**广东省高等教育自学考试《计算机高级程序设计》课程考试大纲**

**（课程代码：08074）**

**Ⅰ 课程性质与课程目标**

一、课程性质和特点

“计算机高级程序设计”是为满足计算机应用领域及相关专业对计算机应用人才的需要而开设的。通过本课程的学习，学生能够掌握Python语言程序设计技术，提前适应企业开发的技术要求，为以后从事计算机程序开发工作奠定基础。

“计算机高级程序设计”是一门实践性很强的课程，要求考生不仅要掌握Python语言的基本语法、字符串处理相关知识、文件处理相关知识，而且能够使用面向对象思想设计程序及完成数据分析处理。考生应切实掌握Python基础语法、文件、面向对象、科学计算与可视化等知识，为以后进行程序设计做好知识储备。

二、课程目标

设置本课程的主要目的是使考生掌握计算机程序设计中使用的常用技术，提高解决实际问题的基本能力，为进行计算机程序设计奠定必要的基础。

通过本课程的学习，考生应达到以下目标。

1.掌握Python基本数据类型、表达式及常见内置函数。

2.掌握Python程序控制结构。

3.掌握Python异常处理。

4.掌握组合数据类型的使用。

5.掌握Python函数编写及使用。

6.掌握文件处理方法。

7.掌握面向对象编程方法。

8.掌握使用科学计算与可视化技术进行数据分析处理。

三、与相关课程的联系与区别

学习本课程应具备一定的计算机程序设计基础知识，要求考生已经学习过C语言、数据库编程相关知识。

四、课程的重点和难点

本课程的重点是Python语言基础，包括基础语法、程序结构控制、组合数据类型等。难点是面向对象程序设计、科学计算与可视化。

**Ⅱ 考核目标**

本大纲是“计算机高级程序设计”课程的个人自学、社会助学和考试命题的依据，本课程的考试范围以本考试大纲所限定的内容为准。

本大纲在考核目标中，按照识记、领会、简单应用和综合应用四个层次规定其应达到的能力层次要求。四个能力层次是递进关系，各能力层次的含义是：

识记：要求考生能够识别和记忆“计算机高级程序设计”课程中有关知识点的概念性内容（如教材中给出的基本定义、语法格式、步骤方法、限定规则和特点等），并能够根据考核的不同要求，做出正确的表述、选择和判断。

领会：要求考生在识记的基础上，能够领悟各知识点的内涵和外延，熟悉各知识点之间的区别与联系，能够根据相关知识点的特性来解决不同的问题；并能够进行简单的分析。

简单应用：要求考生运用Python程序设计开发的少量知识点，分析和解决一般的应用问题，例如，分析执行结果、补充程序中空白和编写简单程序等。

综合应用：要求考生综合运用Python程序设计开发的多个知识点，分析解决较复杂的应用问题，并可进行程序设计、分析执行结果、补充程序中的空白和针对应用问题编写程序等。

**Ⅲ 课程内容与考核要求**

第一章 Python概述

一、学习目的与要求

本章对Python进行了简单概述，要求了解程序基本概念、程序的执行方式，熟练掌握开发环境的安装与使用，掌握Python程序的基本语法规则，了解Python中模块的概念，掌握库的安装及使用。

二、课程内容

1. 程序

2. Python简介、环境的安装和使用

3. Python语言的基本语法规则

4. Python模块

三、考核知识点与考核要求

（一） 程序

识记：程序设计语言类型；程序执行方式；

领会：模块的导入和使用

（二）Python简介、环境的安装和使用

识记：IDLE中Python程序运行方式

（三）Python语言的基本语法规则

领会：注释；标识符；关键字；强制缩进；多行语句；同一行执行多条语句

（四）Python模块

识记：模块的概念；扩展库安装

领会：math模块常用成员；random模块常用函数

四、本章重点、难点

本章的重点：Python语言的基本语法规则

本章的难点：Python模块的使用

第二章 Python基本数据类型、表达式和内置函数

一、学习目的与要求

本章简要介绍Python基本数据类型、表达式和内置函数，要求了解基本数据类型、常量、变量的概念，掌握Python中的运算符和表达式的使用，了解内置函数的概念并掌握常用内置函数的使用。

二、课程内容

1. Python数据类型

2. Python运算符与表达式

3. Python常用内置函数

三、考核知识点与考核要求

（一） Python数据类型

识记：组合数据类型的定义；常量与变量的概念

领会：整形、浮点型、复数型、布尔型数据的描述方法；常量及变量的定义与使用

（二）Python运算符与表达式

识记：运算符的优先级和结合性

领会：位运算符、身份运算符的使用

简单应用：算数运算符、关系运算符、逻辑运算符、成员测试运算符的使用

（三）Python常用内置函数

识记：迭代器函数

简单应用：常用数学运算函数、类型转换函数、基本输入输出函数、range函数、len函数、type函数的使用

四、本章重点、难点

本章的重点：数值型数据的描述方法；Python运算符的使用

本章的难点：Python常用内置函数的使用

第三章 程序控制结构

一、学习目的与要求

本章简要介绍程序控制结构，要求了解传统流程图和N-S流程图表示问题求解的方法，掌握单分支、双分支、多分支、嵌套分支结构程序设计，掌握循环结构程序设计，掌握Python中的异常处理方法。

二、课程内容

1. 控制结构概述

2. 选择结构

3. 循环控制结构

4. 异常处理结构

三、考核知识点与考核要求

（一） 控制结构概述

识记：传统流程图描述方法；N-S流程图描述方法

（二）选择结构

综合应用：单分支、双分支、多分支、嵌套分支结构程序设计

（三）循环控制结构

简单运用：break语句、continue语句、else语句的使用

综合应用：运用while循环语句、for循环语句进行简单循环结构及多重循环结构程序设计

（四）异常处理结构

识记：异常的分类

简单应用：运用try...except语句、try...except...else语句、try...except...else...finally语句进行异常处理程序设计

四、本章重点、难点

本章的重点：选择结构程序设计、循环结构程序设计

本章的难点：异常处理结构

第四章 组合数据类型

一、学习目的与要求

本章简要介绍Python中的各种组合数据类型，了解组合数据类型的概念，掌握列表、元组、集合、字典类型的创建及访问方法，掌握列表、元组、集合、字典类型的运用。

二、课程内容

1. 组合数据类型概述

2. 列表类型

3. 元组类型

4. 集合类型

5. 字典类型

三、考核知识点与考核要求

（一） 组合数据类型概述

识记：组合类型数据分类

（二）列表类型

领会：列表的创建、列表的访问、列表的切片、列表的操作

综合应用：运用列表解决实际问题

（三）元组类型

领会：元组的创建、元组的访问、元组的切片、元组的操作

综合应用：运用元组解决实际问题

（四）集合类型

领会：集合的创建、集合的访问、集合的操作

综合应用：运用集合解决实际问题

（五）字典类型

领会：字典的创建、字典的访问、字典的操作

综合应用：运用字典解决实际问题

四、本章重点、难点

本章的重点：列表、元组、集合、字典的创建、访问及操作方法

本章的难点：列表、元组、集合、字典的运用

第五章 函数

一、学习目的与要求

本章简要介绍Python中的函数知识，要求掌握函数的定义和调用方法，掌握函数的参数传递，掌握变量作用域，了解lambda函数。

二、课程内容

1. 函数的定义与调用

2. 参数传递

3. 变量作用域

4. lambda表达式函数

三、考核知识点与考核要求

（一） 函数的定义与调用

识记：函数的概念；函数的作用

领会：函数的定义方法；函数的调用及返回方法；函数的嵌套调用和递归调用；

简单应用：运用递归函数解决问题

综合运用：运用函数进行模块化程序设计

（二）参数传递

识记：函数的形参、实参概念

领会：函数参数传递方法；可变长形参

简单应用：实参解包

（三）变量作用域

识记：变量作用域类型

领会：全局变量作用域和使用；局部变量的作用域

（四）lambda表达式函数

领会：lambda表达式函数的定义和调用

四、本章重点、难点

本章的重点：函数的定义及调用；参数的传递

本章的难点：运用函数进行模块化程序设计

第六章 字符串和正则表达式

一、学习目的与要求

本章简要介绍字符串数据类型，要求了解字符串编码和解码，掌握字符串运算，掌握字符串处理内置函数，掌握str类对象及相关处理方法。

二、课程内容

1. 字符串概述

2. 字符串处理的函数和方法

三、考核知识点与考核要求

（一） 字符串概述

识记：常见转义字符的含义

领会：字符串的表示

简单应用：字符串的运算

（二）字符串处理的函数和方法

领会：字符串处理的常用内置函数

简单应用：字符串对象的拆分、连接方法；字符串对象的查找、替换方法；测试串的方法；格式化数据的方法；大小写转换和字符映射的方法

四、本章重点、难点

本章的重点：字符串的运算；字符串处理常用内置函数

本章的难点：字符串对象的使用方法

第七章 文件

一、学习目的与要求

本章简要介绍Python中对文件的操作方法，要求了解文件的概念和分类，掌握文件的打开、关闭、读写、定位的方法，掌握数据维度的概念。

二、课程内容

1. 文件基本处理

2. 数据维度

三、考核知识点与考核要求

（一） 文件基本处理

识记：文件类型；文件基本处理流程

领会：文件打开模式

综合运用：文件的打开方法；文件关闭方法；文件的读写方法

（二）数据维度

识记：一维数据；二维数据；高维数据

领会：CSV文件格式；JSON格式

四、本章重点、难点

本章的重点：文件处理流程；文件打开模式

本章的难点：文件读写方法

第八章 Python面向对象编程

一、学习目的与要求

本章简要介绍Python面向对象编程方法，要求了解面向对象编程方法特点，了解对象型数据的特点和成员构成，掌握类和实例的关系，掌握类和实例的创建方法，了解面向对象的继承，了解面向对象的多态。

二、课程内容

1. 面向对象概述

2. 类对象与实例对象

3. 对象成员

4. 继承和多态

三、考核知识点与考核要求

（一） 面向对象概述

识记：面向过程；面向对象；对象；类；封装；继承；多态

类对象与实例对象

识记：对象的结构

领会：类的定义方法；类的构造方法；实例的创建方法；实例的访问方法

（二）对象成员

识记：对象的私有方法与公有方法

领会：类方法；静态方法

简单应用：类的属性成员定义；实例属性成员访问；实例方法

（三）继承和多态

识记：继承的概念；多态的概念；常用的重载方法

领会：继承的定义；方法重载

四、本章重点、难点

本章的重点：面向对象概念；类的定义；

本章的难点：类方法；静态方法

第九章 科学计算与可视化

一、学习目的与要求

本章简要介绍科学计算与可视化技术，要求了解numpy数组概念，掌握numpy数组的创建方法、运算类型、数组访问以及常用成员方法，了解pandas中Series和DataFrame两种数据类型，掌握pandas两种数据类型的创建和访问方式，了解matplotlib绘图方法。

二、课程内容

1. 数值计算模块

2. 数据分析模块

3. 数据可视化模块

三、考核知识点与考核要求

（一） 数值计算模块

识记：ndarray类型数组基本属性

领会：数组运算；数组方法

简单应用：numpy常用数组创建函数；访问数组元素

（二）数据分析模块

识记：Series数据结构；DataFrame数据结构

领会：DataFrame统计；DataFrame排序

简单应用：Series数据创建方法；Series访问；DataFrame数据创建方法；DataFrame访问

（三）数据可视化模块

识记：图像制作过程步骤；

领会：折线图绘制；柱形图绘制；散点图绘制；

四、本章重点、难点

本章的重点：DataFrame数据创建及访问

本章的难点：可视化图形绘制

**Ⅳ 关于大纲的说明与考核实施要求**

一、自学考试大纲的目的和作用

课程自学考试大纲是根据专业自学考试计划的要求，结合自学考试的特点而确定。其目的是对个人自学、社会助学和课程考试命题进行指导和规定。

课程自学考试大纲明确了课程学习的内容以及深广度，规定了课程自学考试的范围和标准。因此，它是编写自学考试教材和辅导书的依据，是社会助学组织进行自学辅导的依据，是自学者学习教材、掌握课程内容知识范围和程度的依据，也是进行自学考试命题的依据。

二、课程自学考试大纲与教材的关系

课程自学考试大纲是进行学习和考核的依据，教材给出了学习掌握课程知识的基本内容与范围，教材的内容还包括大纲所规定的课程知识的扩展与发挥。课程内容在教材中可以体现一定的深度或难度，但在大纲中对考核的要求一定要适当。

大纲与教材所体现的课程内容基本一致；大纲里面的课程内容和考核知识点，教材里一般也要有。反过来教材里有的内容，大纲里就不一定体现。

三、关于自学教材

《Python语言程序设计基础教程》，傅清平、李雪斌、徐文胜编著，清华大学出版社出版，2022年第1版。

教材中第5章5.1.4、5.3.3，第6章6.3和第7章7.3内容不纳入考核范围，考生可根据个人兴趣学习。

四、关于自学要求和自学方法的指导

本大纲的课程基本要求是依据专业考试计划和专业培养目标而确定的。课程基本要求还明确了课程的基本内容，以及对基本内容掌握的程度。基本要求中的知识点构成了课程内容的主体部分。因此，课程基本内容掌握程度、课程考核知识点是高等教育自学考试考核的主要内容。

为有效地指导个人自学和社会助学，本大纲已指明了课程的重点和难点，在章节的基本要求中一般也指明了章节内容的重点和难点。

本课程共4学分。由于成人学习的个性化特点，建议业余自学时间不低于64个学时。

建议学习本课程时注意以下几点：

1.在学习本课程教材之前，应先仔细阅读本大纲，了解本课程的性质和特点，熟知本课程的基本要求，在学习本课程时，能紧紧围绕本课程的基本要求。

2.在自学每一章的教材之前，先阅读本大纲中对应章节的学习目的与要求、考核知识点与考核要求，以使在自学时做到心中有数。

3.学习计算机高级程序设计的目的是用Python相关技术解决实际问题，程序设计能力的培养除要学习课程书本知识之外，上机实践是学习程序设计最有效的途径，为此，要求考生加强上机实践，提升编程能力，能在计算机上解答教材中的习题。

4.本课程在学习过程中可以结合一个具体项目需求，利用软件工程相关知识，通过本课程学到的内容逐步实践。

建议学时如下：

|  |  |  |
| --- | --- | --- |
| 章次 | 学习内容 | 建议学时 |
| 第1章 | Python概述 | 4 |
| 第2章 | Python基本数据类型、表达式和内置函数 | 8 |
| 第3章 | 程序控制结构 | 8 |
| 第4章 | 组合数据类型 | 8 |
| 第5章 | 函数 | 6 |
| 第6章 | 字符串和正则表达式 | 6 |
| 第7章 | 文件 | 8 |
| 第8章 | Python面向对象编程 | 8 |
| 第9章 | 科学计算与可视化 | 8 |

六、对考核内容的说明

1. 本课程要求考生学习和掌握的知识点内容都作为考核的内容。课程中各章的内容均由若干知识点组成，在自学考试中成为考核知识点。因此，课程自学考试大纲中所规定的考试内容是以分解为考核知识点的方式给出的。由于各知识点在课程中的地位、作用以及知识自身的特点不同，自学考试将对各知识点分别按四个认知层次确定其考核要求。

2. 在考试之日起6个月前，由全国人民代表大会和国务院颁布或修订的法律、法规都将列入相应课程的考试范围。凡大纲、教材内容与现行法律、法规不符的，应以现行法律法规为准。命题时也会对我国经济建设和科技文化发展的重大方针政策的变化予以体现。

七、关于考试命题的若干规定

1.本课程考试采用闭卷笔试形式,考试时间为150分钟；满分100分,60分及格。

2.本大纲各章所规定的基本要求、知识点及知识点下的知识细目，都属于考核的内容。考试命题既要覆盖到章，又要避免面面俱到。要注意突出课程的重点、章节重点，加大重点内容的覆盖度。

3.不应命制超出大纲中考核知识点范围的题目，考核目标不得高于大纲中所规定的相应的最高能力层次要求。命题应着重考核自学者对基本概念、基本知识和基本理论是否了解或掌握，对基本方法是否会用或熟练。不应命制与基本要求不符的偏题或怪题。

4.本课程在试卷中对不同能力层次要求的分数比例大致为：识记占20%，领会占30%，简单应用占30%，综合应用占20%。

5.试题的难易程度分为4个等级：易、较易、较难和难四个等级。每份试卷中不同难度试题的分数比例一般为：易占20%，较易占30%，较难占30%，难占20%。

必须注意试题的难易程度与能力层次有一定的联系，但两者不是等同的概念，在各个能力层次都有不同难度的试题。

6.各种题型的具体样式参见本大纲附录。

**附录 题型举例**

一、单项选择题

下列选项中正确的变量名是( )

A.2sum B.if

C.sum2 D.a\*b

二、简答题

1.请描述面向对象和面向过程的区别。

三、简单设计题

1. 如果一个数恰好等于它的因子之和，这个数就称为“完数”。例如28=1+2+4+7+14则28就是一个完数。编写程序，输出1000以内的所有完数个数。

四、综合设计题

编写程序，按行读取数据文件中的字符串，输出每一行字符串中大写字母的个数。

要求：

数据文件名为data.txt。

使用相对路径读取文件，数据文件与源代码文件在同一文件夹。

求字符串中大写字母个数以函数方式实现。