**《Python程序设计》课程教学大纲**

**一、课程简介**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **课程中文名** | Python程序设计 | | | | | | |
| **课程英文名** | Programming in Python | | | | **双语授课** | | □是 ☑否 |
| **课程代码** | 10112161 | **课程学分** | 4 | **总学时数** | | 64 | |
| **课程类别** | □通识教育课程  □公共基础课程  🗹专业教育课程  □综合实践课程  □教师教育课程 | **课程性质** | ☑必修  □选修  □其他 | **课程形态** | | □线上  ☑线下  □线上线下混合式  □社会实践  □虚拟仿真实验教学 | |
| **考核方式** | ☑闭卷 □开卷 □课程论文 □课程作品 □汇报展示 □报告  ☑课堂表现 ☑阶段性测试 ☑平时作业 ☑其他 （可多选） | | | | | | |
| **开课学院** | 大数据与智能工程 | | **开课**  **系(教研室)** | 物联网工程系、数据科学与大数据技术系 | | | |
| **面向专业** | 物联网工程、数据科学与大数据技术 | | **开课学期** | 第3学期 | | | |
| **课程负责人** | 陈曦 | | **审核人** | 邢昌元、黄金龙 | | | |
| **先修课程** | 无 | | | | | | |
| **后续课程** | Python程序设计综合实训 | | | | | | |
| **选用教材** | 陈波、刘慧君，Python编程基础及应用，高等教育出版社，2020 | | | | | | |
| **参考书目** | 1．嵩天、礼欣、黄天羽，Python语言程序设计基础. 高等教育出版社, 2017  2．赵广辉等，Python程序设计基础. 高等教育出版社, 2021 | | | | | | |
| **课程资源** | 1．头歌实践教学平台 http://www.educoder.net  2．网易公开课——哈佛大学公开课CS50 <https://open.163.com/newview/movie/free?pid=EG295PU6R&mid=FG297O5KC>  3．https://www.kaggle.com/ | | | | | | |
| **课程简介** | 《Python程序设计》是一门面向物联网工程、数据科学与大数据技术专业的必修课程。本课程性质为基础性、实践性和应用性兼具，旨在培养学生掌握Python编程语言及程序设计基础，并具备利用Python解决本专业领域实际问题的能力。教学内容涵盖Python语言基础、程序控制结构、函数与模块、数据处理与文件操作等方面。通过学习，学生应能熟练运用Python进行基本编程，掌握数据处理与文件操作技巧。这将为学生今后在各自专业领域内应用Python进行数据处理、模拟仿真、算法实现等提供有力支持。 | | | | | | |

**二、课程目标**

**表 2-1 课程目标**

|  |  |
| --- | --- |
| **序号** | **具体课程目标** |
| **课程目标 1** | 能够读懂简单问题求解算法和程序，能够提取给定问题的基本数据，并能够用简单变量、组合数据类型变量、文件等进行描述。 |
| **课程目标 2** | 能够用结构化或面向对象程序设计思想、方法等描述或设计给定问题处理过程，并能够用规定形式表达。能够对所选择的数据表达和计算表达的功能进行分析确定功能的有效性。 |
| **课程目标** 3 | 具有较强的自主学习能力，能够正确选择和使用恰当的软硬件平台，能恰当运用信息化社会数据处理与问题求解自动化的基本方法与手段解决本专业领域数据处理、模拟仿真等问题。 |

**表2-2 课程目标与毕业要求对应关系（物联网工程专业）**

| **毕业要求** | **指标点** | **课程目标** |
| --- | --- | --- |
| 2.问题分析：能够应用数学、自然科学和工程科学的基本原理，识别、表达、并通过文献研究分析物联网复杂工程问题，以获得有效结论。【L】 | 2.2 通过文献研究、数据分析等方法，揭示物联网复杂工程问题内在规律，分析一个或多个模型的合理性，得到有效结论。 | 1 |
| 4.研究：能够基于科学原理并采用科学方法对物联网复杂工程问题进行研究，设计实验、分析与解释数据，并通过信息综合得到合理有效的结论。【H】 | 4.3 能选用、搭建或开发物联网应用系统实验环境，开展实验并正确记录、整理实验数据，对实验结果进行分析和解释，得到合理有效的结论，并确定结论的影响因素。 | 2 |
| 5.使用现代工具：能够针对大数据应用领域的复杂工程问题，选择和使用恰当的技术、资源和现代工程工具，进行信息表达、建模、设计、模拟、验证，并能够在实践中了解这些工具使用的局限性。【M】 | 5.2能够根据物联网复杂工程问题需求，利用网络查询、检索专业文献资料等方式获取技术资源。开发或选择恰当的软硬件平台、信息资源、现代工程工具和计算机专业模拟软件。 | 3 |

**三、课程学习内容与方法**

**（一）理论学习内容及要求**

**表3-1 课程目标、学习内容和教学方法对应关系**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **序号** | **课程模块** | **学习内容** | **学习任务** | **课程目标** | **学习重点难点** | **教学方法** | **学时** |
| 1 | Python快速入门 | * 课程说明 * Python极简史 * 解释执行 * 工具和软件 | 1.预习：整章教材、课件  2.线上学习：模块一视频内容  3.个人作业：  线上平台自测训练——Python语言基础  4.拓展阅读：  （1）Python之禅  （2）PEP8 编码规范  （3）WPS-求伯君-中国第一代程序员  （4）对Python学习者建议  （5）Python辅助学习网站  （6）《Python编程之美》阅读编写规范和autopep8工具的使用方法。 | 1 | 重点：  1. Python语言开发环境配置  2. Python程序语法元素分析  3. 第三方库的两种导入方法 | 讲授法：  尝试引导学生从国际视野看待问题，注重工程伦理问题，同时促进学生学科交叉融合思考问题解决方案。 | 2 |
| 2 | 数据类型 | 1.基本数据类型与运算   * print * 变量与变量命名 * 匈牙利命名法则 * 简单数据类型 * 类型转换 | 1.预习：整章教材、课件  2.线上学习：模块二视频内容  3.练习：  线上平台自测训练——数据类型、序列类型、集合与字典  4.个人作业：在线项目实训  5.拓展阅读：  （1）浮点数计算误差的处理  （2）统计python程序运行时间的方法  （3）字符编码以及UTF-8和UTF-8-sig的区别  （4）小数据池  （5）运算符与原地运算符  （6）十赌九输——赌徒博弈 | 1，2 | 重点：  1.字符串的表示  2.字符串的通用操作  3.字符串处理方法  4.整型、浮点型的应用与类型转换  5.序列通用操作、列表定义、应用、排序  6.使用集合去除重复数据。  7.使用字典进行词频统计。  难点：  1.索引与切片、split()方法  2.字符串与数值类型的转换  1.列表的应用  2.嵌套列表的操作  3.字典的灵活应用 | 讨论法：  课前提出“数值类型转换的方法种类与应用场景”、“分析几种幂运算方法的异同”、“分析列表的不同复制或赋值方法的异同点”、“如何利用pandas和numpy 读写数据”等讨论问题，课上组织学生讨论，引导学生思考，提升学生沟通能力和团队协作能力。 | 12 |
| 2.语法初步   * 缩进 * 操作符/运算符 * 数值运算及其优先级 * 函数与模块 * 获取用户输入 * 进制 * 注释 |
| 3.列表与元组   * 列表元素访问和修改 * 元素的增加、删除 * 列表的嵌套 * 列表快速生成 * 列表运算 * 成员函数 * 列表的复制 * 切片 |
| 4.字符串   * 字符串格式化 * center()函数 * find()函数 * join()函数 * replace()函数 * split()函数等 |
| 5.集合与字典   * 创建 * dict函数 * 基本操作 * 字典的嵌套 * 字典-字符串映射替换 * 成员函数 * 字典的遍历 |
| 3 | 程序结构 | 1.程序设计的三种基本结构   * 序列解包与链式赋值 * 条件、断言 * for循环、while循环 * break、continue语句 * 循环else子句 * 双重循环 * 序列缝合与循环解包 * 带下标的遍历、反向遍历 * del、exec、eval | 1.预习：整章教材、课件  2.线上学习：模块三视频内容  3.练习：线上平台自测训练：流程控制、函数  4.个人作业：在线项目实训  5.拓展阅读：  （1）https://cs50.harvard.edu/ai/2020  （2）利用faker模块产生模拟数据  （3）为函数加文档测试  （4）《像计算机科学家一样思考Python（第2版）》 | 2 | 重点：  1.程序调试。  2.函数的编写与调用  难点：  1.分支结构的逻辑关系和写法  2.循环结构的逻辑  3.lambda函数的使用 | 讲授法：  引导学生应用计算思维设计程序。  案例教学：  以实际案例展开教学，辅助学生构建工程思维。 | 6 |
| 2.函数和代码复用   * 函数定义、函数参数 * 作用域 * 递归 |
| 4 | 文件和异常 | 1.文件   * 基本操作 * 标准输入、输出、错误流、管道重定向 * 文件内容迭代 * json、csv、excel文件读写 | 1.预习：整章教材、课件  2.线上学习：模块四视频内容  3.练习：  线上平台自测训练：  文件、Matplotlib数据可视化  4.个人作业：在线项目实训  5.拓展阅读：  （1）圆周率数据处理与应用(工程思想)  （2）提取视频中的文字  （3）绘制人物关系图  （4）《Python编程快速上手》中涵盖了pdf文件、word文件、excel电子表格问的各个模块。 | 2，3 | 重点：  1.文本文件的读写方法  2.CSV文件的读写方法  难点：  1.统计分析方法  2.使用python的结构处理文件中的数据 | 查阅文献：  引导学生查阅文献，会使用帮助文档；了解学科前沿知识，能够应用AIGC辅助设计。 | 6 |
| 2.异常处理   * 曾经的异常 * try except else finally * 警告 * 出错日志 |
| 3.数据可视化 |
| 5 | 面向对象 | * OOP概述 * 创建类、创建对象 * 函数名重载 * 继承 * 父对象与子对象 * 类对象的属性和方法 | 1.预习：整章教材、课件  2.线上学习：模块五视频内容  3.练习：  线上平台自测训练：  面向对象  4.个人作业：在线项目实训 | 2 | 重点：  1.创建和使用类  2.继承  难点：  重写 | 讲授法：  引导学生运用面向对象思想设计程序。  案例教学：  以实际案例展开教学，辅助学生构建工程思维。 | 3 |
| 6 | 模块及扩展库 | * Python扩展库 * 创建使用模块 * 模块测试 * 包 * 应用程序的目录结构 * 让程序找到你的模块 * 标准库 * 扩展库的安装方法 | 1.预习：整章教材、课件  2.线上学习：模块六视频内容  3.练习：  线上平台自测训练：  Numpy  4.个人作业：在线项目实训  5.拓展阅读：  （1）《Python自然语言处理》  （2）openCV官网 | 3 | 重点：  扩展库的使用  难点：  模块测试 | 讲授法：  引导学生会阅读帮助文档，能够使用第三方库设计程序。 | 3 |

**（二）实验学习内容及要求**

**表3-2 课程目标、学习内容和教学方法对应关系**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **序号** | **项目名称** | **项目来源** | **教学目标（观测点、重难点）** | **学时数** | **项目类型** | **要求** | **每组人数** | **教学方法** | **课程目标** |
| 1 | 实验1：初识Python语言 | 工程实践 | 学生能够配置Python语言开发环境，完成 Python程序语法元素分析，根据实际题目要求会仿照已知案例编写代码。 | 2 | 设计研究 | 必做 | 1 | 实验指导 | 1 |
| 2 | 实验2：深入Python语言 | 工程实践 | 能够设计规范的简单问题求解算法，并完成相应的Python程序设计与实现。 | 20 | 设计研究 | 必做 | 1 | 实验指导 | 2、3 |
| 能够提取给定问题的基本数据，并能够用简单变量、字符串类型变量、序列类型变量、集合与字典、文件等进行描述， |
| 能够用结构化程序设计思想和方法设计和描述问题处理过程，并能够用函数表达。 |
| 能够对所选择的数据表达和计算表达进行分析，能区分错误与异常，会恰当运用异常处理等技术手段保证功能的有效性。 |
| 3 | 实验3：运用Python语言 | 工程实践 | 掌握程序设计方法，能够对所选择的数据表达和计算表达的功能进行分析。 | 10 | 综合性 | 必做 | 1 | 实验指导 | 3 |
| 会使用现代化手段持续学习,具备基本的工程伦理素养以及与大模型沟通的能力，会使用第三方库解决大数据领域的工程应用问题。 |

## 四、课程考核

**（一）考核内容与考核方式**

**表4 课程目标、考核内容与考核方式对应关系**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **课程目标** | **考核内容** | **所属**  **学习模块/项目** | **考核占比** | **考核方式** |
| 课程  目标1 | 1. Python程序实例解析 | 模块2/项目1 | 8% | 课堂表现  作业完成情况  实验  阶段性测试  闭卷考试 |
| 2.基本数据类型 | 模块1/项目2 |
| 课程  目标2 | 1.程序的控制结构 | 模块3/项目2 | 72% | 课堂表现  作业完成情况  实验  阶段性测试  闭卷考试 |
| 2.函数和代码复用 | 模块3/项目2 |
| 3.基本运算 | 模块2/项目2 |
| 4.文件操作 | 模块4/项目2 |
| 5.序列类型 | 模块2/项目2 |
| 6.集合与字典 | 模块2/项目2 |
| 7.面向对象 | 模块5/项目2 |
| 课程  目标3 | 1.数据可视化 | 模块4/项目3 | 20% | 课堂表现  作业完成情况  实验  阶段性测试  闭卷考试 |
| 2.异常处理 | 模块4/项目2 |
| 3.常用第三方库的使用 | 模块6/项目2、3 |

**表4-2 课程目标与考核方式矩阵关系**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 课程  目标 | 考核方式 | | | | | 考核占比 |
| 期末考试成绩比例50% | 课堂表现比例5% | 作业完成情况比例15% | 实验比例15% | 阶段性测验比例15% |
| 课程目标1 | 5% | 10% | 10% | 15% | 10% | 8%=5%\*50%+10%\*5%+10%\*15%+15%\*15%+10%\*15% |
| 课程目标2 | 80% | 60% | 70% | 65% | 60% | 72%=80%\*50%+60%\*5%+70%\*15%+65%\*15%+60%\*15% |
| 课程目标3 | 15% | 30% | 20% | 20% | 30% | 20%=15%\*50%+30%\*5%+20%\*15%+20%\*15%+30%\*15% |

## （二）成绩评定

**1.平时成绩评定**

**（1）课堂表现（10%）**：通过学生在课堂上的表现情况、发言与讨论情况，来评价学生相关的能力。

**（2）作业完成情况（30%）**：围绕课程的学习目标进行作业设计。如让学生简述对知识的认识，考核学生对于概念的理解情况，帮助学生将理论转化为自己的理解并实践。

**（3）实验（30%）**：学生收集资料能力，研究设计能力，设计程序解决实际问题能力和合作研究能力。

**（4）阶段性测验（30%）**：每完成一个阶段的学习，就进行阶段性测验，用来测验各模块掌握情况，考核方式为上机考试。

**2.期末成绩评定**

主要考察学生对程序设计基本概念、操作程序和具体方法的理解与运用等。考核方式为上机考试。要求学生掌握基本概念、操作程序，运用具体方法解决相关问题。

**3.总成绩评定**

总成绩由平时成绩、期末考核成绩构成。

构成比例总成绩（100%）= 平时成绩（50%）+ 期末成绩（50%）

## （三）评分标准

## 1.平时成绩

## （1）课堂表现：参与投票、问卷、抢答、选人、讨论、随堂练习等各类课程活动可以获得相应分数。

## （2）作业完成情况：按作业的平均分计分。

## （3）实验：取学生各个实验成绩平均分。

## （4）阶段性测验：取学生阶段性测验平均分，未做测验按“0”分计算。

## 2.期末成绩以机考试卷参考答案及评分细则为准。

## 五、其它说明

本课程大纲依据2023版专业人才培养方案，由大数据与智能工程学院物联网工程系、数据科学与大数据技术系讨论制定，大数据与智能工程学院教学工作委员会审定，教务处审核批准，自2023级开始执行。