

程序设计基础 课程标准 编制人：邱煜佳

一、课程基本信息

课程名称	程序设计基础（人工智能程序设计I）
课程性质	专业基础课
适用专业	人工智能技术应用
总课时	80课时（理论 32 课时+实践 48课时）
先修要求	无（面向零基础，逐步递进）
课程目标	<ol style="list-style-type: none">1. 掌握 Python 语言基础语法、数据类型、程序控制结构等核心知识，建立完整的 Python 编程知识体系；2. 熟练运用函数、文件操作、组合数据类型等实现基础编程任务，具备代码编写与调试能力；3. 了解面向对象编程、GUI 开发、数据库操作等进阶内容，能结合第三方库（如 NumPy、Pandas）完成数据处理、图形绘制等实践任务；4. 培养编程思维与问题解决能力，为后续专业课程（如数据分析、人工智能、Web 开发）奠定基础。

二、课程内容与课时分配

(一) 理论教学内容 (32 课时)

章节	教学内容	课时	教学重点与难点
第 1 章 Python 概述	1. Python 语言的基本特征 (优点、缺点、指南与帮助) 2. Python 程序开发环境 (下载安装、pip 工具、IDE 安装、程序路径、程序打包)	2	重点: Python 跨平台性、开源特性; pip 工具使用、PyCharm/Anaconda 安装配置 难点: 程序路径分隔符差异、pyinstaller 打包流程
第 2 章 数据类型与常用内置对象	1. 标识符与关键字 2. 数据类型 (数值型、布尔型、字符串、列表、元组、字典、集合) 3. 变量 (赋值、链式赋值、解包赋值、删除) 4. 运算符与表达式 (算术、赋值、关系、逻辑、成员、标识、位运算符) 5. 数据类型判断与转换 6. 常用内置函数与模块 (math、random、time)	4	重点: 列表/字典/集合的特性与区别; 运算符优先级; 内置函数 (如 len()、max()) 与模块调用 难点: 位运算符应用; random 模块随机数生成; time 模块时间格式转换
第 3 章 数据输入输出	1. 数据输入 (input()函数与类型转换) 2. 数据输出 (print()函数: 无格式、%格式化、format 格式化) 3. 语句 (书写规则、注释语	3	重点: input()与 eval()结合实现多类型输入; format()格式化输出技巧 难点: 复杂格式化场景 (如对齐、精度控制); 注释规范与代码可读性

章节	教学内容	课时	教学重点与难点
	句) 4. 顺序结构及程序举例		
第4章 程序控制结构	1. 选择结构 (单分支、双分支、多分支、嵌套) 2. 循环结构 (while、for、continue/break、else 子句、循环嵌套) 3. NumPy 模块与科学计算 (数组创建、算术运算、统计函数)	5	重点: 多分支选择结构逻辑; for 循环与 range()结合; NumPy 数组基本操作 难点: 循环嵌套逻辑设计; NumPy 数组索引与切片; 线性代数应用 (如矩阵乘)
第5章 字符串	1. 字符串基本概念 (定义、索引、切片、运算) 2. 字符串常用操作 (对齐、大小写转换、删除、查找替换、分割连接、判断)	3	重点: 字符串切片技巧; find()/replace()/split()等方法应用 难点: 复杂字符串处理 (如子串统计、中英文混合处理)
第6章 组合数据类型与中文分词	1. 列表 (创建、访问、切片、添加/修改/删除元素、排序) 2. 元组 (创建、访问、与列表的区别) 3. 字典 (创建、访问、添加/修改/删除键值对) 4. 集合 (创建、运算、去重) 5. 中文 jieba 分词 (安装、三种分词模式、自定义词典)	4	重点: 列表方法 (append()/sort()); 字典键值对操作; 集合去重与交集/并集运算 难点: jieba 分词模式选择; 组合数据类型嵌套应用 (如列表嵌套字典)
第7章 函数与图形绘	1. 函数 (定义、调用、参数传递、递归、高阶函数、变	5	重点: 函数参数传递 (位置参数、关键字参数、默认参

章节	教学内容	课时	教学重点与难点
制	量作用域) 2. 自定义模块与包 3. 图形绘制 (Turtle 库基础绘图、Matplotlib 库折线图/柱状图/饼图)		数) ; 递归函数设计; Turtle 绘图逻辑 难点: 高阶函数 (map()/filter()) ; Matplotlib 中文显示与图形美化
第 8 章 文件读写与数据处理	1. 文本文件读写 (打开/关闭、读写操作、指针定位、with 语句) 2. 常用文件读写 (CSV、Excel、二进制文件) 3. Pandas 数据处理 (数据类型、读写文件、数据分析) 4. 网页数据爬取 (原理、爬取与解析技术)	3	重点: 文本文件 with 语句安全操作; CSV/Excel 文件读写; Pandas DataFrame 基本操作 难点: 二进制文件 struct 库应用; Pandas 数据筛选与统计; 网页爬取解析 (如 bs4)
第 9 章 面向对象程序设计	1. 面向对象概念 (封装、继承、多态) 2. 类和对象 (定义、创建、属性、方法) 3. 继承与多态实现	2	重点: 类的定义与对象创建; __init__ 构造方法; 继承语法 难点: 多态实现; 私有属性与方法封装
第 10-12 章 进阶应用 (整合)	1. 程序异常处理 (try-except、自定义异常) 2. GUI 程序设计 (Tkinter 组件、布局管理器、事件处理) 3. 数据库程序设计 (SQLite/MySQL 基本操作、Python 连接)	1	重点: 异常处理逻辑; Tkinter 简单窗口开发; 数据库连接与查询 难点: GUI 事件绑定; 数据库事务处理

(二) 实践教学内容 (48 课时)

实践项目	实践内容	课时	实践目标	对应实验教材模块
实践 1: Python 环境搭建与基础语法	<ol style="list-style-type: none"> 1. 安装 Python、PyCharm/Anaconda 2. 编写"Hello World"程序、简单算术运算 3. 练习变量赋值与数据类型转换 	6	<ol style="list-style-type: none"> 1. 掌握 Python 开发环境配置 2. 熟练使用 IDLE/PyCharm 编写运行基础代码 3. 理解数据类型转换逻辑 	Python 开发环境搭建与基础语法实践
实践 2: 数据类型与表达式练习	<ol style="list-style-type: none"> 1. 列表/字典/集合的创建与基本操作 2. 复杂表达式计算 (如算术+逻辑运算符组合) 3. 内置函数 (math/random) 应用 	6	<ol style="list-style-type: none"> 1. 熟练操作组合数据类型 2. 掌握表达式优先级与复杂计算 3. 能使用 random 生成随机数据、math 实现数学运算 	数据类型与常用内置对象
实践 3: 程序控制结构编程	<ol style="list-style-type: none"> 1. 分支结构: 成绩等级判断、闰年判断 2. 循环结构: 累加求和、素数判断、九九乘法表 3. 循环嵌套: 矩阵打印、图形打印 (如三角形) 	6	<ol style="list-style-type: none"> 1. 能设计多分支逻辑解决实际问题 2. 熟练使用 for/while 循环实现重复任务 3. 掌握循环嵌套逻辑设计 	程序控制结构编程实践
实践 4: 字符串与组合数据类型应用	<ol style="list-style-type: none"> 1. 字符串处理: 文本统计 (字母、数字、空格个数)、回文判断 2. 列表/字典应用: 学生成绩管理 (添加、查询、排序) 	6	<ol style="list-style-type: none"> 1. 能实现复杂字符串分析任务 2. 熟练使用组合数据类型管理结构化数据 3. 掌握 jieba 分词 	字符串与组合数据类型应用实验

实践项目	实践内容	课时	实践目标	对应实验教材模块
	3. jieba 分词：中文文本分词与词频统计		与词频统计方法	
实践 5：函数与模块化程序设计	1. 自定义函数：奇偶数判断、阶乘计算、最大公约数 2. 递归函数：斐波那契数列、汉诺塔问题 3. 高阶函数：map()实现列表元素平方、filter()筛选偶数 4. 模块化程序设计	6	1. 能设计自定义函数解决特定问题 2. 理解递归思想并实现递归函数 3. 掌握高阶函数简化代码逻辑 4. 掌握模块化程序设计方法，能综合运用函数与模块完成小型系统	函数与模块化程序设计
实践 6：文件读写与数据处理	1. 文件读写基础 2. 文件与目录管理 3. 文件指针与遍历 4. 使用 jieba 统计数据	6	1. 掌握文本文件与 CSV 文件的打开、读写、关闭方法 2. 掌握文件指针定位、内容遍历、文件合并、重命名、复制、删除等常用操作 3. 能够创建目录结构并按需求完成文件移动、复制、删除任务 4. 能够使用 jieba 等第三方库对文本文件进行词频统计与数据分析	文件操作与数据持久化

实践项目	实践内容	课时	实践目标	对应实验教材模块
实践 7: 面向对象编程与异常处理	1. 面向对象编程 2. 程序异常处理	6	1.掌握类的定义、实例化、属性与方法的使用。 2.理解继承、多态、封装及运算符重载的实现方法。 3.掌握 @property、@staticmethod、@classmethod 装饰器的使用。 4.理解异常体系, 掌握 try-except-finally-else 结构及自定义异常。 能用面向对象思想结合异常处理完成小型综合项目。	面向对象编程与异常处理
实践 8: 综合项目实战	1. SQLite3 实战 2. MySQL 实战 3.基于菜单循环实现通信录/成绩管理等小型数据库应用系统	6	1. 能整合多章节知识实现小型应用 2. 具备问题分析与代码调试能力 3. 培养项目化编程思维	数据库技术综合实验

三、教学方法与手段

- 理论教学：**采用“讲授+案例+微课”三位一体模式，每章节围绕核心知识点设计 1-2 个实际案例（如成绩管理、数据统计），通过代码演示讲解语法规则与逻辑；结合微课视频（如环境搭建、复杂函数演示），嵌入理论教材对应章节，供学生课后复习。
- 实践教学：**采用“任务驱动+实验指导”模式，基于实验教材提供的步骤引导与问题解析，每次实践课明确任务目标与验收标准，学生自主完成后提交实验报告；针对共性问题集中讲解，鼓励学生分组讨论复杂任务（如递归逻辑、项目设计），培养协作能力。
- 辅助工具：**使用 PyCharm 作为主要 IDE，演示代码调试技巧（如断点、变量监视）；利用在线判题平台（如牛客网、LeetCode 简单题）布置课后练习，结合实验教材的课后习题巩固知识点。

四、考核方式与成绩评定

考核环节	考核内容	占比	说明
平时成绩	1. 课堂出勤与互动 2. 课后作业（语法练习、小型编程题，选自理论教材习题） 3. 实践课任务完成情况（含实验报告）	30%	作业每周 1 次，实践任务当堂提交+实验报告课后 3 天内提交，缺交 1 次扣 2 分，迟到/早退 3 次按旷课 1 次处理
期中测试	Python 基础语法、数据类型、程序控制结构、函数基础（理论+实践）	20%	闭卷笔试（理论知识，40 分钟）+机试（基础编程题，50 分钟），机试需在规定时间内完成 3-4 道编程题
期末考核	1. 理论部分：面向对象、文件操作、数据库、GUI 等进阶内容 2. 实践部分：综合编程题	50%	理论笔试（30 分钟）+机试（70 分钟），机试需整合多章节知识完成 1 个综合项目或 3-5 道编程题

考核环节	考核内容	占比	说明
	(如数据处理+可视化、小型系统设计)		
总评成绩	平时成绩 (30%) + 期中测试 (20%) + 期末考核 (50%)	100%	总评 ≥ 60 分为及格, ≥ 85 分为优秀

五、教材与参考资料

1. **主教材**: 《Python 程序设计基础 (微课版) 》, 李津、钟频主编, 人民邮电出版社。
2. **实验教材**: 《Python 程序设计基础实验指导 (微课版) 》, 配套主教材设计实验案例与操作步骤。
3. **参考资料**:
 - Python 官方网站: <https://www.python.org>
 - Python 官方文档 (中文版) : <https://docs.python.org/zh-cn/3/>
 - PyPI 官方包管理平台: <https://pypi.org>
 - 爬虫相关第三方库 BeautifulSoup 官方文档:
https://beautifulsoup.readthedocs.io/zh_CN/v4.4.0/
 - 数据分析与科学计算 NumPy 官方文档: <https://numpy.org/doc/stable/>
 - 数据分析与科学计算 Pandas 官方文档:
https://pandas.pydata.org/docs/user_guide/index.html
 - 数据可视化 Matplotlib 官方文档: <https://matplotlib.org/>

六、课程说明

1. 本大纲基于 80 课时设计，可根据专业需求调整章节侧重点（如计算机类专业可增加数据库、GUI 开发课时，非计算机专业可简化 NumPy、面向对象内容）。
2. 实践课需保证每人 1 台计算机，预装 Python 3.8+、PyCharm、Anaconda（含 NumPy、Pandas、Matplotlib）、jieba 等库，实验教材需提前分发至学生，确保实践前了解实验目标与步骤。
3. 鼓励学生课后参与开源项目、Python 竞赛，结合理论教材的拓展阅读与实验教材的进阶任务，提升编程实践能力。