制	1. 分支结构—if-else 语句 2. 循环结构—while 语句
5	Python 中的函数	1. 函数的定义 2. 函数中的变量（形参与实参、局部与全局变量） 3. 函数的嵌套 4. 内置函数
6	Python 中的文件读写	1. 文件的内容的读取与写入
7	Python 模块与库函数	1. 模块与库函数的定义 2. 内置 math 函数库的简介与使用
8	Python 第三方库及其应用	1. numpy—科学计算库（数组与矩阵） 2. matplotlib—图表绘制函数库 3. scipy—科学计算库（插值、积分、常微分方程） 4. scikit-learn—机器学习函数库

三、考核方式

课程目标	考核内容	考核方式
培养学生的计算思维能力，了解脚本语言程序设计的基本知识，掌握程序设计的基本方法。	熟练掌握程序语言的基本语法、流程控制、函数等概念，能够识读相关程序，理解程序的算法及实现功能	课堂提问与讨论、课后作业、期中/期末考试
能够识读和编写程序，理解编程语言及应用方式，掌握利用计算机解决问题的能力，并通过实验训练学生的动手能力，解决实际问题能力	针对工程和科学研究特定的研究领域，熟练使用 Python 语言与特定算法编写程序完成一定需求	上机实践、课后作业、期中/期末考试
培养学生创新能力和发现问题、分析问题和解决问题的能力，使用计算机基础知识与程序设计来进行方案设计、理论模型构建、实际运行、数据处理等步骤。包括对复杂工程问题的预测与模拟，并能够理解其局限性。	根据实际的算法，完成程序编写、分析与调试程序，完成对数据处理、可视化等相关功能	上机实验操作、课后创新实践、实验报告

成绩评定方法：

	期中/末成绩权重	上机实验操作权重	课后作业权重	学生考勤及上课问答权重
课程目标 1	0.4	0.3	0.1	0.2
课程目标 2	0.3	0.4	0.1	0.2
课程目标 3	0.3	0.5	—	0.2

课程目标（即毕业要求指标点）达成度评价方法：

对于理论结合实践课程，平时成绩（包含课堂表现与上机实践）占比 20%，期中成绩占比 30%，期末成绩占比 50%。最终的计算公式为：

分目标达成度 = (平时平均分*平时权重*20%+期中平均分*期中权重*30%+期末平均分*期末权重*50%)/(100*平时权重*20%+100*期中权重*30%+100*期末权重*50%)

评分标准:

课程目标	90-100 (优秀)	75-89 (良好)	60-74 (及格)	0-59 (不及格)
培养学生的计算思维能力,了解脚本语言程序设计的基本知识,掌握程序设计的基本方法	完全具备 计算思维能力, 熟练掌握 脚本语言程序设计的基本知识, 理解 程序设计的基本方法。	具备 计算思维能力, 掌握 脚本语言程序设计的基本知识, 了解 程序设计的基本方法。	形成 计算思维能力, 掌握 脚本语言程序设计的基本知识, 了解 程序设计的基本方法。	缺乏 计算思维能力和脚本语言程序设计的基本知识, 只了解 程序设计的基本方法。
能够识读和编写程序,理解编程语言及应用方式,掌握利用计算机解决问题的能力,并通过实验训练学生的动手能力,解决实际应用问题的能力	熟练识读和编写程序 , 理解 编程语言及应用方式, 掌握 利用计算机解决问题的能力,并通过实验训练学生的 动手能力 。	能够识读和编写程序 , 理解 编程语言及应用方式, 掌握 利用计算机解决问题的能力,并通过实验训练学生的 动手能力 。	能够识读程序 , 理解 编程语言及应用方式,掌握利用计算机解决问题的能力,并通过实验训练学生的 动手能力 。	不能够识读程序 和 理解 编程语言及应用方式, 不能够 利用计算机解决问题的能力, 完全不具备动手能力
培养学生创新能力和发现问题、分析问题和解决问题的能力,使用计算机基础知识与程序设计来进行方案设计、理论模型构建、实际运行、数据处理等步骤。包括对复杂工程问题的预测与模拟,并能够理解其局限性。	完全具备 创新能力和发现问题、分析问题和解决问题的能力, 完全独立 使用计算机基础知识与程序设计来进行方案设计、理论模型构建、实际运行、数据处理等步骤。包括对复杂工程问题的预测与模拟,并能够 理解其局限性	具备一定 创新能力和发现问题、分析问题和解决问题的能力, 能够 使用计算机基础知识与程序设计来进行方案设计、理论模型构建、实际运行、数据处理等步骤,解决实际应用问题。包括对复杂工程问题的预测与模拟	初步形成 创新能力和发现问题、分析问题和解决问题的能力, 在教师/网络的协助下能够 使用计算机基础知识与程序设计来进行方案设计、理论模型构建、实际运行、数据处理等步骤,解决实际应用问题。	缺乏 创新能力和发现问题、分析问题和解决问题的能力, 缺乏 使用计算机基础知识与程序设计来进行方案设计、理论模型构建、实际运行、数据处理等步骤,解决实际应用问题。