

《Python 语言与应用 (Python Language and Application)》

教学大纲

制定时间：2024 年 4 月

一、课程基本信息

(一) 适用专业：智能科学与技术

(二) 课程代码：3ZN1007A

(三) 学分/课内学时：3 学分/48 学时

(四) 课程类别：专业教育

(五) 课程性质：必修/理论课

(六) 先修课程：《C 语言程序设计与应用》、《数据结构》

(七) 后续课程：《机器学习》、《数据分析与挖掘技术》、《人工智能》

二、课程教学目标

课程定位：

本课程是智能科学与技术专业教育必修课程，主要安排是在大学第二年级进行授课。通过对本门课程的学习,使学生系统地获得一门信息学科中常用的数据分析处理编程语言，掌握基本的编程技能，并注重培养学生抽象分析问题和设计算法、编程实现解决问题的能力 and 常见的程序设计能力、排错能力以及 Python 软件包查找、使用能力，从而使学生会利用程序语言去建模、解决一些数据分析、数据挖掘、人工智能等方面的数据处理问题。

总体目标：

本课程主要深入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，掌握程序设计的基本理论、方法和应用，掌握高级程序设计国家标准的有关基本规定，会查阅有关国家标准和手册，养成严格遵守和执行有关国家标准的各项规定的良好习惯。能够较正确而熟练地使用 Python 进行程序的设计；能够识读和编写较复杂程度的程序；能够使用 Python 解决实际问题。培养学生计算思维能力、创新能力和发现问题、分析问题和解决问题的能力。

教学目标：

课程目标是通过项目为导向的教学实践活动，使学生全面掌握基于 Web 的程序设计与开发技能，了解项目设计规范，熟悉项目设计方法，养成良好的项目开发习惯，培养团队协作精神。围绕课程目标，在课程改革过程中形成“创新式立体化教学模型”，该模型以高素质高技能人才培养为核心，以“阶梯递进”课程体系为主体，采取四段五化教学模式和组件嵌入式项目构建模式，充分体现教、学、做一体化。

(一) 具体目标

目标 1: 了解程序设计语言发展过程；理解 Python 语言的特点；掌握 Python 语言开发环境和运行环境配制方法；掌握 Python 语言的基本语法，包括缩进、变量命名等；掌握 Python 标准库的导入和使用；了解程序的异常处理方法。掌握 Python 语言中数字类型的概念和使用；运用 Python 的标准数据库进行数值计算；掌握字符串类型的概念和使用；掌握字符串类型的格式化操作方法和应用。

目标 2: 了解程序的基本结构并绘制流程图；掌握程序的分支结构，运用 if 语句实现分支结构；掌握程序的循环结构，运用 for 语句和 while 语句实现循环结构。掌握函数的定义和调用方法；理解函数的参数传递过程以及变量的作用范围；了解 lambda 函数；理解函数递归的定义和使用方法。

目标 3: 了解 Python 语言基本组合数据类型；理解列表概念并掌握 Python 中的列表使用；理解字典概念并掌握 Python 中的字典使用；运用列表管理采集的信息，构建数据结构；运用字典处理复杂的数据信息。掌握面向对象程序设计思想概述，理解类的基本概念，掌握类的定义和使用，理解类的继承和多态。

目标 4: 掌握文件的读写方法以及打开和关闭等基本操作；运用 tkinter 库进行基本的图像处理；多进程编程和进程的状态，掌握创建进程与进程枚举和终止等方法，理解线程的概念，掌握使用 threading 模块创建线程。掌握网络通信模型和 TCP/IP 协议簇、掌握 Socket 编程。

(二) 课程目标与毕业要求的对应关系

毕业要求	毕业要求指标点	课程目标	教学单元	评价方式
2. 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智能系统中的复杂工程问题，以获得有效结论。	观测点 2.5:能运用基本原理分析实际工程的影响因素，证实解决方案的合理性。	目标 1	Python 概述 Python 语言基础 Python 模块	平时作业 阶段测验 机试考核
4. 能够基于科学原理，采用适当的工程方法对智能系统的复杂工程问题进行研究，包括系统方案设计、分析与解释数据、并通过信息综合得到合理有效的结论。	观测点 4.1:能够采用正确的系统设计方法和工具分析智能应用系统的功能、性能和参数；	目标 2 目标 3	序列数据结构 Python 函数 Python 面向对象程序设计	平时作业 阶段测验 机试考核
5. 能够针对智能系统的复杂工程问题，开发、选择与使用恰当的技术、资	观测点 5.1:能运用信息技术、大型团队开发过程管理软件	目标 3 目标 4	Python 面向对象程序设	平时作业 阶段测验 机试考核

源、现代工程工具和信息技术工具,实现对复杂工程问题的预测与模拟,理解其局限性。	等现代工程工具和信息技术工具,对智能系统的设计开发过程进行管理;		计 I/O 编程 网络编程与 图形化界面	
---	----------------------------------	--	-------------------------------	--

三、教学内容与方法

(一) 教学内容及要求

序号	教学单元	教学内容	学习产出要求	推荐学时	推荐教学方式	支撑课程目标	备注
1	Python 概述	python 语言的发展历程、python 的运行环境。	了解 python 语言的发展历程;掌握 python 的运行环境以及 Python 程序的运行步骤。	2	讲授	目标 1	
2	Python 语言基础	Python 的简单输入输出语句;数据类型、变量、常量及表达式;逻辑控制语句(包括顺序、选择、循环)格式及使用。	理解 Python 的简单输入输出语句、数据类型、变量、常量及表达式;掌握逻辑控制语句(包括顺序、选择、循环)格式及使用。	2	讲授 案例实验	目标 1 目标 2	
3	序列数据结构	列表、元组、字典、集合的应用。	掌握列表、元组、字典、集合的概念及运用;掌握方法的运用、参数传递和返回值。	4	讲授 案例实验	目标 2 目标 3	
4	Python 函数	Python 中函数的声明与调用、变量的作用域、参数和返回值、内置函数。	掌握 Python 中函数的声明与调用,理解变量的作用域、参数和返回值,掌握基本内置函数。	4	讲授 案例实验	目标 2 目标 3	
5	Python 面向对象程序设计	面向对象程序设计思想概述、基本概念、类的定义和使用、类的继承和多态。	了解面向对象程序设计思想概述,理解类的基本概念,掌握类的定义和使用,理解类的继承和多态。	4	讲授 案例实验	目标 2 目标 3 目标 4	
6	Python 模块	Python 标准库中的常用模块、sys 模块、platform 模块、与数学和时间有关	了解 Python 标准库中的常用模块、掌握 sys 模块、platform 模块与数学和时间有关的模块;掌握导入模块与使	4	讲授 案例实验	目标 2 目标 3 目标 4	

序号	教学单元	教学内容	学习产出要求	推荐学时	推荐教学方式	支撑课程目标	备注
		的模块、导入模块与使用。	用。				
7	函数式编程	函数式编程的概念与优点、Python 函数式编程常用的函数、闭包和递归函数、迭代器和生成器。	掌握函数式编程的概念与优点，了解 Python 函数式编程常用的函数、闭包和递归函数、迭代器和生成器。	4	讲授 案例 实验	目标 2 目标 3 目标 4	
8	I/O 编程	数据的输入和显示、文件操作、目录编程。	掌握数据的输入和显示，掌握文件操作，掌握目录编程。	8	讲授 案例 实验	目标 2 目标 3 目标 4	
9	图形界面编程	常用 tkinter 组件的使用、消息框弹出、Windows 窗口创建、各种窗口组件、窗体布局、Tkinter 字体。	了解常用 tkinter 组件的使用，掌握消息框弹出、Windows 窗口创建的方法，掌握各种窗口组件、窗体布局、Tkinter 字体的使用。	8	讲授 案例 实验	目标 2 目标 3 目标 4	
10	多任务编程	多进程编程、进程的状态、创建进程与进程枚举和终止、线程的概念、threading 模块。	了解多进程编程和进程的状态，掌握创建进程与进程枚举和终止等方法，理解线程的概念，掌握使用 threading 模块创建线程。	6	讲授 案例 实验	目标 2 目标 3 目标 4	
11	网络编程	网络通信模型和 TCP/IP 协议簇、Socket 编程、电子邮件编程。	了解网络通信模型和 TCP/IP 协议簇、掌握 Socket 编程、电子邮件编程。	2	讲授 案例 实验	目标 2 目标 3 目标 4	

(二) 教学方法

1. 课堂讲授

(1) 采用启发式教学，激发学生主动学习的兴趣，培养学生独立思考、分析问题和解决问题的能力，引导学生主动通过实践和自学获得自己想学到的知识。

(2) 在教学内容上，系统讲授 Python 语言中所涉及的语句、数据结构、标准库引入、I/O 文件编程、多任务与网络编程等基本知识和基本方法，使学生能够系统掌握用于解决计算机等相关专业工程复杂问题的专业基础知识。

(3) 在教学过程中采用电子教案、实验教学相结合，主要通过机房上课，通

过安排提高课堂教学信息量，增强教学的直观性。

(4) 理论教学与工程实践相结合，引导学生应用数学、自然科学和工程科学的基本原理，采用现代设计方法和手段，实现基本的数据处理、分类和聚类、图形化编程、文字统计、网页内容爬取等实践性案例，提升专业相关工程问题提供思维方法和实践能力。

(5) 针对学生课程学习中的问题，在课内通过查看代码帮助学生，课外通过在线随时回答问题。

2. 专题研究

围绕各章教学重点内容，除布置一定数量的实验外，另设置非标准化的过程考核环节，以综合能力的锻炼为主，除了在技术方面的成长外，也提升素材收集、项目规划与实施的能力，培养学生逐步具有应用相关技能解决复杂工程问题的项目开发等能力，结合所研究课题进行报告和设计文稿的撰写，并清晰陈述观点和回答问题的能力。

3. 实验教学

实验教学是人工智能课程中重要的实践环节，目的是培养学生运用实验方法研究解决人工智能类专业复杂工程问题的能力。课程必做实验 8 个，各实验按照实验指导书的要求学生独立或分组完成，并提交实验报告。

四、考核及成绩评定

(一) 考核内容及成绩构成

课程目标	考核内容	成绩评定方式	成绩占总评分比例	目标成绩占当次考核比例	学生当次考核平均得分	目标达成情况计算公式
------	------	--------	----------	-------------	------------	------------

课程目标	考核内容	成绩评定方式	成绩占总评分比例	目标成绩占当次考核比例	学生当次考核平均得分	目标达成情况计算公式
目标 1：了解程序设计语言发展过程；理解 Python 语言的特点；掌握 Python 语言开发环境和运行环境配制方法；掌握 Python 语言的基本语法，包括缩进、变量命名等；掌握 Python 标准库的导入和使用；了解程序的异常处理方法。掌握 Python 语言中数字类型的概念和使用；运用 Python 的标准数据库进行数值计算；掌握字符串类型的概念和使用；掌握字符串类型的格式化操作方法和应用。	Python 程序设计语言开发环境搭建；Python 语言的基本语法；Python 标准库的导入和使用；字符串类型操作。	试卷考核	10%	100%	C ₁	$\frac{C_1}{20\%} \times 10\%$ 10
目标 2：了解程序的基本结构并绘制流程图；掌握程序的分支结构，运用 if 语句实现分支结构；掌握程序的循环结构，运用 for 语句和 while 语句实现循环结构。掌握函数的定义和调用方法；理解函数的参数传递过程以及变量的作用范围；了解 lambda 函数；理解函数递归的定义和使用方法。	分支结构和循环结构、函数的定义及调用、函数的参数传递、函数递归的定义和使用	平时作业	3%	100%	A ₁	$\frac{A_1}{100\%} \times 3\% + \frac{B_1}{100\%} \times 10\% + \frac{C_2}{100\%} \times 15\% +$ 28
		实验	10%	100%	B ₁	
		试卷考核	15%	28%	C ₂	
目标 3：了解 Python 语言基本组合数据类型；理解列表概念并掌握 Python 中的列表	列表、字典等数据结构的使用、面向对象编程、类的定义和继承	平时作业	3%	100%	A ₂	$\frac{A_2}{100\%} \times 3\% + \frac{B_2}{100\%} \times 10\% + \frac{C_3}{100\%} \times 20\%$ 33

课程目标	考核内容	成绩评定方式	成绩占总评分比例	目标成绩占当次考核比例	学生当次考核平均得分	目标达成情况计算公式
表使用；理解字典概念并掌握 Python 中的字典使用；运用列表管理采集的信息，构建数据结构；运用字典处理复杂的数据信息。掌握面向对象程序设计思想概述，理解类的基本概念，掌握类的定义和使用，理解类的继承和多态。		实验	10%	100%	B ₂	
		试卷考核	20%	100%	C ₃	
目标 4：掌握文件的读写方法以及打开和关闭等基本操作；运用 tkinter 库进行基本的图像处理；多进程编程和进程的状态，掌握创建进程与进程枚举和终止等方法，理解线程的概念，掌握使用 threading 模块创建线程。掌握网络通信模型和 TCP/IP 协议簇、掌握 Socket 编程。	文件的读写方法、图形化界面编程、网络编程、多线程	平时作业	4%	100%	A ₃	$\frac{A_3}{100\%} \times 4\% + \frac{B_3}{100\%} \times 10\% + \frac{C_4}{100\%} \times 15\%$ $\frac{\quad}{29}$
		实验	10%	100%	B ₃	
		试卷考核	15%	100%	C ₄	
总评成绩（100%）=实验（30%）+平时作业（10%）+机试考试（60%）			100%	—	—	$\frac{\text{学生总评平均分}}{100}$

（二）平时考核成绩评定

1.平时作业：共 3 次，支持目标 2、目标 3、目标 4，共占总评分 10%，目标 2 占 3%、目标 3 占 3%、目标 4 占 4%。对应目标的评分标准如下：

	对应目标	目标 2: 了解程序的基本结构并绘制流程图; 掌握程序的分支结构, 运用 if 语句实现分支结构; 掌握程序的循环结构, 运用 for 语句和 while 语句实现循环结构。掌握函数的定义和调用方法; 理解函数的参数传递过程以及变量的作用范围; 了解 lambda 函数; 理解函数递归的定义和使用方法。		目标 3: 了解 Python 语言基本组合数据类型; 理解列表概念并掌握 Python 中的列表使用; 理解字典概念并掌握 Python 中的字典使用; 运用列表管理采集的信息, 构建数据结构; 运用字典处理复杂的数据信息。掌握面向对象程序设计思想概述, 理解类的基本概念, 掌握类的定义和使用, 理解类的继承和多态。		目标 4: 掌握文件的读写方法以及打开和关闭等基本操作; 运用 tkinter 库进行基本的图像处理; 多进程编程和进程的状态, 掌握创建进程与进程枚举和终止等方法, 理解线程的概念, 掌握使用 threading 模块创建线程。掌握网络通信模型和 TCP/IP 协议簇、掌握 Socket 编程。	
	考查点	函数编程	分支结构、循环结构	使用列表、字典管理数据	类的基本使用和对象创建	文件的读写、GUI 编程、网络编程	
	总评分占比	20%	10%	20%	10%	40%	
评分标准	100%	项目主题分明, 数据类型使用合理, 程序流程正确, 格式规范, 代码命名规范。 使用函数来管理代码, 能够正确实现函数的定义和调用, 能够正确使用不同的数据类型来管理数据。 项目要有文件操作功能, 有界面帮助用户更好地使用, 界面友好, 对于数据进行可视化展示。 项目功能完善, 能够解决实践中的关键问题。					
	88%	项目主题分明, 数据类型使用合理, 程序流程正确, 格式规范, 代码命名规范。 使用函数来管理代码, 能够正确实现函数的定义和调用, 能够正确使用不同的数据类型来管理数据。 项目要有文件操作功能, 有界面帮助用户更好地使用, 界面友好, 对于数据进行可视化展示。 项目功能基本完善, 能够解决实践中的一些关键问题。					
	79%	项目有宣明的主题, 数据类型使用基本合理, 程序流程正确, 格式规范, 代码命名规范。 使用函数来管理代码, 能够正确实现函数的定义和调用, 能够正确使用不同的数据类型来管理数据。 项目要有文件操作功能, 有界面帮助用户更好地使用, 对于数据进行可视化展示。 项目功能基本完善, 能够解决实践中的一些问题。					
	69%	项目数据类型使用基本合理, 程序流程基本正确, 格式基本规范, 代码命名基本规范。 使用函数来管理代码, 能够实现函数的定义和调用, 能够使用不同的数据类型来管理数据。 项目要有文件操作功能, 有界面帮助用户更好地使用, 对于数据进行可视化展示。 项目功能基本完善, 能够解决实践中的一些问题。					

对应目标	目标 2: 了解程序的基本结构并绘制流程图; 掌握程序的分支结构, 运用 if 语句实现分支结构; 掌握程序的循环结构, 运用 for 语句和 while 语句实现循环结构。掌握函数的定义和调用方法; 理解函数的参数传递过程以及变量的作用范围; 了解 lambda 函数; 理解函数递归的定义和使用方法。		目标 3: 了解 Python 语言基本组合数据类型; 理解列表概念并掌握 Python 中的列表使用; 理解字典概念并掌握 Python 中的字典使用; 运用列表管理采集的信息, 构建数据结构; 运用字典处理复杂的数据信息。掌握面向对象程序设计思想概述, 理解类的基本概念, 掌握类的定义和使用, 理解类的继承和多态。		目标 4: 掌握文件的读写方法以及打开和关闭等基本操作; 运用 tkinter 库进行基本的图像处理; 多进程编程和进程的状态, 掌握创建进程与进程枚举和终止等方法, 理解线程的概念, 掌握使用 threading 模块创建线程。掌握网络通信模型和 TCP/IP 协议簇、掌握 Socket 编程。
考查点	函数编程	分支结构、循环结构	使用列表、字典管理数据	类的基本使用和对象创建	文件的读写、GUI 编程、网络编程
总评分占比	20%	10%	20%	10%	40%
	%				
59%至0	项目数据类型使用不合理, 格式不规范, 代码命名不规范。 使用函数来管理代码, 能够实现函数的定义和调用, 能够使用不同的数据类型来管理数据。 项目没有文件操作功能, 界面不友好, 数据无可可视化展示。				

2.实验: 必做实验 8 次。支撑目标 2、目标 3、目标 4, 共占总评分 30%, 目标 2 占 10%、目标 3 占 10%、目标 4 占 10%。对应目标的评分标准如下:

对应目标	目标 2: 了解程序的基本结构并绘制流程图; 掌握程序的分支结构, 运用 if 语句实现分支结构; 掌握程序的循环结构, 运用 for 语句和 while 语句实现循环结构。掌握函数的定义和调用方法; 理解函数的参数传递过程以及变量的作用范围; 了解 lambda 函数; 理解函数递归的定义和使用方法。	目标 3: 了解 Python 语言基本组合数据类型; 理解列表概念并掌握 Python 中的列表使用; 理解字典概念并掌握 Python 中的字典使用; 运用列表管理采集的信息, 构建数据结构; 运用字典处理复杂的数据信息。掌握面向对象程序设计思想概述, 理解类的基本概念, 掌握类的定义和使用, 理解类的继承和多态。	目标 4: 掌握文件的读写方法以及打开和关闭等基本操作; 运用 tkinter 库进行基本的图像处理; 多进程编程和进程的状态, 掌握创建进程与进程枚举和终止等方法, 理解线程的概念, 掌握使用 threading 模块创建线程。掌握网络通信模型和 TCP/IP 协议簇、掌握 Socket 编程。
考查点	实验操作, 实验内容, 实验报告	实验操作, 实验内容, 实验报告	实验操作, 实验内容, 实验报告

成绩比例		30%	30%	40%
评分标准	100%至90%	能够根据实验要求,能够把主要功能实现,并且对附加的任务要求也能完成。实验态度认真,操作能力强,操作、记录规范,沟通、协作很好。	实验记录全部完成无遗漏,内容丰富、图文并茂,文字数量足够且正确,实验方案有自己独到的思路与见解。	有很强的总结实验和撰写报告的能力,实验报告内容完整、正确,有很好的分析与见解。文本表述清晰,书写工整,格式规范。
	89%至80%	能够根据实验要求,能够把主要功能实现,并且对附加的任务要求也能完成。实验态度认真,操作能力强,操作、记录规范,沟通、协作很好。	实验记录比较完整,内容比较丰富、图文并茂,文字数量足够且基本正确,实验方案有自己的思路与见解。	有较强的总结实验和撰写报告的能力,实验报告内容完整、正确,有较好的分析与见解。文本表述较为清晰,书写比较工整,格式规范。
	79%至70%	能够根据实验要求,能够把主要功能实现,并且对附加的任务要求也能完成。实验态度认真,操作能力强,操作、记录规范,沟通、协作很好。	实验记录比较完整,内容比较丰富,文字内容足够且基本正确。	有良好的总结实验和撰写报告的能力,实验报告内容较完整、正确,有自己的分析与见解。文本表述较为清晰,书写较为工整,格式较为规范。
	69%至60%	能够根据实验要求,能够把主要功能实现,并且对附加的任务要求也能完成。实验态度认真,操作能力强,操作、记录规范,沟通、协作很好。	实验记录基本完整,内容基本够,有少量错误。	有一定的总结实验和撰写报告的能力,实验报告内容基本完整、正确,没有分析或见解。文本表述基本清晰,书写基本工整,格式基本规范。
	59%至0	动手操作能力差;操作、记录不规范,不能完成实验要求。	实验记录未完成,内容不够,错误多。	总结实验和撰写报告的能力差,实验报告内容不完整、错误多。文本表述不清晰,书写潦草、格式不规范。

五、参考学习资料

推荐教材 1:《Python 程序设计》,耿祥义、张跃平,人民邮电出版社,2017,第 5 版,ISBN: 978711549879

推荐教材 2:《Python 语言程序设计》,袁方 肖胜刚 齐鸿志,清华大学出版社,2019,第 1 版,ISBN: 9787302520290

参考资料 1:《Python 3.7 编程快速入门》,潘中强 薛焱,清华大学出版社,2019,第 1 版,ISBN: 9787302517993

参考资料 2:《Python 程序设计》,董付国,清华大学出版社,2016,第 2 版,ISBN: 9787302436515

参考资料 3:《Python 编程入门经典》, (美) 凯 S.霍斯特曼 (Cay S. Horstmann), 清华大学出版社, 2016, 第 10 版, ISBN: 9787111547426

参考资料 4:《Python 程序设计开发宝典》, 董付国, 清华大学出版社, 2017, 第 1 版, ISBN: 9787302472100

推荐学习网站: 廖雪峰: <https://www.liaoxuefeng.com/wiki/1016959663602400>

菜鸟教程: <https://www.runoob.com/python/python3-tutorial.html>

C 语言中文网: <http://c.biancheng.net/python/>

制订人: 屈治华

审核人: 张倩

(填写说明: 教学大纲由课程负责人制订, 课程所属基层教学组织负责人审核。课程负责人本人是基层教学组织负责人的, 由课程所属学院负责人或学院教学指导委员会相关专业的委员进行审核)