《Python程序设计》教学大纲（理论）

（生物信息学专业）

### 前 言

作为多技术融合发展的产物，计算机技术已经深入到各个学科，与专业教学结合得更加紧密，这使得培养既熟悉专业又能把计算机技术与专业需求紧密结合的复合型人才成为目前高校人才培养的趋势。该课程在内容选取上注重实用性和代表性，突出应用型本科特色；在内容组织上，将相关知识点与实习任务相结合中，让学生通过对任务的分析和实现来掌握相关理论知识，逐步帮助学生构建完整的知识体系。本课程的特点如下：

1.在讲授《Python语言设计》时，通过对教学过程中是否有“培养学生的自学能力、应用能力和创造能力”的反思，采用“基础讲解-任务引导-任务实施-难点解析”的教学模式，并在其中融入计算思维，在注重培养学生实际操作能力的同时，更注重培养学生的信息素养。以理论为依托，以实验为向导，学生综合应用所学知识来解决问题，逐步帮助学生构建起完整的知识体系。

2.突出应用型本科特色，注重计算思维培养。将计算机课程由计算机操作提升到计算思维培养的高度，创建以计算思维、项目实施为核心的教学模式，为学生进行学科研究与创新创业提供有力支撑。该课程以计算思维为导向，以“突出应用”和“强化能力”为目标。

本教学大纲所安排的教学内容是为四年制生物信息学专业的学生而制定。《Python程序设计》课程为专业必修课，总学时数为 48，其中理论课教学的学时数为 24。

### 理论一 Py-1 Python函数调用与内存

1. 目的要求：
2. 了解：函数调用在Python编程中的重要性。
3. 熟悉：Python的运行时内存管理和执行流程。
4. 掌握：Python内置函数、字符串函数及模块提供的函数的使用。
5. 教学内容：
6. 函数调用（Function Calls）：
   1. 讨论Python中的内置函数并演示其使用方法；
   2. 探讨Python中的字符串函数及其在编程任务中的应用；
   3. 解释模块中函数的重要性，并介绍如何利用它们；
   4. 强调Python官方文档的重要性并提供阅读建议。
7. Python内存管理：
   1. 探讨全局空间、堆和调用栈的概念及其重要性；
   2. 解释模块在内存中的表示及其重要性；了解函数如何在内存中存储和操作；
   3. 讨论全局变量的重要性及其与函数定义的关系。
8. 教学学时安排：2学时

教学方法：在机房中强调多媒体与实践相结合。主要采用PPT播放来展示理论内容，辅以板书进行关键点强调。并在关键时刻，切换到程序执行页面，让学生实时观察程序演示，确保理论与实际操作相互融合。

### 理论二 Py-2 字符串

1. 目的要求：
2. 了解：如何从基本概念出发，设计并开发复杂的程序。
3. 熟悉：软件开发中算法设计的基本原则和方法。
4. 掌握：Python中的字符串创建、切割和操作技巧。
5. 教学内容：
6. 字符串基本操作（String Operations）概述。
7. 探讨如何在字符串中包含引号，以及引号在字符串中的角色；
8. 了解字符串切片的功能和用法，将字符串分割成子串；
9. 对比方法与函数的相似性和差异性；学习多参数字符串方法的调用方式。
10. 教学学时安排：2学时
11. 教学方法：

在机房中强调多媒体与实践相结合。主要采用PPT播放来展示理论内容，辅以板书进行关键点强调。并在关键时刻，切换到程序执行页面，让学生实时观察程序演示，确保理论与实际操作相互融合。

### 理论三 Py-3 条件语句和控制流程

1. 目的要求：
2. 了解：编程中的控制结构及其在Python中的应用。
3. 熟悉：Python中条件表达式的作用及其与常规条件语句的区别。
4. 掌握：不同类型的条件语句及其在程序流程中的角色。
5. 教学内容：
6. 程序流程（Program Flow）：讨论代码大多按照书写的顺序执行的情况；探讨有时程序不会严格按照代码的顺序执行的情况及其原因。
7. 条件语句：介绍if语句的结构及其如何使得我们可以选择性地执行代码块；理解if-else结构如何使我们能在两种情况中选择一个执行；深入了解if-elif-else结构，及其如何允许在多个条件中进行选择；了解如何使用简洁的条件表达式来根据布尔结果选择返回值；探讨条件语句如何影响函数内的执行和调用帧的绘制；了解Python的范围概念特性及其对测试代码的影响。
8. 教学学时安排：2学时
9. 教学方法：

在机房中强调多媒体与实践相结合。主要采用PPT播放来展示理论内容，辅以板书进行关键点强调。并在关键时刻，切换到程序执行页面，让学生实时观察程序演示，确保理论与实际操作相互融合。

### 理论四 Py-4 列表和元组

1. 目的要求：
2. 了解：元组是不可变的，而列表是可变的，因此列表更具灵活性。
3. 熟悉：序列、元组和列表是编程知识体系的重要部分。
4. 掌握：处理集合、组织数据、执行复杂操作的基础概念。
5. 教学内容：
6. 元组与列表的表达式：引入并展示元组的概念和与字符串的关系；介绍列表和其与元组的相似性及区别；描述列表的可变性及其赋值方式。
7. 元组与列表的操作：讲解切片对列表的影响；介绍列表和元组的方法；介绍列表作为一种可变对象在函数中的操作和修改。
8. 教学学时安排：2学时
9. 教学方法：

在机房中强调多媒体与实践相结合。主要采用PPT播放来展示理论内容，辅以板书进行关键点强调。并在关键时刻，切换到程序执行页面，让学生实时观察程序演示，确保理论与实际操作相互融合。。

### 理论五 Py-5 迭代和For循环

1. 目的要求：
2. 了解：累加器技巧与基于范围的For循环。
3. 熟悉：For循环的使用及其对可迭代对象的处理。
4. 掌握：迭代的基本概念及其在编程中的应用，For循环中元素位置的遍历与列表的可变性处理。
5. 教学内容：
6. 迭代：描述迭代的定义及在编程中的作用。
7. For-循环初探：引入for循环，并突出其对各种可迭代类型的应用。
8. For-循环进阶：展现如何在函数内部应用for循环。
9. 范围内的For循环：介绍与传统可迭代对象不同的迭代方法。
10. 定位式与可变For循环：解析for循环中元素位置的遍历方式。
11. For循环测试方法：探讨for循环的测试策略。
12. 教学学时安排：2学时
13. 教学方法：

在机房中强调多媒体与实践相结合。主要采用PPT播放来展示理论内容，辅以板书进行关键点强调。并在关键时刻，切换到程序执行页面，让学生实时观察程序演示，确保理论与实际操作相互融合。

### 理论六 Py-6 嵌套循环与嵌套列表

1. 目的要求：
2. 了解：表格数据结构，浅拷贝与深拷贝。
3. 熟悉：使用嵌套循环处理嵌套列表的方法，嵌套切片与拷贝
4. 掌握：嵌套循环在处理多维数据中的重要作用，嵌套列表的结构与应用。
5. 教学内容：
6. 对象列表：描述列表中包含的不同对象。
7. 嵌套列表探究：深入介绍嵌套列表的操作与特性。
8. 表格与嵌套列表：定义表格数据的结构。
9. 嵌套切片与拷贝：探讨切片嵌套列表的方法。
10. 不可变嵌套列表函数：介绍处理嵌套列表数据的不可变函数的技巧。
11. 可变嵌套列表函数：深入探讨处理嵌套列表数据的可变函数的方法。
12. 教学学时安排：2学时
13. 教学方法：

在机房中强调多媒体与实践相结合。主要采用PPT播放来展示理论内容，辅以板书进行关键点强调。并在关键时刻，切换到程序执行页面，让学生实时观察程序演示，确保理论与实际操作相互融合。

### 理论七 Py-7 递归与字典

1. 目的要求：
2. 了解：使用字典与递归解决实际问题的能力。
3. 熟悉：递归的定义、原理及其在编程中的应用。
4. 掌握：Python中字典的基础概念、属性和操作。
5. 教学内容：
6. 递归定义与函数：递归在数学和自然语言中的应用。
7. 递归和调用堆栈：为什么我们应该关心递归的调用堆栈。
8. 字典表达式和赋值：如何在Python中编写和更改字典。
9. 字典可视化：展示字典的文件夹表示方式。
10. 嵌套字典：展示字典的嵌套方式。
11. 字典和For-Loops：在for循环中使用字典的方法。
12. 教学学时安排：2学时
13. 教学方法：

在机房中强调多媒体与实践相结合。主要采用PPT播放来展示理论内容，辅以板书进行关键点强调。并在关键时刻，切换到程序执行页面，让学生实时观察程序演示，确保理论与实际操作相互融合。

### 理论八 Py-8 While循环与文件

1. 目的要求：
2. 了解：While循环在高级应用中的重要性。
3. 熟悉：While循环在文件读取中的基本概念和应用。
4. 掌握：能够区分While循环和For循环的使用场景及其优劣；While循环中的特定模式和调试方法。
5. 教学内容：
6. While循环的基础：介绍While循环的基本概念；讨论While循环的必要性。
7. 跟踪While循环：探索While循环和For循环的功能差异；学习调试方法。
8. While循环与文件读写：在文件读写中应用While循环。
9. While循环与For循环的比较：深入讨论循环的权衡；学习如何选择最合适的循环方式。
10. 教学学时安排：2学时
11. 教学方法：

在机房中强调多媒体与实践相结合。主要采用PPT播放来展示理论内容，辅以板书进行关键点强调。并在关键时刻，切换到程序执行页面，让学生实时观察程序演示，确保理论与实际操作相互融合。

### 理论九 Py-9 对象与类

1. 目的要求：
2. 了解：对象是什么，以及它们如何与类关联。
3. 熟悉：类的主要特性，包括属性、方法和初始化器。
4. 掌握：如何在Python中定义和使用类。
5. 教学内容：
6. 对象介绍：理解对象是什么；探索对象与类的关系。
7. 类的基本知识：探索类的定义；理解类与对象的关系。
8. 属性和方法：学习如何存储对象属性；理解方法与函数的差异。
9. 对象与类的高级概念：了解如何在Python中表示对象。
10. 教学学时安排：2学时
11. 教学方法：

在机房中强调多媒体与实践相结合。主要采用PPT播放来展示理论内容，辅以板书进行关键点强调。并在关键时刻，切换到程序执行页面，让学生实时观察程序演示，确保理论与实际操作相互融合。

### 理论十 Py-10 面向对象设计

1. 目的要求：
2. 了解：如何合理地实现类和方法的重要性。
3. 熟悉：面向对象设计的核心概念，如接口与实现、不变量强制、数据封装。
4. 掌握：如何有效地设计和实现类的接口；处理属性和方法的不变性。
5. 教学内容：
6. 接口与实现：理解类的多个规范定义；学习类接口的重要性。
7. 接口设计：学习如何构建接口；了解类型的重要概念。
8. 接口实现：学习如何编写类的代码；了解属性的复杂性。
9. 不变量强制：了解如何强制前置条件；了解特殊方法的重要性。
10. 数据封装：了解获取器和设置器的重要性；学习如何使用获取器和设置器。
11. 不可变属性：了解获取器和设置器的作用；理解属性的不可变性。
12. 教学学时安排：2学时
13. 教学方法：

在机房中强调多媒体与实践相结合。主要采用PPT播放来展示理论内容，辅以板书进行关键点强调。并在关键时刻，切换到程序执行页面，让学生实时观察程序演示，确保理论与实际操作相互融合。

### 理论十一 Py-11 子类和继承

1. 目的要求：
2. 了解：子类和继承在面向对象编程中的意义。
3. 熟悉：object类的作用；子类层次结构对阅读文档的影响。
4. 掌握：子类的定义方法；使用方法覆盖和super函数修改继承方法。
5. 教学内容：
6. 类和代码重用：了解子类的引入动机；利用子类实现代码的重用。
7. 子类定义：学习子类的定义方法；了解与普通类的异同。
8. object类：理解每个类都是object类的子类；了解object类的特性。
9. 子类层次结构：学习类的“家族树”概念；了解子类层次结构的重要性。
10. 名称解析：理解Python中的点（.）的含义；了解子类的影响。
11. 教学学时安排：2学时
12. 教学方法：

在机房中强调多媒体与实践相结合。主要采用PPT播放来展示理论内容，辅以板书进行关键点强调。并在关键时刻，切换到程序执行页面，让学生实时观察程序演示，确保理论与实际操作相互融合。

### 理论十二 Py-12 抽象和多态

1. 目的要求：
2. 了解：抽象和多态如何增强代码的可维护性和灵活性。
3. 熟悉：抽象在编程中的意义；如何选择合适的抽象级别；动态类型语言中的多态性；抽象和多态的优势。
4. 掌握：面向对象编程中“抽象”与“多态”的核心概念；Python中的多态实现方式以及其在真实场景中的应用；如何在Python中使用抽象基类和方法；能够识别何时并如何在实际编程中应用这些概念。
5. 教学内容：
6. 抽象的基本概念：了解什么是抽象；探讨抽象在编程中的意义；学习如何选择合适的抽象级别。
7. Python中的抽象基类：了解如何定义抽象基类；学习如何声明抽象方法并强制子类实现。
8. 多态的基本概念：理解什么是多态；探讨多态如何简化代码并增加其可扩展性。
9. Python中的多态实现：了解动态类型语言中的多态性；示例：多态在实际编程中的应用。
10. 优缺点和注意事项：了解抽象和多态的优势；学习如何平衡抽象与具体实现。
11. 教学学时安排：2学时
12. 教学方法：

在机房中强调多媒体与实践相结合。主要采用PPT播放来展示理论内容，辅以板书进行关键点强调。并在关键时刻，切换到程序执行页面，让学生实时观察程序演示，确保理论与实际操作相互融合。

**学时分配**

|  |  |
| --- | --- |
| **教学内容（理论）** | **学时** |
| 理论一 Py-1 Python函数调用与内存 | 2 |
| 理论二 Py-2 字符串 | 2 |
| 理论三 Py-3 条件语句和控制流程 | 2 |
| 理论四 Py-4 列表和元组 | 2 |
| 理论五 Py-5 迭代和For循环 | 2 |
| 理论六 Py-6 嵌套列表和嵌套循环 | 2 |
| 理论七 Py-7 递归和字典 | 2 |
| 理论八 Py-8 While循环与文件 | 2 |
| 理论九 Py-9 对象和类 | 2 |
| 理论十 Py-10 面向对象设计 | 2 |
| 理论十一 Py-11 子类和继承 | 2 |
| 理论十二 Py-12 抽象和多态 | 2 |
| **总计** | **24** |

《Python程序设计》教学大纲（实验）

（生物信息学专业）

### 前 言

《Python程序设计》内容将从Python编程的基本概念出发，深入浅出地实习Python的核心编程要素。从Python的开发环境配置开始，我们将逐步深入Python的数据类型、变量、字符串、数字、列表、元组、字典、条件语句、循环语句、函数、类，直至算法设计、内存管理文件操作等诸多方面。通过本课程的学习，学生将全方位掌握Python的基本语法和应用技巧，具备解决实际问题的能力。实验学时为24个。

### 实验一 Py-Lab1 Python函数调用与内存

#### 一、实验目的

（一）掌握熟悉函数调用和可视化方法，这是提高编程效率的关键。

（二）了解模块和脚本之间的差异，为后续编程实践提供基础。

（三）确保在IDE环境下成功安装和运行实验，保证实验环境的稳定。

**二、实验内容**

（一）使用Python Tutor可视化工具，展现函数调用时的内存变化。

（二）编写一个模块，实现骰子的随机投掷功能。

（三）基于给定模块，编写一个脚本，使得可以直接执行并输出结果。

**三、学时安排**

2 学时

#### 四、实验教学方法

在机房中，教师讲解实验内容后，学生独立操作。教师巡视提供辅导，如遇普遍问题，再为全班集体讲解，确保实践与理论相结合。

### 实验二 Py-Lab2 字符串

#### 一、实验目的

（一）掌握和熟悉字符串的基本操作和方法。

（二）了解如何在Python交互式环境中评估字符串表达式。

（三）对字符串相关的Python内置方法有深入的实践和理解。

**二、实验内容**

（一）评估字符串表达式

（二）切割字符串：学习使用Python的字符串切片操作；输入给出的表达式并观察计算结果；根据给出的部分表达式，找出缺失的字面量或函数名，以使表达式得到预期的结果。

（三）调用字符串方法：学习Python中字符串的各种方法，如index、count、isupper、swapcase等。

**三、学时安排**

2 学时

#### 四、实验教学方法

在机房中，教师讲解实验内容后，学生独立操作。教师巡视提供辅导，如遇普遍问题，再为全班集体讲解，确保实践与理论相结合。

### 实验三 Py-Lab3 条件语句和控制流程

#### 一、实验目的

（一）理解和掌握条件语句。

（二）学会使用Python Tutor进行条件语句的可视化。

（三） 学会调试使用条件语句的函数。

**二、实验内容**

(一）可视化If-Else语句：对给定的函数进行逐步可视化，分析全局变量和函数调用的变化。

（二）调试If-Elif语句:观察和记录在全局空间和调用帧中的变化;分析函数的执行过程。

（三）分析函数中哪一行代码未被执行。

**三、学时安排**

2 学时

#### 四、实验教学方法

在机房中，教师讲解实验内容后，学生独立操作。教师巡视提供辅导，如遇普遍问题，再为全班集体讲解，确保实践与理论相结合。

### 实验四 Py-Lab4 列表和元组

#### 一、实验目的

（一）理解和掌握列表和元组的基本概念。

（二）掌握列表和元组的基本操作及方法。

（三）加强实践能力，完成对应的编程任务。

**二、实验内容**

（一）熟悉使用列表。

（二）根据给定的语句，完成列表操作并记录结果。

（三）使用列表方法，编写put\_in函数和replace\_first函数并测试。

**三、学时安排**

2 学时

#### 四、实验教学方法

在机房中，教师讲解实验内容后，学生独立操作。教师巡视提供辅导，如遇普遍问题，再为全班集体讲解，确保实践与理论相结合。

### 实验五 Py-Lab5 迭代和For循环

#### 一、实验目的

（一）加深对迭代和For循环的理解和运用，增强编程能力。

（二）学习如何通过函数实现迭代逻辑。

（三）通过实际操作，掌握如何使用给定的测试工具进行函数验证。

**二、实验内容**

（一）迭代基础：了解For循环的使用方法；学习如何在函数中实现迭代。

（二）函数实现及测试：使用提供的文件进行函数实现。

（三）具体函数实践：lesser\_than 函数（返回列表中严格小于给定值的元素个数）；clamp 函数（返回一个复制的列表，但修改为使每个元素都在min和max之间）。

**三、学时安排**

2 学时

#### 四、实验教学方法

在机房中，教师讲解实验内容后，学生独立操作。教师巡视提供辅导，如遇普遍问题，再为全班集体讲解，确保实践与理论相结合。

### 实验六 Py-Lab6 嵌套列表和嵌套循环

#### 一、实验目的

（一）加深对嵌套列表和嵌套循环的理解，深化编程技能。

（二）掌握如何通过函数实现嵌套列表的处理。

（三）通过实际操作，掌握如何使用给定的测试工具进行函数验证。

**二、实验内容**

（一）理解嵌套列表的结构和特点，学习如何使用For循环进行嵌套列表的处理。

（二）函数实现及测试：使用提供的文件进行函数实现。

（三）具体函数实践:使用row\_sums 函数，返回一个列表，表示输入表格中每行的总和。

**三、学时安排**

2 学时

#### 四、实验教学方法

在机房中，教师讲解实验内容后，学生独立操作。教师巡视提供辅导，如遇普遍问题，再为全班集体讲解，确保实践与理论相结合。

### 实验七 Py-Lab7 递归和字典

#### 一、实验目的

（一）深入理解和实践字典及其应用

（二）掌握递归函数的原理和应用

（三）通过实验加深对于分治策略的理解

**二、实验内容**

（一）字典操作：使用字典数据结构，实现根据分数返回相应成绩的功能，并进行测试。

（二）递归实践：通过递归实现计数功能，即计算一个值在列表中出现的次数。

（三）递归应用：利用递归技术实现列表中的值替换和删除特定值的第一个出现的功能。

**三、学时安排**

2 学时

#### 四、实验教学方法

在机房中，教师讲解实验内容后，学生独立操作。教师巡视提供辅导，如遇普遍问题，再为全班集体讲解，确保实践与理论相结合。

### 实验八 Py-Lab8 while循环与文件

#### 一、实验目的

（一）掌握和熟悉while循环的使用，其中包括对整数范围的循环以及更复杂条件的循环。

（二）培养学生的实践能力，使他们能够对实际代码进行实验并测试。

（三）通过while循环来实现递归函数，并了解何时选择while循环而不是递归。

**二、实验内容**

(一）while循环操作：通过while循环实现连续数据处理，直到满足特定条件停止。

（二）文件操作基础：学习如何打开、读取、写入和关闭文件，以及如何使用while循环读取文件内容，直到文件结束。

（三）文件操作进阶：结合while循环和文件操作，实现数据的读取、处理和存储，如逐行读取文本文件并进行内容分析或修改。

**三、学时安排**

2 学时

#### 四、实验教学方法

在机房中，教师讲解实验内容后，学生独立操作。教师巡视提供辅导，如遇普遍问题，再为全班集体讲解，确保实践与理论相结合。

### 实验九 Py-Lab9 对象和类

#### 一、实验目的

（一）通过理解如何在Python中定义一个类和实例化一个对象，了解类、对象、属性和方法的核心思想。

（二）掌握类的定义、对象的创建和初始化、类和实例变量的使用、以及方法的调用。

**二、实验内容**

（一）类与对象的定义：学习通过类定义来封装共享属性，体现面向对象编程的基本结构和概念。

（二）基本属性与方法：在类中封装了实例和类级别的属性，并定义了方法以实现特定的行为逻辑，展现了面向对象编程中的封装特性。

（三）测试与调试：创建类实例和操作其属性与方法，验证了面向对象编程的实例化过程和方法调用机制。

**三、学时安排**

2 学时

#### 四、实验教学方法

在机房中，教师讲解实验内容后，学生独立操作。教师巡视提供辅导，如遇普遍问题，再为全班集体讲解，确保实践与理论相结合。

### 实验十 Py-Lab10 面向对象设计

#### 一、实验目的

（一） 掌握面向对象编程的基本概念：理解类、对象、属性和方法的核心思想；学习如何在Python中定义和使用类。

（二）掌握如何使用\_\_init\_\_方法来初始化类的实例，这是面向对象编程中类实例化的关键步骤。

（三）了解如何在类中定义和使用构造函数，以及如何通过构造函数给实例变量赋初值。。

#### 二、实验内容

（一） 类与对象的定义：定义一个名为 DNASequence 的类，代表DNA序列。

（二） 基本属性与方法：属性：为 DNASequence 类定义基本属性，如序列字符串 sequence 和序列名称 name。方法：为 DNASequence 类定义方法，如计算GC含量的 gc\_content 方法和获取反向互补序列的 reverse\_complement 方法。

（三） 测试与调试：实例化 DNASequence 类的对象，并调用其方法以确保代码无误。

**三、学时安排**

2 学时

#### 四、实验教学方法

在机房中，教师讲解实验内容后，学生独立操作。教师巡视提供辅导，如遇普遍问题，再为全班集体讲解，确保实践与理论相结合。

### 实验十一 Py-Lab10 子类和继承

#### 一、实验目的

（一） 理解类的继承、多重继承及其在面向对象编程中的应用，学习如何通过继承扩展类的功能。

（二） 学习如何在Python中实现多重继承，理解如何在子类中使用构造函数初始化实例变量，以及如何从父类继承和使用属性和方法。

（三） 掌握通过继承重用父类代码的方法，并在子类中添加或修改特定行为。

#### 二、实验内容

（一）掌握继承机制：深入理解子类如何继承父类的属性和方法，并学习在子类中如何有效地扩展或定制这些属性和方法。

（二）类与对象的定义：设定了基础框架和继承体系，包括单一和多重继承模式，展现类之间的层次关系和属性共享。

（三）基本属性与方法：实现了属性的继承和方法的定制，展示了如何通过继承融合和扩展功能。

（四）测试与调试：通过实例化和操作验证了继承模型的正确性和方法的实用性。

**三、学时安排**

2 学时

#### 四、实验教学方法

在机房中，教师讲解实验内容后，学生独立操作。教师巡视提供辅导，如遇普遍问题，再为全班集体讲解，确保实践与理论相结合。

### 实验十二 Py-Lab12 综合练习

#### 一、实验目的

（一）理解多态与抽象类。

（二）学习并实现基础的序列比对算法。

（三）文件处理与数据解析。

（四）模拟练习。

**二、实验内容**

（一）抽象概念实现：通过定义抽象类和接口，实现具有共通功能的抽象方法。要求创建一个抽象类，其中包含至少一个抽象方法，然后实现该抽象类的具体子类。  
（二）多态性实践：创建父类和多个子类，子类通过覆盖父类方法展示多态性。实践中，应用多态性来调用不同子类的同名方法，展现不同的行为。

（三）模拟练习，题型与考试相同，让学生属性考试环节。

**三、学时安排**

2 学时

#### 四、实验教学方法

在机房中，教师讲解实验内容后，学生独立操作。教师巡视提供辅导，如遇普遍问题，再为全班集体讲解，确保实践与理论相结合。

**学时分配**

|  |  |
| --- | --- |
| **实验内容** | **学时** |
| Py-Lab1 函数调用与内存 | 2 |
| Py-Lab2 字符串 | 2 |
| Py-Lab3 条件语句和控制流程 | 2 |
| Py-Lab4 列表和元组 | 2 |
| Py-Lab5 迭代和For循环 | 2 |
| Py-Lab6 嵌套列表和嵌套循环 | 2 |
| Py-Lab7 递归和字典 | 2 |
| Py-Lab8 While循环与文件 | 2 |
| Py-Lab9 对象和类 | 2 |
| Py-Lab10 面向对象设计 | 2 |
| Py-Lab11 子类和继承 | 2 |
| Py-Lab12 综合练习 | 2 |
| **总计** | **24** |