《面向对象程序设计》教学大纲

课程编号：072114A

课程类型：□通识教育必修课 □通识教育选修课

☑学科基础课 □专业核心课

□专业提升课 □专业拓展课

总学时：64 讲课学时：32 实验（上机）学时：32

学　　分：4

考试类型：☑考试 □考查

适用对象：计算机科学与技术

□是 ☑否 适合作为其他专业学生的个性化选修课

先修课程：程序设计基础与应用

一、教学目标

《面向对象程序设计》是计算机科学与技术专业本科生的学科基础课。设置本课程的目的是使学生理解面向对象程序设计的思想和特征，掌握Java语言程序设计的各项技术：

1.建立面向对象的思维方式；领会面向对象程序设计的风格和特征。

2.掌握封装、继承、多态性等面向对象程序设计的各项技术。

3.掌握字符串、数组、例外处理、线程与同步等Java程序设计的基础技术。

通过本课程的学习，可以为学生进一步学习各种Java专项技术、构建面向对象软件系统以及网上应用系统奠定相应的程序设计基础。

课程思政教学目标：培养严谨态度，崇尚理性精神，提升逻辑思维能力，养正民族认同感和国家情怀。

二、教学内容及其与毕业要求的对应关系

1.教学内容

本课程的内容可分为三部分：

(1) Java语言基础知识，包括Java程序的基本结构、Java程序开发环境、数据与数据运算以及Java语句等。

(2) 面向对象编程核心内容，包括类和对象的概念、类的定义、继承、接口和包等。

(3) Java编程技能，包括字符串和数组、输入和输出、例外处理以及线程与同步等。

2.教学方法和手段

Java是一种完全面向对象的程序设计语言，学习Java编程的最大困难是建立面向对象的思维方式、熟悉面向对象的编程风格，而其中类和对象的概念是基础。在本课程的教学中，我们从一开始就引入类和对象的概念，并以重复和螺旋式上升的方式将面向对象思想贯穿于课程的各个部分，使学生在学习Java语言语法和特性的同时，逐步建立面向对象的思维方式、领会面向对象的编程风格和特征、掌握面向对象程序设计的各项技能。

本课程采用课堂讲授和上机实验相结合的教学方式，上机实验是学习计算机程序设计必不可少的重要环节。

3.学习要求

(1) 已修完《程序设计基础与应用》课程，对计算机语言、程序、数制，程序的顺序结构、选择结构和循环结构，以及计算机系统的组成等知识有基本的了解。

(2) 认真做好预习和复习，能带着问题听课。课后能按时完成相关作业。

(3) 重视实践环节，提高实际动手编程和调试程序的能力，认真编写和提交实验报告。

三、各教学环节学时分配

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 章节内容 | 讲课 | 实验 | 其他 | 合计 |
| 1 | 第1章 Java程序初步 | 3 | 2 |  | 5 |
| 2 | 第2章 数据与数据运算 | 4 | 4 |  | 8 |
| 3 | 第3章 Java语句 | 2 | 4 |  | 6 |
| 4 | 第4章 类和对象 | 6 | 6 |  | 12 |
| 5 | 第5章 字符串和包装类 | 3 | 2 |  | 5 |
| 6 | 第6章 继承 | 4 | 4 |  | 8 |
| 7 | 第7章 接口和包 | 2 | 4 |  | 6 |
| 8 | 第8章 数组 | 2 | 2 |  | 4 |
| 9 | 第9章 例外处理 | 3 | 2 |  | 5 |
| 10 | 第10章 线程与同步 | 3 | 2 |  | 5 |
| **合计** |  | 32 | 32 |  | 64 |

四、教学内容

第1章 Java程序初步

1.1. Java语言特点

1.2 一个简单的Java程序

1.3 编译和运行Java程序

1.4 Java词法结构

教学重点、难点：

重点是类和对象的概念、程序的基本结构以及开发过程。难点是类和对象的概念。

课程的考核要求：

(1) 理解对象和类的概念，理解对象封装的概念。

(2) 了解Java平台无关性的特点。熟悉Java虚拟机、Java平台、Java API等概念。

(3) 了解Java应用程序的基本结构，以及main方法的格式。能够模仿地定义一些简单的类、创建类的实例对象。

(4) 了解Java开发工具包的组成，掌握编译、运行Java应用程序的过程。

(5) 了解Java语言的词法结构。掌握注释的使用，掌握标识符的命名规则。

复习思考题：

(1) 什么叫对象？什么叫类？类和对象有什么关系？

(2) 为什么说Java是一种半编译、半解释的程序设计语言？

(3) 什么叫Java虚拟机？什么叫Java运行环境？Java虚拟机与Java运行环境的关系如何？

(4) Java语言有哪些词法符号？分隔符与空白符号的区别是什么？

课程思政切入点：

从“软件定义”的概念谈软件在现代经济社会的应用广度和深度，树立学生们的职业荣誉感和使命感。

现代经济社会的各行各业都会涉及软件的应用。在2020中国（太原）人工智能大会上，中国科学院院士、解放军军事科学院副院长梅宏指出，我们正在步入一个软件定义的时代。软件已经渗透各行各业，并促进其发生深刻变化。所谓软件定义，就是用软件去定义系统的功能，用软件给硬件赋能，实现系统运行效率和能量效率最大化。

第2章 数据与数据运算

2.1 基本数据类型

2.2 基本类型转换

2.3 运算符

2.4 表达式

教学重点、难点：

重点是各种类型文字的表示；自动类型转换及其规则，以及在数据运算中发生自动类型转换的各种情况。难点是整数的二进制补码表示及其运算特点。

课程的考核要求：

(1) 了解Java语言支持的两类数据类型：基本类型和复合类型。

(2) 了解各种基本类型的特性，掌握各种类型文字的表示。

(3) 理解变量的概念，掌握变量定义的基本方法，了解变量的分类、局部变量的作用域。

(4) 理解基本类型转换的含义、方法和规则。

(5) 掌握各种运算符的使用。

(6) 理解运算符的优先级、结合性和计算次序。理解表达式的的结果和类型。

复习思考题：

定义一个类Ex0202。该类有一个表示年份的int型成员变量year，并包含以下两个方法：

public void set(int y); // 设置成员变量year的值。

public boolean isLeap(); // 报告year表示的年份是否为闰年。

然后再定义一个应该程序起始类TestEx0202测试该类。

说明：闰年是指能被400整除、或者能被4整除但不能被100整除的年份。

第3章 Java语句

3.1 语句概述

3.2 选择语句

3.3 循环语句

3.4 跳转语句

3.5 几个类及使用

教学重点、难点：

重点是理解方法代码运行时的三种基本流程结构：顺序结构、选择结构和循环结构，掌握if-else、switch、while、do-while和for语句的使用。本章的难点是掌握break、continue语句的使用，知道他们的使用范围及相应的功能。

课程的考核要求：

(1) 了解各类Java语句。知道空语句、表达式语句、块语句和标号语句。

(2) 理解选择结构。掌握实现选择结构的if-else、switch语句的使用。

(3) 理解循环结构。掌握实现循环结构的while、do-while和for语句。

(4) 掌握return、break、continue语句的使用。

(5) 掌握穷举、迭代、递归等常用的数据处理方法。

(6) 了解Math和System类的功能和使用。

(7) 掌握格式化输出和从控制台获取输入的方法。

复习思考题：

(1) 有一个数列1，3，6，10，15，21，28，…，tn-1，tn，其各数存在以下递推关系：

tn = tn-1 + n, t1 = 1

编写并测试求数列第n个数的方法static long method(int n)。

(2) 编写并测试方法static int daysInMonth(int y, int m)，方法的功能是计算y年m月份包含的天数。

说明：闰年2月份包含29天，其他年的2月份包含28天。

(3) 编写并测试方法static int gcd(int m, int n)，方法的功能是求两个整数m和n的最大公约数。

课程思政切入点：

从Java标识符的命名，谈规则与灵活性的关系。标识符的命名有一定的规则，这些规则必须遵守，否则就会导致程序错误。在遵守规则的基础上，应该发挥自己的主观灵活性，命名出可读性强的标识符。

这样的关系原则同样存在于日常的生活和工作中。做事先做人，凡事都要讲规矩，无规矩不成方圆。在学校，要遵守学校的各项规章制度；在家里，要孝顺父母，尽量帮父母多做家务；毕业后，要遵守国家法律法规，做一个遵纪守法的好公民。在这个基础上，我们应该能发挥自己的主观能动性，积极地把自己的学习和工作做好。

第4章 类和对象

4.1 引用类型

4.2 类的定义与对象的创建

4.3 成员变量

4.4 方法

4.5 以对象为单位的信息传递

4.6 “has-a”关系

4.7 对象清除

教学重点、难点：

本章和第6章系统地介绍了定义类的各种方法和技能，属于本课程的核心内容，也是难点。本章侧重于不考虑继承特征情况下的类的定义。

课程的考核要求：

(1) 了解对象的引用，理解引用类型与基本类型的区别。

(2) 了解各种类修饰符，了解类的各种语法成分。

(3) 掌握变量定义的方法，理解成员变量与局部变量的区别、实例变量与静态变量（类变量）的区别。

(4) 掌握方法定义的方法，理解实例方法与静态方法（类方法）的区别。

(5) 掌握静态初始化块的定义，理解类变量的初始化过程。

(6) 掌握构造方法的定义，理解实例创建和实例变量初始化的过程。理解构造方法与实例方法的区别。

(7) 理解方法重载的概念，掌握方法重载的方法。

(8) 掌握以对象为单位的信息传递的方法。

(9) 理解“has-a”关系的概念，掌握包含“has-a”关系的类的定义。

(10) 了解垃圾对象、对象清除、垃圾收集器线程等概念。

复习思考题：

修改例4-6中的MyTime类，使其具有以下软件接口：

class MyTime {

public MyTime(); // 将当前时间设置为00:00:00

public MyTime(int h, int m, int s);

public MyTime(MyTime time); // 用time的状态设置当前时间对象的状态

public void incrementSecond(); // 使当前时间对象增加1秒

public boolean equals(MyTime time);// 判断当前时间与参数时间是否相等

public String toString(); // 以hh:mm:ss格式返回当前时间的字符串表示

}

然后在编写一个应用程序起始类TestMyTime测试该类。特别要测试进入下一分、下一小时、下一个天情况，比如：如果当前时间是13:12:59，那么增加一秒后，当前时间将变为13:13:00；如果当前时间是08:59:59，那么增加一秒后，当前时间将变为09:00:00；如果当前时间是23:59:59，那么增加一秒后，当前时间将变为00:00:00。

第5章 字符串和包装类

5.1 String类

5.2 StringBuffer类

5.3 字符串的特殊处理

5.4 包装类

教学重点、难点：

重点是要理解String型字符串对象与StringBuffer型字符串对象的区别。

课程的考核要求：

(1) 理解String型字符串对象与StringBuffer型字符串对象的区别。

(2) 熟练掌握String类的各种构造方法及成员方法的使用。

(3) 了解字符编码的概念，掌握利用一组字节数据创建字符串的方法。

(4) 了解StringBuffer类实例长度与容量的概念，掌握StringBuffer类的各种构造方法及其成员方法的使用。

(5) 了解系统对字符串文字、字符串连接运算+的处理过程。

(6) 了解基本类型包装类的作用，掌握包装类常用方法的使用。

(7) 了解基本类型与包装类之间的自动转换。

复习思考题：

(1) 编写并测试方法static void output(String str)，方法分行输出参数字符串的各字符，且各字母以大写形式输出。例如字符串”Hello”应输出成如下：

H

E

L

L

O

(2) 编写并测试方法static boolean isPalindrome(String str)，方法的功能是判断参数字符串是否为“回文”。

说明：当一个字符串顺读和倒读时都一样，则该串就是“回文”。如”level”、 ”12321”等都是“回文”。

课程思政切入点：

从火神山、雷神山医院建设谈中国速度与国家制度的优越性，然后引出团队合作精神。对软件从业人员来说，团队合作也是其职业素养之一。从需求分析到软件设计与实现，在到软件的维护和升级，需要不同领域、不同层次的专家、分析师、设计师、程序员等相互协作配合，基于各自的职责，朝着同一目标努力。

第6章 继承

6.1 extends短语

6.2 类成员

6.3 “is-a”关系

6.4 成员变量隐藏

6.5 方法覆盖

6.6 再论构造方法

6.7 扩展抽象类

教学重点、难点：

本章和第4章是Java面向对象程序设计课程的核心。第4章侧重于不考虑继承特征情况下的类的定义，本章则介绍考虑继承特征情况下的类的定义。

课程的考核要求：

(1) 理解继承的概念，了解超类、子类、直接超类、直接子类的概念。

(2) 掌握extends子句的使用，了解Java单继承的特点。

(3) 理解类成员的组成。

(4) 理解“is-a”关系的概念。

(5) 理解变量隐藏的概念，掌握变量隐藏时对变量的访问特点和处理方式。

(6) 理解方法覆盖的概念，掌握覆盖方法的定义、方法覆盖时对方法的调用特点。

(7) 了解并掌握关键字this和super的含义及其在各种场合的使用方法。

(8) 了解考虑继承特征情况下实例变量的初始化过程。

(9) 理解抽象类的特点，掌握扩展抽象类的方法。

(10) 了解Object类，掌握equals、clone、finalize、toString等方法。

复习思考题：

(1) 扩展第4章习题2(2)中的Circle类，定义一个表示球体的名为Spheroid的类，该类提供以下构造方法和实例方法：

public Spheroid(int r); // 构造方法，设置球体半径

public double getSurfaceArea(); // 计算球体表面积

public double getVolume(); // 计算球体体积

(2) 假设已定义表示圆的Circle类，该类的软件接口如下：

class Circle {

public Circle(int r); // 构造方法，设置圆半径

public double getArea(); // 计算圆的面积

public double getPerimeter(); // 计算圆的周长

}

现在请通过扩展Circle类定义一个表示圆柱体的名为Cylinder的类，该类包含一个表示圆柱高的int型的变量h，其软件接口如下：

class Cylinder extends Circle {

public Cylinder(int r, int h); // 构造方法，设置圆柱的底圆半径及高

public double getArea(); // 计算圆柱表面积

public double getVolume(); // 计算圆柱体积

}

第7章 接口和包

7.1 接口

7.2 引用类型转换

7.3 包

教学重点、难点：

接口和包都是Java语言中难点。

课程的考核要求：

(1) 理解接口的概念、作用，掌握接口定义、接口实现的方法以及接口作为引用类型的使用方法。

(2) 掌握引用类型转换的方法和规则。

(3) 理解包的概念、作用。

(4) 掌握package、import语句的使用。

(5) 理解访问控制修饰符(private、public、protected)的含义，掌握各种访问控制级别的使用方法。

复习思考题：

有以下接口和类定义：

interface I {

void m1();

}

class C1 extends Object {

void m2() { }

}

class C2 extends C1 implements I {

public void m1() { }

}

基于以上类定义，下面代码片段中正确的是\_\_\_\_\_\_\_\_。

A) C1 a = new C2(); a.m1(); B) C1 b = new C2(); b.m2();

C) C2 c = new C1(); c.m2(); D) I d = new C2(); d.m2();

课程思政切入点：

从神舟航天工程说起，任何一个系统的各个组成部分要很好地完成各自的功能并相互配合实现系统的整体功能，明确各组成部分的接口是非常重要的。在Java软件开发中，也有同样的道理。一个Java软件由一系列的类组成，每个类的调用接口也需要明确定义，只有这样才能够很好地将它们组装在一起。一个Java接口则可以为多个类或其它接口定义共同的接口。

2016年10月17日7时49分，执行与天宫二号交会对接任务的神舟十一号载人飞船，在酒泉卫星发射中心发射升空后准确进入预定轨道，顺利将2名航天员送上太空。神舟十一号载人飞船的发射成功，标志着我国探索太空的脚步又迈出了关键而坚实的一步，是国家综合实力的重要体现，更是航天人充分发扬航天精神的重要体现。

第8章 数组

8.1 数组类型与数组变量

8.2 数组创建

8.3 数组访问

8.4 二维数组

8.5 数组应用举例

教学重点、难点：

重点是要讲清楚数组对象的概念。在Java中，数组既是一种数据结构，也是一种对象。

课程的考核要求：

(1) 理解数组对象的概念。

(2) 掌握数组类型的表示与数组变量的定义。

(3) 掌握数组创建的方法，包括数组创建表达式、数组初始化块。

(4) 掌握对数组元素的访问，了解对数组成员的访问。

(5) 理解二维数组是数组的数组的概念。掌握对二维数组的使用。

复习思考题：

(1) 编写并测试方法static boolean areEqual(int[] x, int[] y)，方法判断数组x和数组y是否相等。若两个数组对应元素都相等就返回真。

(2) 编写并测试方法static Object[][] counting(String str)，方法计算并返回参数字符串str中每个字符出现的次数。返回的Object[][]型二维数组的行数应该为str中不同字符的个数。二维数组的第1列存放各字符的Character实例；第2列存放各字符对应次数的Integer实例。

第9章 例外处理

9.1 例外分类

9.2 引发例外

9.3 声明抛出例外

9.4 捕捉例外

9.5 自定义例外类型

9.6 构造方法与例外处理

教学重点、难点：

重点是要理解例外的概念、例外的分类、系统对例外的处理过程，以及掌握例外的捕捉方法。

课程的考核要求：

(1) 理解例外的概念、熟悉例外的分类。

(2) 理解系统对受检查的例外和不受检查的例外在处理上的不同点和相同点。

(3) 理解例外引发的概念，掌握throw语句的使用。

(4) 理解例外抛出的概念，掌握throws子句的用法。

(5) 掌握try语句的使用。理解并掌握catch和finally子句的作用的用法。

复习思考题：

假设Factorial类中有一个静态method(int n)方法，当n≥0时，方法返回n的阶乘；当n<0时，方法引发一个受检查的ArgumentException型例外。ArgumentException类应该提供一个能够接收并保存n值的构造方法，以及一个能够返回n值的实例方法。现在请实现Factorial类以及ArgumentException类，并编写应用程序测试它们。

课程思政切入点：

从“知行合一”理念谈学习程序设计的方法。在Java语言程序设计的学习过程中，上课听老师讲解知识点是做到“知”，在实践课中完成编程练习是做到“行”。理解了“知行合一”的哲学思想，就会两者有机地有机融合在一起，不可分割。

“知行”是中国传统哲学的重要范畴，其始于《尚书》与《左传》，《尚书》有“非知之艰，行之惟艰”之说，《左传》有“非知之实难，将在行之”之说。“知行合一”出自于《传习录》，它是明代著名哲学家王守仁的弟子们记录老师的学术讲话及论学书信的集子。

第10章 线程与同步

10.1 线程创建

10.2 线程控制

10.3 互斥与同步

教学重点、难点：

重点包括创建线程的方法、线程状态、线程睡眠（sleep）、线程互斥与同步的概念和方法。难点主要是对象锁的概念、线程同步的程序设计。

课程的考核要求：

(1) 理解线程的概念。

(2) 掌握创建线程的两种方法，深刻理解其中的异同点。

(3) 理解线程状态的概念，理解活动线程的概念。理解线程优先级的概念。

(4) 了解线程控制的基本方法，掌握sleep方法的使用。

(5) 理解临界区与互斥的概念，掌握使用synchronized关键字实现互斥的方法。

(6) 了解线程同步的概念，以及wait、notify、notifyAll方法实现线程同步的机制。

复习思考题：

创建一个包含实例变量i的线程。该线程启动后，首先创建并启动一个精灵线程，然后对变量i循环累加1，直至i的值等于Integer.MAX\_VALUE。精灵线程每隔约1s输出一次变量i的值。

五、考核方式、成绩评定

平时成绩(占30%)，上课提问、测验、上机实验与报告

期末成绩(占70%)，考试(闭卷)

六、主要参考书及其他内容

[1] 娄不夜,王利. Java程序设计(第2版). 北京:清华大学出版社.2010.1

[2] Bruce Eckel著, 陈昊鹏 译. Java编程思想（第4版）. 北京: 机械工业出版社.2007.6

[3] 朱喜福. Java程序设计(第二版).北京: 人民邮电出版社.2007.3

[4] 陈国君. Java程序设计基础（第5版）.北京:清华大学出版社. 2015.5

[5] 耿祥义等. Java大学实用教程(第3版).出版社.北京:电子工业出版社.2012.1

[6] 雍俊海. Java程序设计习题集（含参考答案）.北京:清华大学出版社. 2006.6

执笔人：娄不夜 教研室主任：　高静　　　　系教学主任审核签名：