

(七) 专业类课程教学大纲

《面向对象程序设计（Java）》课程教学大纲

课程英文名	Object Oriented Programming (Java)				
课程代码	A0500820	课程类别	学科基础课	课程性质	学科必修
学 分	3		总学时数	48	
开课学院	计算机学院		开课基层教学组织	程序设计课程组	
面向专业	计算机科学与技术		开课学期	第2学期	

一、 课程目标

《面向对象程序设计(Java)》是计算机类及相关专业的一门重要的学科基础课，也是学习后续课程的重要基础。课程以 Java 语言为依托，传授面向对象程序设计思想，通过讲授 Java 程序设计方法，使学生初步具备根据实际问题，设计结构良好、高效的面向对象程序的能力。结合国家建设和民族复兴的新时代背景，增强学生家国情怀与文化自信，激发使命感和责任心。

通过本课程的学习，要达到以下课程目标：

课程目标 1：能够运用 Java 语言的各种机制，具备阅读和调试 Java 程序的基本能力。

课程目标 2：具备运用面向对象思维方式，为实际应用问题建立问题模型，设计合适 Java 类的初步能力。

课程目标 3：能够综合运用 Java 语言机制，设计结构良好、高效的现代 Java 程序。

课程目标 4：具备自主学习、终身学习意识。

课程目标 5：能够掌握面向对象程序设计的新技术与发展趋势，及时掌握国家相关方面的科技战略需求，树立强烈的爱国主义使命感与责任心。

二、 课程目标与毕业要求对应关系

本课程的课程目标对计算机科学与技术专业毕业要求指标点的支撑情况如表 1 所示。

表 1. 课程目标与计算机科学与技术专业毕业要求对应关系

毕业要求	指标点	课程目标及支撑权重

毕业要求	指标点	课程目标及支撑权重
1.工程知识：掌握数学、自然科学、工程基础、计算机专业领域的知识，并能应用于计算机领域复杂工程问题的解决方案中。	1-2 掌握计算机科学核心知识与理论，能够针对计算机领域复杂工程问题建立模型，并利用模型解决问题。	目标 1: 0.4 目标 2: 0.4 目标 5: 0.2
2.问题分析：能够应用数学、自然科学和工程科学的基本原理，以及科学思维方法，对计算机领域的复杂工程问题进行识别、表达和分析，并通过文献查阅与研究获得有效结论。	2-1 能够应用数学、自然科学、工程科学和计算科学的基本原理识别、表达计算机领域的复杂工程问题。	目标 2: 1.0
3.设计/开发解决方案：能够设计计算机领域复杂工程问题的解决方案，设计与开发满足特定需求的软硬件系统、算法或部件，在设计中考虑社会、健康、安全、法律、文化以及环境等因素，并体现一定的创新意识。	3-1 具备计算思维和程序设计能力，能够针对计算机复杂系统设计与开发满足特定需求的模块或算法。	目标 2: 0.5 目标 3: 0.5
12.终身学习：具有自主学习和终身学习的意识，有不断学习和适应信息技术高速发展的能力。	12-2 具备自主学习的能力，包括技术理解力、归纳总结能力和提出问题的能力。	目标 4: 1.0

三、课程目标与教学内容和方法的对应关系

《面向对象程序设计(Java)》课程目标与教学内容、建议教学方法的对应关系如表 6 所示。教师可根据实际情况和自身特长采用合适的教学方法。

表 6 课程目标与教学内容、建议教学方法的对应关系

教学内容	教学方法	课程目标
1. Java 语言概述	课堂讲授、视频学习、案例分析	1,4,5
2. Java 语言编程基础	课堂讲授、视频学习、案例分析、课后实践	1
3. 类与对象	课堂讲授、视频学习、案例分析、课后实践	1,2,3
4. 继承与多态	课堂讲授、视频学习、案例分析、课后实践	1,2,3
5. 数组与字符串	课堂讲授、视频学习、案例分析、课后实践	1,2,3
6. 图形用户界面	课堂讲授、视频学习、课后实践	1,2,3,4
7. 集合与泛型	课堂讲授、视频学习、课后实践	1,2,3
8. 异常处理	课堂讲授、视频学习、课后实践	1,2,3
9. 文件和流	课堂讲授、视频学习、案例分析、课后实践	1,2,3,4
10. 多线程技术	课堂讲授、视频学习、课后实践	1,2,3

课程教学的详细内容与要求如下：

1. Java 语言概述

(1) 教学内容：

- Java 语言的产生和发展历史；
- Java 语言的特点及优势；

- Java 程序的编辑、编译和运行过程;
- Java 开发工具包和集成开发环境的下载、安装和使用。

(2) 教学重点: Java 语言的特点, Java 程序的编译和运行原理, 开发工具下载、安装和使用。

(3) 教学难点: Java 虚拟机工作原理、Java 编译和运行原理、与 C、C++ 语言的区别。

(4) 教学要求: 能够下载 Java 开发工具包、安装和使用 Java 集成开发环境; 能够编辑、编译和运行 Java 程序; 能够正确理解 Java 虚拟机工作原理、Java 编译和运行原理、与 C、C++ 语言的区别。

思政融合点 1: Java 是一种开源的编程语言。目前, 开源软件以开放、共享、协同的新型生产方式, 成为全球信息技术发展的强大推动力。2020 年中国开源的发展速度很快, 中国开源已经接近或部分达到世界先进水平。举例介绍我国开源生态建设情况, 结合国家相关方面的科技战略需求, 帮助学生树立强烈的爱国主义使命感与责任心。

思政融合点 2: Java 是一种开源的编程语言。安装好 JDK 后, 向学生展示 java 类库的源码, 让学生体会开源, 今后为开源社区贡献代码。

思政融合点 3: 介绍 java 的应用领域和应用前景, 使学生能够把自身价值的实现和国家发展紧密联系起来。介绍学习面向对象编程技术的必要性和重要性, 使学生明白学好本课程能为接下来专业课的学习打好基础, 从而更好地进行创新实践和科学研究。

2. Java 语言编程基础

(1) 教学内容:

- 标识符的命名原则, Java 语言的主要关键字;
- Java 的基本数据类型的分类、识别及使用;
- 常量和变量的表示、声明和使用;
- 主要运算符的含义、使用方法、优先级别和结合性;
- 由运算符和运算对象构成所需的表达式;
- Java 语言语句的定义和分类;
- Java 语言简单语句、条件语句和循环语句的含义和使用;
- Java 语言跳转语句的含义和使用;
- 控制台输入和输出;
- 编码规范; 理解常见的编程错误类型。

(2) 教学重点: 标识符的命名原则, 基本数据类型和类类型转换, 常量和变量, 运算符与表达式, 控制台输入和输出, 程序流程控制语句。

(3) 教学难点: 数据类型的自动转换和强制类型转换, 运算符的准确使用, 程序流程控制语句。

(4) 教学要求: 能够熟练掌握标识符的命名原则, 并根据标识符的命名规则正确命名单词。能够熟练掌握 Java 的基本数据类型、常量和变量、运算符、表达式、程序流程控制语句, 能够正确定义数据类型的转换, 能够准确使用运算符和运算对象构成表达式, 能够运用程序流程控制语句。

设计程序。

3. 类和对象

(1) 教学内容:

- 类和对象的基本概念，面向对象语言的信息封装的含义及其优势；
- 类的定义、对象的创建；
- 对象与基本数据类型变量的区别、静态成员；
- 方法定义、方法调用、参数传递、递归和方法重载；
- 类的构造方法；
- 对象的初始化的概念和初始化顺序；
- 包、访问权限控制、作用域、关键字 this 的概念和用法；
- 类库中 Math、BigInteger、BigDecimal、类 System 的使用。

(2) 教学重点：类的定义，对象的创建，构造方法的定义与使用，成员方法的定义与使用，方法调用、参数传递，方法的重载与覆盖，访问权限控制，关键字 this。

(3) 教学难点：面向对象的思维方式，类与对象的概念，访问控制权限的使用，抽象类的应用，接口的应用。

(4) 教学要求：能够正确理解类和对象的概念，并根据面向对象的思维方式，设计类和创建类的对象。能够正确理解对象与基本数据类型变量的区别，掌握方法的定义、方法的调用、参数传递，递归和方法的重载。掌握类的构造方法，并通过构造方法实现对象的初始化和初始化概念。掌握包及访问控制权限，熟练使用 Java 类库中的 Math、System 等类。

4. 继承与多态

(1) 教学内容:

- Java 语言中类的继承的含义；
- 继承的语法、子类的构造方法；
- 子类对象的初始化顺序；
- 方法覆盖、成员隐藏、关键字 super 和 final、对象类型转换、多态与动态绑定、抽象类和抽象方法、接口；
- 类 Object、包装类、自动装箱与拆箱的概念。
- 内部类的定义与使用。

(2) 教学重点：类的继承、子类构造方法，多态与动态绑定、关键字 super，抽象类和抽象方法、接口。

(3) 教学难点：类的继承与运行时多态、抽象类、接口的综合应用，访问控制权限的使用。

(4) 教学要求：能够正确理解 Java 中类的继承的概念及其语法结构，理解子类的构造函数，并通过子类构造函数实现子类对象的初始化顺序。熟练掌握方法覆盖、成员隐藏、关键字 super 和 final、对象类型转换、多态、抽象类及接口。

5. 数组与字符串

(1) 教学内容:

- 数组和数组元素的概念;
- 一维数组的声明、创建、初始化和使用;
- 二维及多维数组的声明、创建、初始化和使用;
- 数组排序和在数组中查找;
- `foreach`语句, 方法传递与返回数组;
- 字符串的概念, 字符串类 `String`, `StringBuilder` 和 `StringBuffer` 类的创建和使用。

(2) 教学重点: 一维数组的声明、创建、初始化和使用, 字符串类 `String`, `StringBuilder` 和 `StringBuffer` 类的创建和使用。

(3) 教学难点: 数组的应用, 字符串类 `String` 和 `StringBuffer` 的使用方法的区别。

(4) 教学要求: 能够正确理解数组和数组元素的概念, 熟练掌握一维数组的声明、创建、初始化和使用, 理解二维数组及多维数组的声明、创建、初始化和使用, 掌握数组排序和数组查找, 理解字符串的概念, 掌握字符串类 `String`, `StringBuilder` 和 `StringBuffer` 的创建和使用, 并准确区分这三者。

6. 图形用户界面

(1) 教学内容:

- Java 图形用户界面组件类型及关系;
- AWT 组件, 布局管理, 事件处理机制, Swing 组件层次结构以及常用 Swing 组件, Java 图形程序设计。

(2) 教学重点: 常用的 GUI 容器, 布局管理器, 组件, 事件处理机制, 常用事件监听器。

(3) 教学难点: 布局管理器的灵活应用, 事件处理机制。

(4) 教学要求: 了解 Java 图形用户界面组件类型及关系, 能够正确理解 AWT 组件和 Swing 组件, 运用布局管理器实现对 GUI 容器中组件进行合理布局, 并运用事件监听器及事件处理机制实现组件的消息通讯。

思政融合点 4: 本部分教学内容要求学生自学后设计实现一个基于图形界面的管理系统, 同时对系统的设计拔高要求, 要能从计算机系统的可重用性和可维护性角度出发设计结构良好的系统。学生在自学和开发过程中, 不仅要编码实现, 还要搭建结构优良的软件架构, 通过这种实践增强学生的专业自信心, 从而树立研究前沿技术、不断创新的远大志向。

思政融合点 5: 良好的编程习惯是编写良好 java 程序的先决条件。在完成基于图形界面的管理系统时, 要求学生按照 Java 编程规范进行代码编写, 培养学生具备优秀程序员的素养。

思政融合点 6: 学生在完成基于图形界面的管理系统时, 可以小组为单位协作完成, 项目过程中培养学生团结协作、诚信友善的职业素养和沟通交流能力。

7. 异常处理

(1) 教学内容:

- Java 异常处理机制;

- 异常类的定义，异常类的层次结构，异常的捕获和处理、指定方法抛出的异常、抛出异常；
- 理解异常处理原则。

(2) 教学重点：异常的基本概念，异常类的定义，异常的抛出、捕获和处理。

(3) 教学难点：异常处理机制，定义、抛出和使用异常类。

(4) 教学要求：能够正确理解 Java 异常处理机制，掌握异常类的定义，异常类的层次结构，异常的抛出、捕获和处理知识点，并融入异常处理机制设计健壮的 Java 程序。

8. 文件和流

(1) 教学内容：

- JAVA 的输入和输出流的概念，输入和输出类的层次；
- 常见的输入和输出流类： InputStream 和 OutputStream， DataInputStream 和 DataOutputStream， FileInputStream 和 FileOutputStream， PrintStream， 带缓存的输入流和输出流理解、异常处理原则；
- 文件、随机访问文件的含义。 FileReader 和 FileWriter， RandomAccessFile， 带缓存的读写器；
- 对象序列化。

(2) 教学重点：输入输出流的基本概念，文件与目录管理，文件输入流和文件输出流，文件读写。

(3) 教学难点：字节流和字符流的差异，根据需要选择使用恰当的输入输出类，随机读写文件。

(4) 教学要求：能够掌握 Java 的输入和输出流的概念，输入和输出类的层次，掌握常见的字节输入输出流类、字符输入输出流类以及文件，并根据需要选择恰当的输入输出流类，实现文件的读写和随机访问。

9. 集合与泛型

(1) 教学内容：

- java 中泛型的作用和使用方法；
- 集合的概念和 java 的集合架构；
- 常用集合类型：接口 Collection、接口 List、接口 Set、接口 Map、类 ArrayList、类 HashSet、类 HashMap 等的使用方法；
- 迭代器 Iterator 的使用方法。

(2) 教学重点：泛型的基本概念，集合的概念，常用集合类的使用、迭代器的使用。

(3) 教学难点：java 集合架构的理解，List、Set、Map 的使用场景和常用方法

(4) 教学要求：了解泛型和集合的概念，理解 java 的集合架构，掌握 List、Set、Map 的区别，能够在实际应用中能选择合适的集合类和方法来解决问题。

思政融合点 7：从计算机系统的可重用性和可维护性角度出发设计结构良好的系统，学生在不仅要学会编码，还需要学习一些数据结构，才能编写结构优良的软件。

10. 多线程技术

(1) 教学内容:

- 线程的概念;
- JAVA 线程的创建、线程状态与线程控制;
- 线程优先级，线程调度，守护线程，线程异常的概念，线程同步及死锁问题。

(2) 教学重点: JAVA 线程的创建、线程状态与线程控制。

(3) 教学难点: 线程优先级和线程调度、线程同步、死锁问题和处理方法。

(4) 教学要求: 能够正确理解线程的概念，熟练掌握 Java 线程的创建、线程状态与线程控制，了解线程优先级，线程调度，守护线程，线程异常，理解线程同步及死锁问题。

四、 实践环节及要求

本课程的实践环节安排 12 课时的课内上机和 20 课时的课外上机，上机内容和要求安排如下表 7 和表 8 所示：

表 7 课内实验/上机项目安排表

序号	实验项目	学时	基本要求	实验性质	实验类别
1	编写一个简单的 Java 程序	1	熟悉 Java 开发环境，完成实验	必做	验证
2	类和对象的定义、类的继承	2	掌握类和对象的定义、类继承	必做	综合
3	多态与接口	2	掌握类的重载和覆盖的应用，了解多态和接口	必做	综合
4	数组、集合、字符串处理与异常 处理应用程序的开发	2	掌握数组、集合、字符串处理和异常处理方法	必做	综合
5	输入输出编程	2	理解 Java 输入输出	必做	综合
6	课程设计	3	综合完成一个 Java 程序设计	必做	综合

表 8 课外实验/上机安排表

序号	实验项目	学时	基本要求	实验性质	实验类别
1	类与对象练习	3	掌握类和对象的定义	必做	综合
2	继承与多态练习	4	掌握类继承、多态、接口	必做	综合
3	数组与字符串练习	2	掌握数组、字符串处理	必做	综合
4	课程设计图形界面练习	3	了解图形界面编程	必做	综合
5	文件和流练习	3	掌握常用的文件和流类	必做	综合
6	集合与泛型练习	3	掌握常用的集合类和接口	必做	综合
7	多线程技术练习	2	了解多线程的使用方法	必做	综合

五、 与其它课程的联系

先修课程：程序设计基础

后续课程：数据结构

六、学时分配

表 9 学时分配表

教学内容	讲课时数	实验时数	实践学时	上机时数	自学时数 (课外上机)	习题课	讨论时数
1 Java 语言概述	2						
2 Java 语言编程基础	4			1		0.5	
3 类与对象	6			3	3	0.5	
4 继承与多态	6			3	4	1	
5 数组与字符串	3			1	2	0.5	
6 图形用户界面	2			1	3		
7 异常处理	2			1			
8 文件和流	3			2	3		
9 集合与泛型	3				3	0.5	
10 多线程技术	2				2		
合计	33			12	20	3	
总计				48 学时+20 课外上机自学学时			

七、课程目标达成途径及学生成绩评定方法

1. 课程目标达成途径

课程目标的达成途径如表 10 所示。

表 10 课程目标与达成途径

课程目标	达成途径
课程目标(1)：能够运用 Java 语言的各种机制，具备阅读和调试 Java 程序的基本能力。	以引导式、启发式和总结式教学方法为主，通过重点/难点内容讲解、课后作业、进行随堂提问、课堂程序演示等模式，帮助学生能够运用 Java 语言的各种机制，具备阅读、理解和调试各类 Java 程序的能力。
课程目标(2)：具备运用面向对象思维方式，为实际应用问题建立问题模型，设计合适 Java 类的初步能力。	以启发式、分析式和研讨式教学方法为主，结合典型应用案例，通过上机实践、作业、课程设计、随堂提问、课堂讨论等模式，帮助学生具备运用面向对象思维方式，为实际应用问题建立问题模型，设计合适 Java 类的初步能力。
课程目标(3)：能够综合运用 Java 语言机制，设计结构良好、高效的现代 Java 程序。	以引导式、启发式和总结式教学方法为主，通过重点/难点内容讲解、上机实践、课后作业、进行随堂提问、课堂讨论等模式，帮助学生掌握设计结构良好、高效的现代 Java 程序，解决程序设计领域的复杂工程问题。
课程目标(4)：具备自主学习和终身学习意识。	以自学方式为主，通过作业或者课程设计的形式，要求学生通过查阅资料自学类库使用方法来解决问题，培养学生的自主学习能力、终身学习意识。

课程目标(5): 能够掌握面向对象程序设计的新技术与发展趋势, 及时掌握国家相关方面的科技战略需求, 树立强烈的爱国主义使命感与责任心。	以案例教学法、讲授法介绍课程的前沿技术, 使学生进一步了解目前国内相关先进技术与取得的成就, 从而建立强烈的民族自豪感与爱国主义使命感。
--	--

2.学生成绩评定方法

该课程为考试课程, 考试方式为闭卷。该课程采用形成性评价与终结性评价相结合的评价方法, 学期总评成绩由两部分构成: 平时成绩和期末考试成绩。采用传统教学模式(以教师讲授为主), 建议平时成绩占比 40%、期末考试成绩占比 60%。如果采用混合教学模式/翻转课堂模式, 建议平时成绩占比 50%、期末考试成绩占比 50%。各部分的建议考核内容、在平时成绩中的建议比例、关联课程目标、在总成绩中的占比等, 如表 11、表 12 所示。各考核内容的详细评分标准见表 13 所示。

表 11 课程考核与成绩评定方法

成绩构成	考核项目	考核关联的课程目标	考核依据与方法	占总评成绩的比重
平时成绩	课程思政实践	(5)	基于课程设计、前言介绍等主题, 通过课堂讨论、阅读文献等多种形式, 考查学生对我国相关先进技术的了解情况以及核心价值观状况	5%
	课堂表现	(1)(2)(3)(4)	通过课堂测验、课堂讨论问答、考勤等情况评定, 占比不超过 20%	35%
	上机作业	(1)(2)(3)(4)(5)	上机作业是否及时完成、作业完成质量。包括上机课程设计成绩。占比不超过 30%	
期末考试	闭卷考试	(1)(2)(3)	考试成绩	60%
总评成绩		(1)(2)(3)(4)(5)	=平时成绩*40%+考试成绩*60%	100%

表 12 课程考核与成绩评定方法(混合教学模式/翻转课堂)

成绩构成	考核项目	考核关联的课程目标	考核依据与方法	占总评成绩的比重
平时成绩	课程思政实践	(5)	基于课程设计、前言介绍等主题, 通过课堂讨论、阅读文献等多种形式, 考查学生对我国相关先进技术的了解情况以及核心价值观状况	5%
	课堂表现	(1)(2)(3)(4)	通过课堂测验、课堂讨论问答、考勤等情况评定, 占比不超过 20%	45%
	线上学习	(1)(2)(3)(4)	通过线上视频学习、线上作业完成情况考核, 占比不超过 30%	
	上机作业	(1)(2)(3)(4)(5)	上机作业是否及时完成、作业完成质量。包括上机课程设计成绩。占比不超过 30%	
期末考试	闭卷考试	(1)(2)(3)	考试成绩	50%
总评成绩		(1)(2)(3)(4)(5)	=平时成绩*40%+考试成绩*60%	100%

表 13 考核内容详细评分标准

考核内容	评分标准			
	90-100	75-90	60-75	<60

课程思政实践	报告条理清晰，文字流畅，字数 ≥ 2000 ，紧扣主题；内容完整且材料丰富，体现强烈的使命感、责任心与民族自豪感	报告条理清楚，字数 ≥ 1500 ，相关性较好；内容完整，材料不够丰富，能体现学生的使命感、责任心与民族自豪感	报告有一定条理，字数 ≥ 1000 ，和主题基本相关；内容基本完整但材料较少，能体现学生的使命感与民族自豪感	报告字数 < 1000 ；内容少，或有抄袭现象，体现不出学生的使命感与民族自豪感
课堂表现	课堂测验、课堂练习、回答问题等据实评价；或参与回答次数在教学班前 15%	课堂测验、课堂练习、回答问题等据实评价；或参与回答次数在教学班前 50%	课堂测验、课堂练习、回答问题等据实评价；或参与回答次数在教学班前 85%	课堂测验、课堂练习、回答问题等据实评价；或参与回答次数在教学班后 15%
线上学习	按照线上教学平台导出的视频学习和作业完成数据据实评价			
上机作业	标准题目：按照作业题目评分标准据实评价			
期末考试	按照期末试卷评分标准据实评价			

八、 教学资源

表 14 课程的基本教学资源

资源类型	资源
教材	《Java 程序设计》(第 4 版), 郎波编著, 清华大学出版社, 2021 年 1 月
参考书籍或文献	(1) Java 编程思想 (第 4 版), [美] Bruce Eckel, 机械工业出版社, 2007 年. (2) Java 核心技术 卷 I 基础知识 (原书第 11 版), [美] Horstmann,C.S. , 机械工业出版社, 2019 年. (3) Effective Java 中文版 (原书第 3 版), 约书亚·布洛克 (Joshua Bloch) 著, 俞黎敏译, 机械工业出版社, 2018 年. (4) 深入理解 Java 虚拟机: JVM 高级特性与最佳实践 (第 3 版), 周志明 编, 机械工业出版社, 2020 年. (5) Java 实用教程 (第 2 版), 王小科、罗二平、刘会衡主编, 人民邮电出版社, 2012 年. (6) Java 语言程序设计, 胡维华等编著, 高等教育出版社, 2012 年. (7) Java 程序设计教程(第 2 版), 刘慧宁等编著, 机械工业出版社, 2015 年. (8) Java2 实用教程 (第五版), 耿祥义, 张跃平编著, 清华大学出版社, 2017 年.
网络资源	Java 官方网站 https://www.oracle.com/java/
教学文档	无

九、 课程目标达成度定量评价

在课程结束后，需要对每一个课程目标（含思政课程目标）进行达成度的定量评价，用以实现课程的持续改进。

课程目标达成度的定量评价算法：

- 使用教学活动（如课程思政实践、上机作业、课堂练习、单元测验、视频学习、在线测试、课程设计等等）成绩或期末考试部分题目得分率作为评价项目，对某个课程目标进行达成度的定量评价；
- 为保证考核的全面性和可靠性，要求对每一个课程目标的评价项目选择至少两种；
- 根据施教情况，评价项目可以由教师自行扩展，权重比例可以由教师自行设计；

4、对某一个课程目标有支撑的各评价项目权重之和为 1；

5、使用所有学生（含不及格）的平均成绩计算。

采用传统讲授方式授课，本课程的课程目标达成度的定量评价算法建议如表 15 所示，采用混合模式/翻转课堂授课，本课程的课程目标达成度的定量评价算法建议如表 16 所示，教师可根据授课方式及考核内容适当调整。

表 15. 课程目标达成度定量评价方法

课程目标	课程目标达成度评价方式
课程目标 1： 能够运用 Java 语言的各种机制，具备阅读和调试 Java 程序的基本能力。	课堂表现 0.2 上机作业 0.4 期末考试：0.4
课程目标 2： 具备运用面向对象思维方式，为实际应用问题建立问题模型，设计合适 Java 类的初步能力。	课堂表现 0.3 上机作业 0.5 期末考试：0.2
课程目标 3： 能够综合运用 Java 语言机制，设计结构良好、高效的现代 Java 程序。	课堂表现 0.2 上机作业 0.6 期末考试：0.2
课程目标 4： 具备自主学习、终身学习意识。	课堂表现 0.2 上机作业 0.8
课程目标 5： 能够掌握面向对象程序设计的新技术与发展趋势，及时掌握国家相关方面的科技战略需求，树立强烈的爱国主义使命感与责任心。	课程思政实践 0.9 上机作业 0.1

表 16. 课程目标达成度定量评价方法（混合模式/翻转课堂授课）

课程目标	课程目标达成度评价方式
课程目标 1： 能够运用 Java 语言的各种机制，具备阅读和调试 Java 程序的基本能力。	课堂表现 0.2 上机作业 0.2 期末考试：0.3 线上学习：0.3
课程目标 2： 具备运用面向对象思维方式，为实际应用问题建立问题模型，设计合适 Java 类的初步能力。	课堂表现 0.3 上机作业 0.3 期末考试：0.2 线上学习：0.2
课程目标 3： 能够综合运用 Java 语言机制，设计结构良好、高效的现代 Java 程序。	课堂表现 0.2 上机作业 0.4 期末考试：0.2 线上学习：0.2
课程目标 4： 具备自主学习、终身学习意识。	课堂表现 0.1 上机作业 0.3 线上学习：0.6
课程目标 5： 能够掌握面向对象程序设计的新技术与发展趋势，及时掌握国家相关方面的科技战略需求，树立强烈的爱国主义使命感与责任心。	课程思政实践 0.9 上机作业 0.1

十、说明

本大纲规定了杭州电子科技大学计算机科学与技术、软件工程、智能计算与数据科学（计算机科学与技术）、计算机科学英才班（计算机科学与技术）、计算机科学与技术（第二学士学位）专业的《面向对象程序设计(Java)》课程的教学要求和教学规范，承担《面向对象程序设计(Java)》课程的教师须遵照本大纲安排授课计划、实施教学过程，完成学生学习成果评价、课程目标达成度评价。

本课程大纲于 2021 级开始执行，生效之日原先版本均不再使用。

十一、 编制与审核

表 14.大纲编制与审核信息

工作内容	责任部门或机构	负责人	完成时间
编制	程序设计课程组	龚晓君	2022.02.25
审核	程序设计课程组	徐小良	2022.03.05
审定	计算机学院教学工作委员会	傅婷婷	2022.05.15