**《C++程序设计》课程教学大纲**

**一、课程简介**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **课程中文名** | C++程序设计 | | | | | | |
| **课程英文名** | C++ Program Design | | | | **双语授课** | | □是 ☑否 |
| **课程代码** | 10112110 | **课程学分** | 4 | **总学时数** | | 64（含实践32） | |
| **课程类别** | □通识教育课程  □公共基础课程  ☑专业教育课程  □综合实践课程  □教师教育课程 | **课程性质** | □必修  ☑选修  □其他 | **课程形态** | | □线上  ☑线下  □线上线下混合式  □社会实践  □虚拟仿真实验教学 | |
| **考核方式** | □闭卷 □开卷 □课程论文 ☑课程作品 □汇报展示 □报告  □课堂表现 □阶段性测试 □平时作业 □办公自动化竞赛 | | | | | | |
| **开课学院** | 大数据与智能工程学院 | | **开课**  **系(教研室)** | 物联网工程系 | | | |
| **面向专业** | 计算机科学与技术 | | **开课学期** | 第4学期 | | | |
| **课程负责人** | 袁伟青 | | **审核人** | 曾俊、张素兰、邢昌元 | | | |
| **先修课程** | 高等数学、线性代数 | | | | | | |
| **后续课程** | 企业级应用系统综合实训、毕业论文（设计） | | | | | | |
| **选用教材** | 1.郭炜. 新标准C++c程序设计[M]. 北京:高等教育出版社, 2016. | | | | | | |
| **参考书目** | 1.Stephen Prata等. C++Primer Plus(第六版)[M]. 北京:人民邮电出版社, 2012.  2.朱红等. C++程序设计教程[M]. 北京:清华大学出版社, 2016.  3.钱能. C++程序设计教程(第3版)[M]. 北京:清华大学出版社, 2020. | | | | | | |
| **课程资源** | 1. 程序设计与算法(三)C++面向对象程序设计.中国大学MOOC(国家精品课程).   https://www.icourse163.org/course/PKU-1002029030?from=searchPage&outVendor=zw\_mooc\_pcssjg\_ | | | | | | |
| **课程简介** | 本课程是计算机科学与技术专业选修课程，是高级语言程序设计技术方面的技术基础课，具有很强的实践性。课程内容是面向对象程序设计的基本概念，包含C++语言基本语法和简单程序设计、自定义数据类型；函数的定义和使用方法；类和对象的定义和使用；数据的共享与保护；数组、指针和字符串的概念和使用及动态存储分配的方法；继承与派生的概念及定义派生类的方法；多态性的概念及运算符重载、虚函数的实现方法等等。还包含面向对象程序设计的基本思想和编程方法，并通过大量的实际例子，来示范利用C++程序设计语言和面向对象程序设计技术来解决实际问题的过程和方法。 | | | | | | |

**二、课程目标**

**表2-1 课程目标**

|  |  |
| --- | --- |
| **序号** | **具体课程目标** |
| **课程目标 1** | 明晰C++程序设计的基本概念和常用术语，能够复述C++程序设计的基础知识；能够选择合适的C++软件开发环境；识别并确认C++程序的设计、编码、调试、测试、发布等基本技能。能够应用过程化程序结构组织，全局数据、静态数据、动态数据和名空间概念。能够使用对象化程序设计方法。抽象数据类型的构造及实现方法，操作符重载。 |
| **课程目标 2** | 能够阐释类继承体系，多态特征；能够理解虚函数实现机制；进一步掌握面向对象程序设计方法和技术。 |
| **课程目标** 3 | 能够操作函数模板和类模板  理解并实践C++异常处理机制  学会使用C++标准库和标准模板类库。 |

**表2-2 课程目标与毕业要求对应关系（计算机科学与技术专业）**

| **毕业要求** | **指标点** | **课程目标** |
| --- | --- | --- |
| **毕业要求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，识别、表达、并通过文献研究分析计算机复杂工程问题，以获得有效结论。【M】 | 2.3 针对计算系统设计与实现的需要，选择或建立合适的数学模型，分析计算系统中相互制约因素，优化问题求解。 | 课程目标1 |
| **毕业要求3.设计/开发解决方案：**能够针对计算机复杂工程问题的解决方案，设计并实现满足特定需求的软件系统，并能够在设计和实现环节中体现创新意识，考虑社会、健康、安全、法律、文化以及环境等因素。【L】 | 3.4 能够进行计算机软件系统的子系统、子模块的设计与实现，并能够从全局的角度协调它们之间的关系。 | 课程目标2 |
| **毕业要求4.研究：**能够基于科学原理并采用科学方法对计算机复杂工程问题进行研究，设计实验、分析与解释数据，并能通过信息综合得到合理有效结论。【H】 | 4.2 能够根据实验设计的目标，有效地提取和搜索实验数据，保证数据的真实性和完整性。 | 课程目标3 |

**三、课程学习内容与方法**

**（一）理论学习内容及要求**

**表3-1 课程目标、学习内容和教学方法对应关系**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **序号** | **课程模块** | **学习内容** | **学习任务** | **课程目标** | **学习重点难点** | **教学方法** | **学时** |
| 1 | 开发环境及基础知识 | 1.C++发展史、C++在编程界的地位，C++编程流程；  2.程序开发过程及编译过程 | 1.拓展阅读：C++的诞生与发展  2.线上学习：C++的主流编译器，以及C++的应用 | 课程目标1 | 重点：程序开发过程及编译过程  难点：程序开发过程及编译过程 | 课堂讲授：引导学生了解计算机计算机语言的发展，特别是C++语言的诞生，以及在高级语言中的地位，应用领域  小组讨论：共享新技术的信息，憧憬未来科技世界 | 2 |
| 2 | 面向对象基础 | 1.类的基本概念，掌握类的定义方法；信息隐藏、封装和抽象数据类型等基本概念以及它们与类的关系； | 1. 拓展阅读：面向对象编程的特点；C++语言中类的作用：便于数据安全、代码重用、代码维护的特性。 2. 线上学习：C++语言程序设计关于类的封装的课程视频   3.个人作业：用C++语言编程实现类的定义以及构造函数、析构函数的书写。 | 课程目标1 | 重点：类的基本概念；信息隐藏、封装和抽象数据类型等基本概念以及它们与类的关系；对象、方法和消息等基本概念，定义和运用对象、方法；方法的定义和引用，方法重载，const对象和const成员函数；构造函数和析构函数的特点，构造函数和析构函数的功能；复合类的含义，复合类构造函数的定义和使用方法；  难点：构造函数和析构函数的功能；复合类的含义，复合类构造函数的定义和使用方法； | 课堂讲授：  引导学生全面了解C++编程语言类的语法规则，掌握类的构造函数和析构函数的意义与书写。  案例教学：  引导学生学习类的定义以及对象的实例化，实际编写代码来领会类是支持数据封装的工具，对象则是数据封装的实现。 | 12 |
| 2.对象、方法和消息等基本概念，定义和运用对象、方法； |
| 3.方法的定义和引用，方法重载，const对象和const成员函数； |
| 4.构造函数和析构函数的特点，构造函数和析构函数的功能； | 课程目标2 |
| 5.复合类的含义，复合类构造函数的定义和使用方法； |
| 6.指向对象的指针、指向类成员的指针和this指针的概念，成员对象和对象数组； |
| 7.静态类成员，接口和实现的分离，友元。 |
| 3 | 继承与虚函数、多态 | 1.运算符重载 | 1.拓展阅读：面向对象语言中继承与派生机制提高代码的复用性特性  2.线上学习：继承中的构造与析构规则；以及运算符重载的函数本质  3.个人作业：定义出基类和派生类；掌握不同继承方式下对基类成员的访问控制 | 课程目标2 | 重点：不同继承方式的派生类对基类成员的访问控制规则；运算符重载的函数书写规则；  难点：派生类的构造与析构函数；赋值运算符的重载；  对于多态的理解； | 课堂讲授：引导学生了解派生类的继承规则；运算符重载的函数本质以及语法要求；多态的实现。  案例教学：利用编程实例掌握派生继承以及运算符重载的代码规则；理解多态的意义。  小组讨论：多态的实现以及意义。 | 10 |
| 2．熟悉三种继承方式（public、private和 protected） | 课程目标2 |
| 3．继承的构造函数与析构函数 | 课程目标2 |
| 4．虚函数原理与应用 |
| 4 | 输入/输出流 | 1.命名空间 | 1. 拓展阅读：C++命名空间namespace的出现意义，以及使用   线上学习：C++的输入输出流； | 课程目标1 | 重点：标准流对象  难点：流对象的成员函数使用 | 课堂讲授：引导学生了解高级语言命名空间的引入原因；  案例教学：C++语言的输入输出格式控制， | 4 |
| 2.标准输入输出及格式控制  。 | 课程目标1 |
| 5 | 文件操作 | 1.文件流类的继承关系；ofstream类建立文件、打开文件和读文件； | 1.拓展阅读：阅读C++中的文件操作类介绍，提供的三个子类，掌握文件的操作  2.线上学习：文件流的涵义与文件的操作。 | 课程目标1 | 重点：文件流类成员函数使用  重点：文件流类成员函数使用 | 课堂讲授：C++中的文件流类及其成员函数的使用  案例教学：通过编程实例使学生掌握串流类和文件流类的使用，并进行基本的文件操作。 | 2 |
| 2.ifstream类打开文件和写文件；使用文件流读写文本文件；文件流读写二进制文件 | 课程目标1 |
| 6 | 泛型函数设计与模板 | 1．模板的机制 | 1.拓展阅读：阅读C++中参数化容器类与C++模板的实现  2.线上学习：函数模板和类模板的使用 | 课程目标3  课程目标3 | 重点：模板的机制  难点：模板的编写 | 课堂讲授：C++中的模板的引入的意义以及应用；  案例教学：通过编程实例使学生掌握模板的使用，并进行基本的模板实例化代码编写操作。 | 2 |
| 2．模板的编写 |
| 合计 |  |  |  |  |  |  | 32 |

1. **实验学习内容及要求**

**表3-2 课程目标、学习内容和教学方法对应关系**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **序号** | **项目名称** | **项目来源** | **教学目标（观测点、重难点）** | **学时数** | **项目类型** | **要求** | **每组人数** | **教学方法** | **课程目标** |
| 1 | 初入C++ | 实验教材 | 1.学生能够了解和使用Visual Studio 2010集成开发环境 | 2 | 验证性 | 必做 | 1 | 实验指导 | 课程目标1、2 |
| 2.学习完整的C++程序开发过程； |
| 3.学生能够理解简单的C++程序结构；（重点） |
| 4.学生能够掌握C++程序基本的输入输出操作。（难点） |
| 2 | 基于函数的模块化程序设计 | 实验教材 | 1.学生能够掌握使用不带默认参数的函数 实现参数传递 | 2 | 验证性 | 必做 | 1 | 实验指导 | 课程目标2、3 |
| 2.学生能够掌握带默认参数的函数实现参数传递 |
| 3.学生能够学习使用引用实现实参值的更改  （重点、难点） |
| 3 | 类的定义和对象的创建 | 实验教材 | 1、学生能够理解类的声明和使用； | 4 | 设计研究性 | 必做 | 1 | 实验指导 | 课程目标2 |
| 2、学生能够理解对象的声明和使用;（重点、难点） |
| 3、学生能够掌握具有不同访问属性的成员的访问方式； |
| 4 | 基于数组构造的程序设计 | 实验教材 | 1、学生能够掌握一维数组和二维数组的定义、赋值和输入输出的方法；（重点） | 4 | 验证性 | 必做 | 1 | 实验指导 | 课程目标2 |
| 2、学生能够掌握C++构造函数在数组中的使用（难点） |
| 3、学生能够理解与数组相关的算法 |
| 5 | 对象数组与指针的应用 | 实验教材 | 1、学生能够理解对象数组和对象指针的意义（重点、难点） | 2 | 验证性 | 必做 | 1 | 实验指导 | 课程目标2 |
| 2、学生能够理解C++中静态成员的意义 |
| 6 | 封闭类与友元 | 实验教材 | 1、学生能够掌握封闭类的概念及其对象的创建；（重点） | 2 | 验证性 | 必做 | 1 | 实验指导 | 课程目标2 |
| 2、学生能够掌握友元的应用（难点） |
| 7 | 运算符重载 | 实验教材 | 1、学生能够掌握运算符重载的意义和语法规则（重点） | 4 | 设计研究性 | 必做 | 1 | 实验指导 | 课程目标2 |
| 2、学生能够掌握普通算数运算符的重载方法（难点） |
| 3、学生能够掌握运算符重载函数代码设计与编写 |
| 4、学生能够理解浅拷贝和深拷贝的含义 |
| 8 | 继承与派生 | 实验教材 | 1、学生能够理解并掌握派生类的构造函数和析构函数（重点） | 4 | 设计研究性 | 必做 | 1 | 实验指导 | 课程目标1 |
| 2、学生能够理解继承与派生的意义 |
| 3、学生能够理解类之间的派生关系和复合关系。 |
| 4、学生能够掌握赋值兼容，派生类的复制构造函数和赋值运算符的原则（难点） |
| 9 | 虚函数与多态 | 实验教材 | 1、学生能够掌握基类的虚函数定义与使用。 | 4 | 设计研究性 | 必做 | 1 | 实验指导 | 课程目标2 |
| 2、学生能够理解使用虚函数和派生类实现多态（重点、难点） |
| 3、学生能够掌握使用基类指针调用派生类的虚函数。 |
| 10 | 输入输出流设计 | 实验教材 | 1、学生能够深入理解C++的输入输出含义及其实现方法 （重点、难点） | 2 | 验证性 | 必做 | 1 | 实验指导 | 课程目标2 |
| 2、学生能够掌握文本文件的输入和输出 |
| 3、学生能够掌握二进制文件的输入和输出 |
| 11 | 类属设计 | 实验教材 | 1、学生能够掌握函数模板和模板函数的用法。（重点、难点） | 2 | 设计研究性 | 选做 | 1 | 实验指导 | 课程目标3 |
| 2、学生能够掌握类模板和模板类的用法。 |
|  |  |  |  | 32 |  |  |  |  |  |

## 四、课程考核

**（一）考核内容与考核方式**

**表4-1 课程目标、考核内容与考核方式对应关系**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **课程目标** | **考核内容** | **所属**  **学习模块/项目** | **考核占比** | **考核方式** |
| 课程  目标 1 | 1、C++程序设计的基本概念和常用术语， C++程序设计的基础知识；使用C++软件开发环境； C++程序的设计、编码、调试、测试、发布等基本技能。 | 模块1-3 | 40% | 课堂表现  实验项目  阶段性测试  实验大作业 |
| 2、过程化程序结构组织，全局数据、静态数据、动态数据和名空间概念。 | 模块2-5 |
| 3、对象化程序设计方法。抽象数据类型的构造及实现方法，操作符重载。 | 模块1，7 |
| 课程  目标 2 | 1、类继承体系，多态特征 | 模块6，5，3 | 40% | 课堂表现  阶段性测试  实验项目  实验大作业 |
| 2、虚函数实现机制 | 模块1 |
| 3、面向对象程序设计方法和技术。 | 模块2 |
| 模块7 |
| 课程  目标 3 | 1、函数模板和类模板  理解并实践C++异常处理机制 | 模块1，7 | 20% | 实验项目  实验大作业 |
| 2、C++标准库和标准模板类库的使用。 | 模块1-4 |
| 合计 |  |  | 100% |  |

**表4-2 课程目标与考核方式矩阵关系**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 课程  目标 | 考核方式 | | | | 考核占比 |
| 课堂表现  9% | 实验项目12% | 阶段性测验  9% | 实验大作业70% |
| 课程目标1 | 60% | 35% | 35% | 40% | 40%=9%\*60%+12%\*35%+9%\*35%+70%\*40% |
| 课程目标2 | 0% | 35% | 25% | 48% | 40%=9%\*0%+12%\*35%+9%\*25%+70%\*48% |
| 课程目标3 | 40% | 30% | 40% | 12% | 20%=9%\*40%+12%\*30%+9%\*40%+70%\*12% |

**（二）成绩评定**

**1.平时成绩评定**

**（1）课堂表现（20 分）**：通过学生在课堂上发言、提问情况，评价学生的课程参与能力。

**（2）作业完成情况（10分）**：理论课程的阅读材料完成情况，实验课程的实验项目完成情况，主要评价学生的实际操作能力。

**（3）阶段性测验（30分）**：学生在平时测试、期中测验中综合应用课程知识的情况。

**（4）实验项目（40分）：**通过实验项目的完成情况，评价学生的基本应用能力和工程实践能力。

**2.期末成绩评定**

期末考核主要考察学生对物联网工程专业的知识体系、研究范畴和研究方向的理解与应用。通过充分的程序设计实践，掌握C++程序设计各种方法，理解抽象编程本质，理解抽象数据类型、封装、继承、多态、模板、异常等高级编程概念。学会用对象化的方法，组织和调试有数百行语句规模的多文件、多类型的程序工程，并进一步了解C++内部特性，学会标准模板库的简单使用，实质性提高操作C++开发环境和编程实现的能力。方式为开卷考试+大作业，其中大作业采用开卷考试，实验考试采用大作业。

**3.总成绩评定**

考核方式建议：大作业。

学期成绩评定 平时成绩（40%）+期末成绩（60%）。

平时成绩可以包括学生出勤情况、课堂参与情况、平时作业情况、期中测验成绩等。期中测验由课程组或上课教师自定。

**（三）评分标准**

**表4-3课程目标考核方式评分标准**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **考核项目** | **评分标准** | | | | |
| **90-100分** | **80-90分** | **70-80分** | **60-70分** | **0-60分** |
| 课堂表现 | 理论课上严格遵守课堂纪律，按要求完成课堂练习且正确率高。积极主动参与课堂讨论，讨论表述的论点正确有新意，有自己的见解，能体现良好的职业道德、职业使命感和社会责任感。  实验课不迟到早退，遵守机房管理规范、爱护实验设备，课上积极与老师互动，实验室项目按时提交。 | 理论课上遵守课堂纪律，按要求完成课堂练习且正确率比较高。经常参与课堂讨论，表述的论点正确，有一定新意，讨论能体现良好的职业道德、职业使命感和社会责任感。  实验课不迟到早退，遵守机房管理规范、爱护实验设备，课上老师互动一般，实验室项目按时提交。 | 理论课上比较遵守课堂纪律，按要求完成课堂练习且有一定正确率。较少参与课堂讨论，表述的论点基本正确，讨论基本能体现职业道德、职业使命感和社会责任感。  实验课不迟到早退，遵守机房管理规范、爱护实验设备，课上不积极与老师互动，实验室项目未按时提交。 | 理论课上比较遵守课堂纪律，按要求基本完成课堂练习且有基本正确。偶尔参与课堂讨论，表述思路一般，逻辑性不强，说服力一般。  实验课有迟到早退现象，实验室上不认真听讲，课上不积极与老师互动，实验室项目未按时提交。 | 理论课上不太遵守课堂纪律，不参与课堂讨论，未按要求完成课堂练习或正确率低。实验课缺课，实验室项目未提交 |
| 作业完成情况 | 实操项目中所采用的方法和实验步骤能够解决问题，操作正确，实验项目内容完整，程序书写规范，项目完成结果与标准结果一致。 | 实操项目中所采用的方法和实验步骤能够解决问题，操作较正确，实验项目内容较完整，程序书写，项目完成结果与标准结果较一致。 | 实操项目中所采用的方法和实验步骤基本能够解决问题，操作基本正确，实验项目内容较完整，程序书写基本规范，项目完成结果与标准结果基本一致。 | 实操项目中所采用的方法和实验步骤有一定错误，操作正确，实验项目内容不太完整，程序书写基本规范，项目完成果与标准结果有一定的差距。 | 实操项目中所采用的方法和实验步骤错误，操作不正确，实验项目内容不完整，程序书写不规范，项目完成果与标准结果差距太大。 |
| 实验项目 | 能较好利用实验过程中的原理和步骤规范书写实验报告，实验报告能完美体现实验项目的目标原理和实现过程，实验项目书写内容完整，并具有优秀的的信息素养，职业道德、职业使命感和社会责任感。 | 能利用实验过程中的原理和步骤规范书写实验报告，实验报告能较好体现实验项目的目标原理和实现过程，实验项目书写内容完整，并具有良好的信息素养，职业道德、职业使命感和社会责任感。 | 能基本利用实验过程中的原理和步骤较为规范的书写实验报告，实验报告基本能体现实验项目的目标原理和实现过程，实验项目书写内容基本完整，并具有较好的信息素养，职业道德、职业使命感和社会责任感。 | 能基本体现实验过程中的原理和步骤，按时书写并提交实验报告，实验报告基本符合实验项目的目标原理和实现过程，实验项目书写内容基本完整，并具有一定的的信息素养，职业道德、职业使命感和社会责任感。 | 不能体现实验过程中的原理和步骤，没有按时书写并提交实验报告，实验报告与实验项目的目标原理和实现过程不符合，实验项目书写内容不够完整，不能体现一定的信息素养，职业道德、职业使命感和社会责任感。 |
| 阶段性测验 | 掌握C++语言的基本知识的同时，较好的掌握C++语言的编程性技巧和上机调试程序的方法，掌握程序设计的常用算法。能够全面、深入理解和熟练掌握所学内容，并能够用其分析、设计和解决测试问题，能举一反三。 | 掌握C++语言的基本知识的同时，能够掌握C++语言的编程性技巧和上机调试程序的方法，基本掌握程序设计的常用算法。能够全面、深入理解和熟练掌握所学内容，并能够用其分析、设计和解决测试性问题，并有所启发。 | 基本掌握C++语言的基本知识，能够基本掌握C++语言的编程性技巧和上机调试程序的方法，了解程序设计的常用算法。能够理解和熟练掌握所学内容，并能够用其分析、设计和解决测试性问题。 | 基本掌握C++语言的基本知识，没有掌握C++语言的编程性技巧和上机调试程序的方法，不了解程序设计的常用算法。能勉强使用所学内容分析、设计和解决测试性问题。 | 不了解C++语言的基本知识，没有掌握C++语言的编程性技巧和上机调试程序的方法，对程序设计的常用算法一无所知。不能使用所学内容分析、设计和解决测试性问题。。 |
| 实验大作业 | 实验大作业项目有创意。能正确的使用和描述项目中使用到的数据结构与算法。能熟练使用C++代码实现项目功能，并且项目功能丰富。项目代码符合编程规范，项目文档完整、格式规范。项目工作量饱满，体现了一定的C++程序实际应用能力。 | 实验大作业项目有一定创意。能正确的使用和描述项目中使用到的数据结构与算法。能使用C++代码实现项目功能，并且项目功能基本丰富。项目代码符合编程规范，项目文档完整、格式规范。项目工作量饱满，能基本体现C++程序实际应用能力。 | 实验大作业项目中规中矩。能正确的使用和描述项目中使用到的数据结构与算法。基本能使用C++代码实现项目功能，并且项目功能较为齐全。项目代码基本符合编程规范，项目文档完整、格式规范。项目工作量符合要求，能基本体现C++程序实际应用能力。 | 实验大作业项目能按时完成。能基本正确的使用和描述项目中使用到的数据结构与算法。基本能使用C++代码实现项目功能，并且项目功能基本实现了。项目代码基本符合编程规范，项目文档能按要求完成。项目工作量基本符合要求，能基本体现C++程序实际应用能力。 | 实验大作业项目不能按时完成。不能正确的使用和描述项目中使用到的数据结构与算法。无法使用C++代码实现项目功能。项目代码混乱，无序，不符合编程规范，项目文档不能按要求完成。项目工作量明显偏少，不能基本体现C++程序实际应用能力。 |

**五、其它说明**

本课程大纲依据2023版物联网工程专业人才培养方案，由大数据与智能工程学院物联网工程系讨论制定，大数据与智能工程学院教学工作委员会审定，教务处审核批准，自2023级开始执行。