**广东省高等教育自学考试《程序设计入门—Java》课程考试大纲**

**（课程代码：08553）**

**Ⅰ 课程性质与课程目标**

**一、课程性质和特点**

该课程是以java程序设计为例进行编程教学的一门课程。Java语言是计算机相关专业人员应当掌握的被广泛使用的一门编程语言，是众多计算机相关领域必掌握的基础技术。

通过该课程的学习，使学生获得基本全面的Java程序设计知识和面向对象编程思想，掌握基于Java开发软件的科学方法，提高学生面向对象开发能力、软件系统设计能力，使学生能够运用 Java 语言作为一种思维工具解决处理现实问题，为学习后续课程、以及将来从事软件设计与开发打下扎实的基础，以适应当代社会对软件开发与管理的需求。

**二、课程目标**

1. 熟练掌握java编程基本语法，能熟练运用数据类型、表达式、程序流程控制等java语法进行编码；
2. 理解面向对象编程思想，理解类的封装、继承、多态性，理解类与对象关系，能熟练定义类及对象的创建和使用；熟练掌握接口、抽象类的设计与使用，熟悉异常处理机制。
3. 了解java.lang包中的基础类库以及java.util包中的工具类库，包括字符串类、数学类、日期时间类、数据类型包装类等；
4. 掌握利用集合框架接口进行数据处理，包括Set、List、Map接口等。
5. 通过综合案例的分析、设计与开发，学习应用程序设计语言解决和处理实际问题的思维方法。

**三、与相关课程的联系与区别**

程序设计入门、移动应用开发、软件开发工具等课程密切相关。程序设计课程以java语言为例进行教学，学习数据类型、表达式、流程控制、数组等基本语法的使用，掌握面对对象编程方法以及综合软件分析、设计与开发实践，使学生具备以程序开发解决实际问题能力，是其他几门课的先修课程。

移动应用开发课程以android开发为例进行授课，android应用开发针对手机、智能手表等移动设备进行应用开发，涉及界面设计和代码编写，前端采用xml语言，但一般采用可视化开发，代码编写主要采用java语言。

软件开发工具课程主要进行动态web应用系统开发，涉及web开发前后端开发的相关关键技术，主流web开发后端主要采用java技术，必须具备java程序设计基础，掌握面向对象编程方法。

所以程序设计入门（java）课程是移动应用开发、软件开发工具课程的先修课程。

**四、课程的重点和难点**

本课程的重点难点如下：

1. **课程重点**

本课程重点是培养学生的基本编程能力，理解和掌握面向对象编程思想和方法，能够利用java语言进行程序设计以解决实际问题。重点内容包括：

1. java基本语法：数据类型、表达式、运算符、表达式、流程控制、数组。
2. 面向对象编程思想及实现：类、对象、抽象类、接口、封装、继承和多态；
3. **课程难点**

本课程难点是循环语句编程以及面向对象高级应用，包括：

1. 运用数组、集合、流程控制等语法实现特定算法的编程技巧。
2. 类的封装、继承和多态内涵及实现；静态方法与实例方法差异；类、抽象类与接口三者区别与关系。

**Ⅱ 考核目标**

本大纲在考核目标中，按照识记、领会、应用三个层次规定其应达到的能力层次要求。三个能力层次是递进关系，各能力层次的含义是：

**识记**：要求考生记忆并理解大纲各章中知识点，主要包括数据类型、运算符与表达式、流程控制语句、数组、集合等基本语法的定义；类与对象的概念与关系、接口和抽象类的作用与异同；异常处理作用与机制；三种集合接口概念及关系。

**领会：**指考生能够领悟和理解大纲知识点内涵和外延，涉及的概念及其之间的关系，并能正确使用大纲知识点完成验证性实验。本课程要求能在开发环境下对相关知识点进行编码实现。主要包括能够正确运用表达式运算、流程控制语句、数组、集合、常用API类等进行算法实现；能够根据类的描述完成类的设计与实例化，理解类、抽象类、接口关系及应用场景并能完成相应设计，能通过继承、实现接口等方法实现类的继承、多态性。

**应用：**指考生能够运用大纲知识点解决实际应用问题**。**本课程指能够使用java程序语言解决实际问题，针对特定任务，通过编码实现相应功能。能实现算法编写，比如写一个程序，统计一个班级学生英语成绩的均值、按成绩排序等；能运用面向对象思想实现应用场景的类的设计，比如设计一个学生类，能够描述学生的专业、年级、学院、性别等基本信息，设计相应方法能够实现信息修改等相关功能等。

**Ⅲ 课程内容与考核要求**

第一章 初识Java

**一、学习目的与要求**

了解java语言特点，熟悉java开发环境设置，掌握java程序编写基本流程。

**二、考核知识点与考核要求**

1. java语言发展

识记：计算机语言分类；java发展史与特点；了解java体系。

2. java开发环境

识记：jdk概念。

领会：jdk主要目录结构。

应用：jdk安装；classpath、path环境变量配置。

3. java程序编写

识记：java类结构；main方法。

领会：类的概念、main方法作用；java程序编写原则；。

应用：类文件编写方法；编译命令javac；运行java类命令 java。

4. eclipse使用

识记：eclispe概念。

领会：eclispe使用方法。

应用：eclispe安装；eclipse环境下java项目创建；类的创建与运行。

**三、本章重点、难点**

1.本章重点

jdk安装；java程序编写与运行基本过程；eclipse环境下java项目创建；

2.本章难点

classpath、path环境变量配置；javac、java编译与运行命令的使用；

第二章 java基本语法

**一、学习目的与要求**

掌握java语言语法知识，包括数据类型、变量常量、运算符表达式，能够完成基本的java程序编写，实现输入输出及表达式运算。

**二、考核知识点与考核要求**

1. java关键字与标识符

识记：java主要关键字；java标识符语法规范；标识符命名风格。

2. 常量与变量

识记：常量的表示；变量的概念。

领会：变量的作用。

应用：声明常量；字面值常量；变量的定义。

3. 基本数据类型

识记：数据类型作用；数据类型分类与表示。整型：byte、short、int、long；浮点型：float、double；字符型：char；布尔型：blooean。

领会：不同数据类型常量值的表示；不同数据类型变量定义与赋值。

应用：在程序中正确定义变量及赋值。

4. 数据输入与输出

识记：System.out类、Scanner类作用。

领会：System.out类、Scanner类使用方法。

应用：使用System.out类输出结果；使用Scanner类及对应输入方法从控制台获取输入值。

5. 运算符与表达式

识记：各类运算符表示：算术运算符（+、-、\*、/、%、--、++）；赋值运算符（+=、-=、\*=、\=、%=）；关系运算符（>、<、>=、<=、==、!=）;逻辑运算符（&&、||、|、&、|）；条件运算符（条件？=表达式1：表达式2）

领会：不同运算符作用；运算符优先级；运用运算符构成表达式。

应用：综合运用各类运算符构成表达式，完成特定任务运算。

6. 数据类型转换

识记：数据类型转换概念

领会：两种类型转换方式；表达式中不同数据类型自动转换规则。

应用：在程序中实现数据类型强制转换。

**三、本章重点、难点**

1.本章重点

不同数据类型变量的定义与赋值；各类运算符的使用；输入输出方法。

2.本章难点

Scaner输入方法；综合运用各类运算符编写表达式完成特定计算。

第三章 Java 程序的流程控制

**一、学习目的与要求**

理解三种程序结构作用，熟悉分支、循环结构相关语法，掌握分支、循环代码的实现。

**二、考核知识点与考核要求**

1. 三种程序结构

识记：三种程序结构概念：顺序结构、分支结构、循环结构。

领会：三种程序结构作用及应用场景。

2. 分支结构

识记：分支语法结构，包括if；if-else；if-else多分支；if嵌套；switch多分支。

领会：分支语法使用方法；不同分支语法差异。

应用：能够针对不同需求采用不同分支语法实现分支结构程序代码编写。

3. 循环结构

识记：几种循环语法结构：while；do…while；for；continues；break。

领会：不同循环语句差异。

应用：能够采用相关循环语法编写代码，解决特定现实问题。

4. 方法

识记：方法的概念、结构与作用。

领会：掌握定义方法的三个要素，返回值、方法名、参数，会定义方法

应用：能根据实际需求完成方法定义和调用；了解递归实现。

**三、本章重点、难点**

1.本章重点

表达式；if语句；while和for循环语句；方法的定义和调用；

2.本章难点

循环语句嵌套；循环跳转语句。

第四章 数组

**一、学习目的与要求**

掌握一维和二维数组的定义、初始化及应用；掌握数组的查找、修改方法；了解数组相关的算法应用。

**二、考核知识点与考核要求**

1. 一维数组

识记：数组的概念；数组的定义方法。

领会：数组的作用；数组应用场景；数组的相关操作。

应用：定义数组；数组初始化；数组查找；数组修改；数组排序；Arrays类的使用。

2. 二维数组

识记：二维数组概念。

领会：二维数组作用、应用场景及相关操作。

应用：二维数组定义；二维数组初始化；二维数组遍历。

**三、本章重点、难点**

1.本章重点

一维数组定义；一维数组初始化；一维数组遍历；二数组定义；二维数组初始化；二维数组遍历。

2.本章难点

一维数组元素的移动；一维数组元素排序；二维数组遍历及应用。

第五章 面向对象基础

**一、学习目的与要求**

熟悉面向对象三个特征，掌握类的定义、对象的创建与使用，掌握类的方法构，掌握继承、方法重写及多态的应用，理解this、super、static等关键字作用。

**二、考核知识点与考核要求**

1. 面向对象特征

识记：三个特征:封装性、多态性、继承性。

领会：三个特征含义及作用。

2. 类与对象

识记：类的结构及相关关键字表示，class、new、构造方法、this、static等概念；访问权限。

领会：类的设计方法、对象的创建和使用。

应用：能根据应用场景完成类设计与使用，包括：定义类；创建对象；使用对象；this的使用；static定义静态变量、静态方法与静态代码块；使用权限声明变量和方法。

3. 继承与多态

识记：继承与多态的概念及应用场景；extends继承、super等关键字。

领会：子类的设计方法；向上转型。

应用：能根据特定场景进行子类设计，包括：使用extends定义子类；使用super调用父类方法和成员变量；方法重写（静态多态）；方法覆盖（动态多态）；instanceof运算符的使用方法。

**三、本章重点、难点**

1.本章重点

类的定义；对象的创建与使用；子类的定义。

2.本章难点

static静态变量静态方法的声明与实例方法的区别；方法重载及多态的理解；向上转型应用。

第六章 面向对象高级

**一、学习目的与要求**

掌握抽象类的定义与使用；掌握接口的定义与实现；掌握异常处理机制。

**二、考核知识点与考核要求**

1. 抽象类

识记：抽象类的概念、作用及应用场景；abstract抽象关键字。

领会：抽象类及抽象方法的定义。

应用：能根据特定场景设计抽象类及抽象方法；通过继承使用抽象类。

2. 接口

识记：接口的作用及应用场景；关键字interface、implements作用。

领会：接口的定义和实现方法。

应用：能根据特定场景进行接口的定义和实现。

3. 异常处理

识记：异常处理概念；异常处理的作用及应用场景；几种异常处理方法。

领会：try…catch…finally、throw、throws异常处理语法应用。

应用：能够采用try…catch…finally、throw、throws等对特定场景代码进行异常处理。

**三、本章重点、难点**

1.本章重点

抽象类的定义与使用；接口的定义与使用；try…catch异常处理；throws声明异常抛出。

2.本章难点

接口与抽象类的异同；throw与throws差异。

第七章 常用java API

**一、学习目的与要求**

掌握Object类的常用方法、字符串类的使用、随机数的产生及日期时间类应用，了解基本数据类型包装类。

**二、考核知识点与考核要求**

1. Object类

识记：Object类概念及作用； Object常用方法。

领会：toString（）、equals（）两个方法的作用。

应用：能够通过覆盖toString（）实现返回特定信息；能够通过覆盖equals方法实现按特定方式进行对象相等比较。

2. 字符串类

识记：字符串类概念； String类与StringBuffer类的差异。

领会：能够正确使用String类与StringBuffer类定义变量。

应用：能够正确使用String类与StringBuffer类定义变量；能够调用字符串类常用方法进行字符串操作。

3. 随机数

识记：Math类的random方法作用；Random类作用。

领会：Math和Random产生随机数的异同及不同的应用场景。

应用：能用Math类生成随机数；能用Random类生成随机数。

4. 基本类型包装类

识记：boolean、byte、char、int、short、long、float、double八种基本类型对应的包装类。

领会：基本类型对应的包装类的及不同的作用及应用场景；拆箱装箱操作。

应用：能用包装类进行数据类型转换。

5. 日期时间类

识记：日期时间类的作用；Date和Calendar类的作用。

领会：能用Date和Calendar类生成日期对象。

应用：能根据特定场景要求，正确使用Date和Calendar类生成日期对象。

**三、本章重点、难点**

1.本章重点

Object类的toString（）和equals（）的重写；String类的常用字符串操作方法使用。

2.本章难点

Object类的equals（）重写；String类的操作。

第八章 集合框架

**一、学习目的与要求**

了解java集合框架体系，掌握List、Set、Map接口的特点及应用；掌握Collections类对集合的常用操作；掌握泛型的应用。

**二、考核知识点与考核要求**

1. 集合框架

识记：集合框架体系；接口之间相互关系。

2. Collection接口

识记：Collection接口常用方法。

3. List

识记：List接口集合作用；List与ArrayList关系；泛型、Iterator接口、foreach及应用场景。

领会：能够定义List集合变量并进行增删改等操作，能够通过Iterator接口和foreach对集合进行遍历访问。

应用：能针对特定需求，利用List集合实现相关数据的存储与运算。

4. Set

识记：Set作用，TreeSet、HashSet异同。

领会：能够使用HashSet定义集合，以及集合元素的修改与访问。

应用：能针对特定需求，利用Set集合实现相关数据的存储与运算，解决实际问题。

5. Map接口

识记：Map集合概念及作用及主要方法。

领会：能使用HashMap定义集合变量并能够进行数据操作和遍历。

应用：能根据特定场景要求，正确使用HashMap类定义集合遍历，按任务需求实现相关功能。

6. Collections类

识记：Collections类的sort方法；Comparable、Comparator接口的作用。

领会：能用Collections类的sort方法对集合进行排序，能够使用Comparable实现类的可排序，能用Comparator接口设计比较器。

应用：能根据特定场景要求，正确使用Collections类、Comparable接口、Comparator接口对特定数据进行排序，完成实际任务。

**三、本章重点、难点**

1.本章重点

采用ArrayList、Map、Set定义集合对象，对集合元素进行增删改、遍历、排序等操作。

2.本章难点

泛型的理解和使用；Comparable、Comparator接口的使用。

**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，教材是学习掌握课程知识的基本内容与范围，教材的内容是大纲所规定的课程知识和内容的扩展与发挥。

**三、关于自学教材**

《java程序设计案例教程》，胡运玲，王海燕主编，人民邮电出版社，2022年第1版。

本教材第9章适合作为课程设计的参考样例，不纳入本大纲考核范围，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。

**（一）多动手进行验证性实验，理论与实践相结合,深入理解知识点**

Java程序设计是一门动手编程能力培养课程，自学者应理论和实践相结合学习本门课程。搭建好实验环境，多动手进行编码实验，在学习过程中参照教程中语法代码、案例，在电脑上调试运行，通过理论介绍和实验结果对比，能更深入理解相关语法含义，通过理解能更容易记忆相关知识点。

**（二）多做习题，提高编码能力**

多做课后编程题，利用所学语法知识，通过数据建模、程序流程分析、代码实现等过程，解决某个特定小问题，能够提高利用程序语言解决实际问题能力。也可根据个人兴趣或个人现实中的实际问题利用程序来解决，例如设计一个程序计算成绩绩点、学生成绩排序成绩等。通过大量的编码训练，编码能力将会大幅度提升，对应的理论知识也自然就更能深入理解。

**（三）基于项目驱动方法学习，提升软件开发综合能力**

以团队或个人基于某个特定应用场景，开发一个java应用项目，解决特定实际问题。可以参考综合案例，按照系统分析、功能设计、系统实现等软件开发步骤，开发一个小型应用系统。通过项目的设计与开发，能够提高综合分析、设计、编码以及创新应用能力，能够更全面训练java程序设计思维能力。

**五、对社会助学的要求**

**（一）编程开发入门指导**

软件技术基础课程以java编程为主，自学者往往在刚入门时在开发环境、编码过程、程序编写规范、运行调试等方面遇到各种问题。助学者可就以上几个方面进行指导，帮助自学者能够独立的进行完整的编码实验，并给出编程中遇到问题如何解决的自主学习方法建议。

**（二）注意培养自学者自主学习能力**

程序编写总会遇到各种问题，包括语法错误、逻辑错误、环境错误等各种问题。助学者应在指导过程中注重培养学生自主学习解决问题的能力。首先指导学生学会调试程序，看懂程序报错信息，根据报错信息去互联网搜索解决问题办法；其次，培养学生养成良好的编码习惯，学会通过集成工具查看各种帮助信息；最后要学会充分利用各种网络学习资源培养自主学习能力，善于利用人工智能进行辅助学习，例如利用大语言模型文心一言进行java学习规划、答疑等。

**（三）帮助自学者对重点难点的辅导**

Java面向对象编程是课程重点，同时也是难点，助学者应帮助自学者理解面向对象编程思想，深入理解类与对象的概念及其关系，掌握类的设计方法，包括成员变量、构造方法、实例方法、静态方法的设计，对象的创建与调用，深入理解继承、多态等概念及其实现。助学者应结合案例进行这些知识结构的讲解。

**（四）建议每学分2-3个助学学时。**

**六、对考核内容的说明**

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三个认知层次确定其考核要求。

**七、关于考试命题的若干规定**

1、本课程考试采用闭卷笔试方式考核，考试时间150分钟，按百分制记分,60 分为及格。

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、命题不应有超出大纲中考核知识点范围的题，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。不应出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30%，领会占40%，应用占30%。

5、要合理安排试题的难易程度，试题的难度可分为：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：2:3:3:2。

6、课程考试命题的主要题型一般有单项选择题、填空题、简答题、程序分析题、程序设计题等题型。

**附录 题型举例**

1. **单项选择题**
2. 一个类中可以定义多个名字相同，但参数不同的方法，这叫方法的（ ）
3. 继承 B.覆盖 C.改写 D.重载
4. **填空题**
5. 面向对象编程的三个特点是\_\_\_\_\_\_\_\_、继承、\_\_\_\_\_\_\_\_。
6. **程序分析题**

1.阅读程序，回答以下问题

abstract class Food {

String name;

String type;

public Food(String name, String type) {

super();

this.name = name;

this.type = type;

}

public abstract String HowToCooking();

}

interface Edible {

String HowToEat();

}

class Pork extends Food implements Edible{

public Pork(String name, String type) {

super(name,type);

}

@Override

public String HowToEat() {

return "烹饪后食用";

}

@Override

public String HowToCooking() {

return "煎炸煮炒";

}

}

public class TestFood{

public static void main(String[] args) {

Pork pork=new Pork("猪肉","肉类");

System.out.println(pork.HowToEat()) ;

System.out.println(pork.HowToCooking()) ;

}

}

1. 写出运行结果
2. 说明程序中各个类或接口之间的关系
3. **程序设计题**

按如下要求设计类

1. 设计一个耳机类，类名Earphone
2. 定义成员变量：产地、价格，类型。（属性名自行命名）
3. 定义构造方法：包括三个参数，分别初始化三个成员变量。
4. 定义一个降价方法：参数为降价量，无返回值，实现价格降价。（方法名自定）
5. 编写main()方法, 实例化一个Earphone对象，参数值自定，调用对象的降价方法，参数值自定。