东北大学秦皇岛分校本科课程教学大纲

一、课程基本信息

<table>
<tr><td>开课单位</td><td colspan="3">计算机与通信工程学院</td><td>课程代码</td><td colspan="5">3100212006</td></tr>
<tr><td>中文名称</td><td colspan="8">C++程序设计基础</td></tr>
<tr><td>英文名称</td><td colspan="8">Fundamentals of C++ Programming</td></tr>
<tr><td>总学时</td><td>80</td><td rowspan="2">理论学时</td><td rowspan="2">56</td><td rowspan="2">实验学时</td><td rowspan="2">24</td><td rowspan="2">上机学时</td><td rowspan="2">0</td><td rowspan="2">设计学时</td><td rowspan="2">0</td><td rowspan="2">课外学时</td><td rowspan="2">0</td></tr>
<tr><td>学　分</td><td>4.5</td></tr>
<tr><td>课程类型</td><td colspan="8">☐理论课程　☑理论（含实验）课程　☐实践类课程</td></tr>
<tr><td>课程类别</td><td colspan="8">☐通识类　☑学科基础类　☐专业方向类　☐实践类</td></tr>
<tr><td>适用专业</td><td colspan="8">计算机科学与技术</td></tr>
<tr><td>先修课程</td><td colspan="8">无</td></tr>
<tr><td>建议开课学期</td><td colspan="8">☑1-1 ☐1-2 ☐2-1 ☐2-2 ☐3-1 ☐3-2
☐4-1 ☐4-2 ☐5-1 ☐5-2 ☐*（代表每学期都开设）</td></tr>
<tr><td>考核方式</td><td colspan="3">☑考试 ☐考查</td><td>成绩分制</td><td colspan="4">☑百分制 ☐五级制
☐二级制</td></tr>
<tr><td>使用教材</td><td colspan="8">1. 谭浩强，《C++程序设计》，清华大学出版社，2021 年，第四版</td></tr>
<tr><td>参考书及教学资源</td><td colspan="8">1.张冰，《面向对象程序设计 C++语言编程》，机械工业出版社，2008 年
2.钱能，《C++程序设计教程（第二版）》，清华大学出版社，2005 年
3.刁成嘉等，《C++面向对象编程基础》，机械工业出版社，2008 年
4. Paul Deitel，《C++ How to Program》，Pearson，2016 年</td></tr>
<tr><td>其他说明</td><td colspan="8">无</td></tr>
</table>

二、课程简介

中文描述

本课程为计算机科学与技术专业本科生的主干专业课，从程序数据类型和表达式、基于过程的程序设计、基于对象的程序设计、面向对象的程序设计四个方面，系统阐述了程序设计的基本过程和主要知识点。

《C++程序设计》是计算机科学与技术专业本科生的一门基础课程，它是为学生掌握当前最基本的面向对象程序设计方法而设置的。通过本课程的学习，使学生比较熟练地掌握 C++语言的语法规则、基本概念和简单算法，掌握基本的程序设计过程和技巧；具备初步的高级语言程序设计能力和编程经验，能够熟练应用集成环境进行 C++语言的编写、编译与调试，在此基础上增加面向对象的基本知识，掌握 C++输入输出流操作，以及类的基本定义和使用。

该课程培养学生应用问题分析、程序设计和编码、程序调试能力，使学生掌握必要的算法设计技能，初步掌握软件开发的思路和方法，掌握软件工程技术规范，为后继课程的学习、为各类专业应用打下坚实的基础，达到使用 C++语言工具解决应用问题的水平。

English Description

This course is the main professional course for undergraduates majoring in computer science. It systematically expounds the basic process and main knowledge points of program design from four aspects: program data type and expression, process-based program design, object-based program design and object-oriented program design.

C + + programming is a basic course for undergraduates majoring in computer science. It is designed for students to master the most basic object-oriented programming method. Through the study of this course, students can master the grammar rules, basic concepts and simple algorithms of C + + language, and master the basic programming process and skills; Have the preliminary high-level language programming ability and programming experience, can skillfully use the integrated environment for C + + language writing, compiling and debugging, on this basis, increase the basic knowledge of object-oriented, master the C + + input and output stream operation, as well as the basic definition and use of class.

This course cultivates students' abilities of application problem analysis, program design and coding, and program debugging. It enables students to master the necessary algorithm design skills, preliminarily master the ideas and methods of software development, and master the technical specifications of software engineering, so as to lay a solid foundation for the study of subsequent courses and various professional applications, and reach the level of using C + + language tools to solve application problems.

三、课程目标

1. 知识目标：

1-1：掌握 C++程序设计的初步知识，了解程序发展史和软件工程的基本概念。

1-2：掌握程序设计中三种设计结构、函数、数组、指针和用户自定义类型的使用，建立起基于过程的程序设计思想，为后续课程打下基础。

1-3：在基于过程设计基础上，进一步掌握基于对象的程序设计思想，建立类和对象的基本概念，逐步培养学生面向对象程序设计的思路。

1-4：掌握继承和多态，了解继承在软件开发中的重要意义，使得学生具有面向对象程序设计的能力。

2. 技能目标：

2-1：具备基于过程的程序设计的能力，能够根据需求设计出相应的软件程序。

2-2：具备面向对象的软件设计能力，能够设计信息管理系统。

3. 德育目标：

3-1：培养严谨的科学精神，引导学生做事要有精益求精的态度和持之以恒的耐心。

3-2：扩展学生的兴趣领域和知识面，使其将个人兴趣与国家发展相结合。

3-3：培养学生民族自豪感，继承中华民族传统美德，并将其发扬光大。

表 1 课程目标与毕业要求的对应关系

毕业要求	指标点	课程目标	关联性（H/M/L）
1、工程知识：能够将数学、自然科学、工程基础和专业知识用于解决复杂工程问题。	指标点 1-2：掌握程序设计、数据结构、算法分析与设计、计算机数字系统、操作系统等专业知识，具备计算机程序设计开发能力和计算机与信息系统设计开发与维护能力；	1-1、1-2、1-3、1-4	H
3、设计/开发解决方案：能够设计针对复杂工程问题的解决方案，设计满足特定需求的系统、单元或流程，并能够在设计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。	指标点 3-1：能够设计针对本专业相关复杂工程问题的解决方案，能够设计和开发实现特定功能、满足特定需求的计算机、软件或网络系统；	2-1、2-2	H
5、使用现代工具：能够针对复杂工程问题，开发、选择与使用恰当的技术、资源、现代工程工具和信息技术工具，包括对复杂工程问题的预测与模拟，并能够理解其局限性。	指标点 5-1：能够对本专业相关复杂工程问题进行建模与分析，理解获取相关信息参数的必要性与基本方法，并理解其局限性；	2-1、2-2、3-1、3-2、3-3	M

四、主要教学内容

（一）理论教学

第 1 章 C++初步知识 （支撑教学目标 1-1、3-1）

【1】知识点：

(1) 从 C 到 C＋＋；最简单的 C＋＋程序；

(2) C＋＋程序的构成和书写形式；

(3) C＋＋程序的编写和实现；关于 C＋＋上机实践。

【2】学习目标：

了解： C++程序设计的基本内容。

【3】重点内容：

C++程序的构成和书写形式；理解程序的开发过程。

【4】难点内容：

C++程序的构成和书写形式。

【5】思政案例：

在讲解简单的 C++程序时，告诉学生，即使是一个很简单的小错误，甚至是一个标点符号的错误，都可能导致整个程序无法正常运行，所以在写程序时一定要严谨，让学生通过在程序设计、调试中遇到的各种问题，培养学生严谨，一丝不苟，细心精益求精的态度。

第 2 章 数据类型与表达式（支撑教学目标 1-1、2-1）

【1】知识点：

(1) C＋＋的数据类型、常量、变量；

(2) C＋＋的运算符：算术运算符、算术表达式、赋值运算符与赋值表达式；逗号运算符与逗号表达式。

【2】学习目标：

了解： 数据类型的概念及其分类。

理解： 常量的概念及其表现形式：数值常量、字符常量、符号常量；变量的概念及其表现形式。

掌握： 算术运算符、算术表达式、赋值运算符和逗号运算符的使用。

【3】重点内容：

变量和常量定义；表达式；运算符的结合性和优先级。

【4】难点内容：

++；--运算符的前缀和后缀。

第 3 章 程序设计初步（支撑教学目标 1-2、2-1）

【1】知识点：

(1) 面向过程的程序设计和算法；

(2) C＋＋程序设计语句和赋值语句；

(3) C＋＋的输入与输出；

(4) 顺序结构程序的编写基本思路；

(5) 关系运算和逻辑运算；

(6) 选择结构和 if 语句；

(7) 循环结构和循环语句。

【2】学习目标：

了解： 算法的概念和算法的应用背景。

理解： 简单 I/O 流控制语句，理解关系运算符（<、<=、>、>=、==、！=）、逻辑运算符（||、&&、！）。

掌握： 选择结构（包括 if 语句的使用、if 语句的嵌套、条件表达式、switch 语句）；循环结构（包括 while 语句、do－while 语句、for 语句）的程序设计方法。

【3】重点内容：

(1) C++程序和语句；

(2) 数据的输入与输出（包括 getchar 函数、putchar 函数、scanf 函数、printf 函数）；

(3) 程序流程的控制结构。

【4】难点内容：

程序流程的控制结构（break 语句和 continue 语句）。

第 4 章 利用函数实现指定的功能（支撑教学目标 1-2、2-1）

【1】知识点：

(1) 函数的概念；

(2) 定义函数的一般形式；

(3) 函数参数和函数的值；

(4) 函数的调用、嵌套调用、递归调用；

(5) 内置函数；

(6) 局部变量和全局变量；

(7) 变量的存储类别和变量属性小结；

(8) 关于变量的声明和定义；

(9) 内部函数、外部函数和头文件。

【2】学习目标：

了解： 函数的概念和形式（包括无参函数、有参函数、函数定义的一般形式）；函数调用的机制；变量的存储类别和变量属性。

理解： 函数参数和函数的值、内部函数、外部函数和头文件。

掌握： 函数的调用、嵌套调用、递归调用；内置函数；局部变量和全局变量。

【3】重点内容：

(1) 函数的定义及调用；

(2) 变量的作用域。

【4】难点内容：

(1) 静态局部变量在程序中的使用方法；

(2) 递归函数和重载函数等特殊函数用法。

第 5 章 数组（支撑教学目标 1-2、2-1）

【1】知识点：

(1) 数组的概念；

(2) 定义和引用一维数组；

(3) 定义和引用二维数组；

(4) 用数组作函数参数；

(5) 字符数组；

(6) C++处理字符串的方法--字符串类与字符串变量。

【2】学习目标：

了解： 数组的概念。

理解： 定义和引用一维数组，定义和引用二维数组。

掌握： 用数组作函数参数（包括用数组元素作函数实参、用数组名作函数参数）；字符数组（包括字符数组的定义和初始化、字符数组的赋值与引用、字符串和字符串结束标志）；字符串类与字符串变量。

【3】重点内容：

(1) 一维和二维数组的操作及应用；

(2) 字符串的处理（包括 strcat 函数、strcpy 函数、strcmp 函数和 strlen 函数再程序中的使用方法）。

【4】难点内容：

(1) 字符数组；

(2) 字符串类与字符串变量。

第 6 章 指针与引用（支撑教学目标 1-2、2-1）

【1】知识点：

(1) 指针的概念；

(2) 变量与指针；

(3) 数组与指针；

(4) 字符串与指针；

(5) 函数与指针；

(6) 返回指针值的函数；

(7) 指针数组和指向指针的指针；

(8) const 指针和 void 指针类型；

(9) 指针的数据类型和指针运算；

(10) 引用。

【2】学习目标：

了解： 指针数组和指向指针的指针。

理解： 返回指针值的函数；const 指针；void 指针类型；数据类型和指针运算。

掌握： 指针的概念；变量与指针（包括取地址运算符、 指针运算符）；数组与指针（包括指向数组元素的指针和用指针变量作函数参数接收数组地址）；字符串与指针（包括：用字符数组存放一个字符串、用字符串变量存放字符串、用字符指针指向一个字符串）；函数与指针（包括用函数指针变量调用函数、用指向函数的指针作函数参数）；引用。

【3】重点内容：

(1) 指针的概念；

(2) 各类指针的基本应：变量指针；数组指针；字符串指针；函数指针、const 指针、void 指针。

【4】难点内容：

(1) 数组与指针；

(2) 函数与指针和指针数组。

第 7 章 用户自定义数据类型（支撑教学目标 1-2、2-1）

【1】知识点：

(1) 结构体类型；

(2) 共用体类型；

(3) 枚举类型；

(4) 用 typedef 声明新的类型名。

【2】学习目标：

了解： 用 typedef 声明新的类型名。

理解： 共用体类型；枚举类型。

掌握： 结构体类型：结构体定义方法、结构体初始化和引用结构体变量。

【3】重点内容：

(1) 结构体及应用：结构体数组、指向结构体变量的指针、结构体类型数据作为函数参数。

(2) 枚举类型。

【4】难点内容：

(1) 结构的指针用法。

第 8 章 类和对象的特性（支撑教学目标 1-3、2-2）

【1】知识点：

(1) 面向对象程序设计方法概念；

(2) 类的声明和对象的定义；

(3) 类的成员函数；

(4) 对象成员的引用；

(5) 类的封装性和信息隐蔽；

(6) 类和对象的应用。

【2】学习目标：

了解： 面向对象程序设计方法概念。

理解： 对象成员的引用（包括通过对象名和成员运算符访问对象中的成员、通过指向对象的指针访问对象中的成员、通过对象的引用变量来访问对象中的成员）；类和对象的应用。

掌握： 类的声明和对象的定义；类的成员函数（包括成员函数的性质、在类外定义成员函数、inline 成员函数、成员函数的存储方式）；类的封装性和信息隐蔽。

【3】重点内容：

(1) 类的声明和对象的定义；

(2) 类的成员函数；

(3) 类的封装性和信息隐蔽指针的概念。

【4】难点内容：

类的封装性和信息隐蔽。

第 9 章 使用类和对象（支撑教学目标 1-3、2-2）

【1】知识点：

(1) 利用构造函数对类对象进行初始化；

(2) 析构函数；

(3) 调用构造函数和析构函数的顺序；

(4) 对象数组；

(5) 对象指针；

(6) 共用数据的保护；

(7) 对象的动态建立和释放；

(8) 对象的赋值和复制；

(9) 静态成员；

(10) 友元；

(11) 类模板指针的概念。

【2】学习目标：

了解： 析构函数；共用数据的保护；类模板指针的概念。

理解： 调用构造函数和析构函数的顺序；友元。

掌握： 构造函数（包括使用默认参数的构造函数、构造函数的重载），对象数组，对象指针（包括指向对象的指针、指向对象成员的指针、this 指针），对象的动态建立和释放，对象的赋值和复制，静态成员（包括静态数据成员、静态成员函数）、类模板。

【3】重点内容：

(1) 构造函数；

(2) 对象数组；

(3) 对象指针；

(4) 对象的动态建立和释放；

(5) 对象的赋值和复制；

(6) 静态成员；

(7) 类模板。

【4】难点内容：

(1) 对象数组；

(2) 对象指针；

(3) 对象的动态建立和释放；

(4) 类模板。

第 10 章 运算符重载（支撑教学目标 1-3、2-2、3-2）

【1】知识点：

(1) 运算符重载的概念和方法；

(2) 重载运算符的规则；

(3) 运算符重载函数作为类成员函数和友元函数；

(4) 重载双目运算符；

(5) 重载单目运算符；

(6) 重载流插入运算符和流提取运算符；

(7) 不同类型数据间的转换。

【2】学习目标：

了解： 运算符重载函数作为类成员函数和友元函数。

理解： 重载运算符的规则；重载单目运算符；不同类型数据间的转换（包括标准类型数据间的转换、转换构造函数和类型转换函数）。

掌握： 运算符重载的概念和方法；重载双目运算符；重载流插入运算符和流提取运算符。

【3】重点内容：

运算符重载的概念和方法。

【4】难点内容：

运算符重载的概念和方法。

【5】思政案例：

在讲解运算符重载时，通过运算符重载的讲解，告诉学生同一个运算符，通过重载可以进行不同的运算，对学生进行创新思想的教育，从而学培养学生的创新精神，创新是一个民族的灵魂。教师引导学生要想创新首先丰富自己的知识，多方面挖掘自己的能力，扩大自己的兴趣领域，这样才能更快更轻松的融入社会，满足社会和国家发展的需求，充分体现个人价值，并将个人价值和国家发展相结合，使个人发展和国家需求统一结合。

第 11 章 继承与派生（支撑教学目标 1-4、2-2、）

【1】知识点：

(1) 继承与派生的概念；

(2) 派生类的声明方式；

(3) 派生类的构成；

(4) 派生类成员的访问属性；

(5) 派生类的构造函数和析构函数；

(6) 多重继承；

(7) 基类与派生类的转换；

(8) 继承与组合。

【2】学习目标：

了解： 派生类的构成；基类与派生类的转换；继承与组合。

理解： 派生类的构造函数和析构函数；多重继承。

掌握： 继承与派生的概念；派生类的声明方式；派生类成员的访问属性（包括公用继承、私有继承、保护继承）。

【3】重点内容：

继承与派生的概念和方法。

【4】难点内容：

(1) 多重继承：多重继承过程中每个成员的访问属性变化。

(2) 基类与派生类的转换。

【5】思政案例：

在讲解继承和派生知识点时，通过对基层和派生的讲解，让学生知道在编写程序时可以通过继承的方式重复利用原来已经写好的代码，进而告诉学生，中华民族有很多优秀的传统美德，值

得我们继承和发扬，映射出民族自豪感。

第 12 章 多态性与虚函数（支撑教学目标 1-4、2-2）

【1】知识点：

(1) 多态性的概念；

(2) 虚函数；

(3) 纯虚函数与抽象类。

【2】学习目标：

了解： 纯虚函数与抽象类。

理解： 虚函数。

掌握： 多态性的概念。

【3】重点内容：

多态性的概念。

【4】难点内容：

虚函数。

第 13 章 输入输出流（支撑教学目标 1-4、2-2）

【1】知识点：

(1) C++的输入和输出；

(2) 标准输出流；

(3) 标准输入流；

(4) 文件操作与文件流；

(5) 字符串流。

【2】学习目标：

了解： 字符串流。

理解： 标准输出流（包括 cout，cerr 和 clog 流）；标准输入流。

掌握： C++的输入和输出；文件操作与文件流（包括文件的打开与关闭、ASCII 文件的操作、文件操作与文件流、字符串流）。

【3】重点内容：

C++的输入和输出。

【4】难点内容：

文件操作与文件流。

（二）实验教学

实验教学的项目名称及学时分配相关信息如表 2 所示。

表 2 实验教学信息表

序号	实验项目名称	实验学时	每组人数	实验类型	实验性质
1	C++程序中的函数与预处理	4	1	设计性	必做
2	数组和指针的使用	4	1	设计性	必做
3	自定义数据类型的设计使用	2	1	设计性	必做
4	继承与多重继承	4	1	设计性	必做
5	运算符重载的实现	2	1	设计性	必做
6	类模板的使用	4	1	设计性	必做
7	多态性的实现	2	1	设计性	必做
8	文件操作与文件流	2	1	设计性	必做

实验项目一：C++程序中的函数与预处理（支撑教学目标支撑教学目标 1-1、1-2、2-1、3-1、3-2）

【1】实验内容：

熟悉编程环境、掌握基本的数据类型和表达式定义方法；C++程序设计初步、输入/输出、三种基本结构的应用实践；函数的定义与实践。

【2】实验目的：

掌握编程环境，理解 C++程序编译执行的过程；掌握 C++程序设计和调试方法；掌握基本的数据类型和表达式定义方法；掌握输入/输出、三种基本结构定义方法；掌握实现函数定义与调用方法。

【3】教学要求：

了解编程环境，掌握基本的数据类型和表达式定义方法；掌握 C++程序设计和调试方法；掌握 C++中的标准输入/输出和三种基本结构定义方法；掌握 C++程序的编译和、调试和执行过程，理解背后的原理；掌握函数定义和调用方法，理解函数原型声明。

【4】实验方式：程序设计

【5】实验场地与设备：综合楼 1108、1208；一体化实验教学平台、计算机。

【6】思政案例：

在讲解编写计算机程序时候的规范性、高效性等时，让同学们知道计算机程序在我们日常生活中无处不在，而且好的程序会让使用者感到非常友好。因此，同学们要养成规范的编程习惯，培养严谨的科学精神，引导学生做事要有精益求精的态度和持之以恒的耐心；在讲解编写的程序为什么能执行的时候，引入程序执行背后的编译系统、操作系统和计算机组成原理的知识，让同学们了解到一行行代码背后的工作过程，从而拓展学生的兴趣领域和知识面，让同学们知道扎扎实实学好这些专业知识可以迈进芯片设计开发的大门，从而为我国芯片卡脖子问题贡献力量，让个人所学与国家发展相结合，培养学生的爱国情怀。

在讲解编写 C++程序时，学生最怕的就是程序编译出错和执行结果不正确的提示。大一的学生大多都是初学 C++编程，过多的错误提示且是英文的提示语，会导致很多同学产生畏难心理。

代码无法正常运行，无法得到想要的结果，却难以解决，长此以往，会影响学生学好这门基础性很强的课程的信心，逐渐丧失学好实践性较强课程的兴趣，甚至放弃课程的学习。因此，通过水滴石穿的经典小故事引导学生列举类似的名言警句，比如世上无难事，只要肯攀登；高尔基说的钢铁是怎样炼成的；马克思说的万事开头难，每门科学都是如此……，激励同学们面对困难不气馁，只要认真学、多动手练习、一步一个脚印，终能学有所成。

实验项目二：数组和指针的使用（支撑教学目标支撑教学目标 1-1、1-2、2-1）

【1】实验内容：

掌握数组的使用，数组与函数调用的关系；掌握指针的使用，指针与函数、数组的关系。

【2】实验目的：

掌握数组的使用，数组与函数调用的关系；掌握指针的使用，指针与函数、数组的关系。

【3】教学要求：

掌握数组的使用，理解数组与函数调用关系，了解指针的定义、作用。掌握指针的使用，指针与函数、数组的关系。

【4】实验方式：程序设计

【5】实验场地与设备：综合楼 1108、1208；一体化实验教学平台、计算机。

【6】思政案例：

设计使用数组自动构造杨辉三角的题目，给同学们讲解杨辉三角是南宋杰出数学家杨辉所著的《详解九章算法》中提出的，并讲解杨辉三角的样式，分析模型特点，确定数组结构，再到发现递推规律，确定推演公式，核心代码设计。使学生知道杨辉是中国宋代著名的数学家，杨辉三角领先于法国数学家帕斯卡近 400 年，这是我国数学史上伟大的成就。通过对杨辉三角起源认知（类似的案例都可以），激发学生的爱国热情和民族自豪感，同时也让学生树立坚定的信念，向科学家学习，成长为思想政治可靠、专业技术优秀的建设人才。

职业素养是人类在社会活动中需要遵守的行为规范。职业道德、职业思想、职业行为习惯是职业素养中最根基的部分。引导学生在实验过程中要注意编写程序的格式、变量与方法的命名方式、合理添加注释、合理规划程序工程文件，这些都是合格的软件开发从业人员的基本素质。引导学生注重职业道德，尊重他人的知识产权。在校学习未经同学允许，不随意拷贝他人的实验成果；工作后，不偷窥和复制同事的开发成果，不盗取公司的研发成果。使学生明确：职业素养是一个职业人的立身之本，在学生时代，就应该不断提升个人修养和思想道德和水平，着力锻造良好的职业素养。

实验项目三：自定义数据类型（支撑教学目标支撑教学目标 1-1、1-2、2-1）

【1】实验内容：

掌握结构体定义方法与应用。

【2】实验目的：

掌握结构体定义方法与应用。

【3】教学要求：

了解 typedef 的用法。掌握结构体定义方法。

【4】实验方式：程序设计

【5】实验场地与设备：综合楼 1108、1208；一体化实验教学平台、计算机。

【6】思政案例：

强调 C++程序设计实验课程的特点--每一节课之间的知识要点都是紧密联系的，需要学生一步步把基本知识掌握扎实，后面的内容就可融会贯通。自定义数据类型是利用之前所学的基本数据类型来创造出新的类型。这种创新不是凭空创造，而是厚积薄发的创造。从这个课程特点可以引入到思政课内容：做人也必须脚踏实地，一步一个脚印去提高自己，切不可三天打鱼两天晒网。要想成为一个德才兼备具有创新能力的大学生，就必须踏踏实实、不存疑、不懈怠的、扎扎实实学好每一个知识点和每一门课程，虽然过程比较辛苦，但是最后的结果往往会出乎意料。

实验项目四：继承与多重继承（支撑教学目标支撑教学目标 1-1、1-2、1-3、1-4、2-1、2-2）

【1】实验内容：

掌握类的定义方法、对象和构造函数的使用；掌握继承与派生的基本方式；派生类对象的构造与析构；派生类与基类之间的类型转换以及组合类的应用；掌握多重继承。

【2】实验目的：

掌握类的定义方法、对象和构造函数的使用；掌握并灵活使用 C++中类的继承与派生的基本用法，掌握多重继承的思想和用法；初步了解继承与派生方法在模块化程序设计中的作用。

【3】教学要求：

理解类的概念、特性；掌握类的定义方法、对象和构造函数的使用；掌握派生类的声明方式和派生类成员的访问属性，编写程序实现从基类到派生类的多重继承定义。

【4】实验方式：程序设计

【5】实验场地与设备：综合楼 1108、1208；一体化实验教学平台、计算机。

【6】思政案例：

今天的社会无论什么行业想要做出一番成就，单打独斗已经不现实了。所谓人多力量大，三个臭皮匠赛过诸葛亮，类的设计就类似于物以类聚，团结起来可以做更多的事。团队合作显得很重要，社会分工可以促进生产力的发展，一个开发团队做好分工就可以很好地完成任务，提高效率。一个好的、高效的团队，它的队员需要做到：协作、宽容、欣赏、信任、沟通、分享。分别就每个点让同学们讨论，从而理解团结协作的意义！。

在讲解继承与派生的知识点时，通过对相关概念的讲解，以及基本例程的展示，使学生能够了解到 C++程序语言的代码复用机制，指导学生总结之前写过的可以正确运行的代码和现有的方法模板，进而告诉学生继承与派生就是要总结经验，不断进步。不仅要继承前人的正确发展经验，同时还要锐意进取，开拓创新，在已有的基础上不断完善和创造，培养学生的创新精神。

实验项目五：运算符重载的实现（支撑教学目标支撑教学目标 1-1、1-2、1-3、2-1）

【1】实验内容：

掌握运算符重载方法。

【2】实验目的：

掌握运算符重载方法。

【3】教学要求：

理解运算符重载的作用。掌握运算符重载方法。

【4】实验方式：程序设计

【5】实验场地与设备：综合楼 1108、1208；一体化实验教学平台、计算机。

实验项目六：函数模板和类模板的使用（支撑教学目标支撑教学目标 1-1、2-1、3-1、3-2）

【1】实验内容：

掌握函数模板和类模板的形式和使用方法。

【2】实验目的：

掌握函数模板和类模板的形式和使用方法。

【3】教学要求：

掌握函数模板和类模板的作用；掌握函数模板和类模板的使用方法。

【4】实验方式：程序设计

【5】实验场地与设备：综合楼 1108、1208；一体化实验教学平台、计算机。

实验项目七：多态机制的实现（支撑教学目标支撑教学目标 1-1、1-2、1-3、2-1、3-1、3-2）

【1】实验内容：

掌握虚函数、纯虚函数的概念和多态机制的实现。

【2】实验目的：

掌握虚函数、纯虚函数的概念和多态机制的实现。

【3】教学要求：

掌握虚函数、纯虚函数的概念。理解并会使用多态。

【4】实验方式：程序设计

【5】实验场地与设备：综合楼 1108、1208；一体化实验教学平台、计算机。

【6】思政案例：

在讲解继承与派生的知识点时，通过对相关概念的讲解，以及基本例程的展示，使学生能够了解到 C++程序语言的代码复用机制，指导学生总结之前写过的可以正确运行的代码和现有的方法模板，进而告诉学生继承与派生就是要总结经验，不断进步。不仅要继承前人的正确发展经验，同时还要锐意进取，开拓创新，在已有的基础上不断完善和创造，培养学生的创新精神。

实验项目八：文件操作与文件流（支撑教学目标支撑教学目标 1-1、1-2、1-3、2-1、3-1、3-2）

【1】实验内容：

掌握输入输出流、文件的操作方法，包括文件的打开与关闭、ASCII 文件的操作、文件操作与文件流、字符串流等。

【2】实验目的：

掌握输入输出流、文件的使用。

【3】教学要求：

正确使用 I/O 流对象编程实现对磁盘文件的打开、编辑和写入、读取。

【4】实验方式：程序设计

【5】实验场地与设备：综合楼 1108、1208；一体化实验教学平台、计算机。

【6】思政案例：

在讲解知识点过程中，介绍我国在文件编辑、汉字系统、输入输出的软件发展历程，以及求伯君、王江民等我国著名 IT 前辈的奋斗史，进而激励学生的民族自豪感和自强不息的奋斗精神。党的十八届五中全会通过的“十三五”规划《建议》，明确提出实施网络强国战略以及与之密切相关的“互联网+”行动计划。国家正着力实现关键技术自主可控，为维护国家安全、网络安全提供技术保障。中国信息化需求巨大，但在一些关键技术领域如操作系统、芯片技术、CPU 技术等方面，还难以做到自主可控，对国家安全造成威胁。引导学生明确：建设网络强国，不仅仅是靠网络技术，还要有软件技术等其他各类技术的支撑。引导学生作为计算机类专业的一员，应更加明晰专业人才的培养目标，更加明确专业领域内工作岗位和工作内容的社会价值，自觉树立远大职业理想，将职业生涯、职业发展脉络与国家发展的历史进程融合起来。

五、教学安排与学时分配

本课程以课堂教学为主，辅以实验教学形式。课堂教学围绕本课程的基本概念、基本理论、基本原理和实践操作进行授课。课程围绕立德树人根本任务，以培养德智体美劳全面发展社会主义建设者和接班人为总目标，以 C++语言的语法规则、基本概念和简单算法为基础，逐步展开，深入讲解高级语言程序设计能力和编程经验。课程充分利用线下课堂、线上教学资源和思政案例开展教学活动，发挥理论知识传授、实验实践操作及价值引领作用，厚植家国情怀和强国理念，扎实基础知识，提升实践技能和创新精神，树立正确的人生观、世界观和价值观。线下面授课堂的学时分配如表 3 所示。

表 3 学时分配表

讲课内容		学时				
		讲课	实验	上机	设计	课外
第 1 章	C++的初步知识	2		0	0	0
第 2 章	数据类型与表达式	4	1	0	0	0
第 3 章	程序设计初步	6	1	0	0	0
第 4 章	利用函数实现指定的功能	4	2	0	0	0
第 5 章	数组	4	2	0	0	0

第 6 章	指针与引用	6	2	0	0	0
第 7 章	用户自定义数据类型	4	2	0	0	0
第 8 章	类和对象的特性	2	1	0	0	0
第 9 章	使用类和对象	8	3	0	0	0
第 10 章	运算符重载	4	2	0	0	0
第 11 章	继承与派生	6	4	0	0	0
第 12 章	多态性与虚函数	2	2	0	0	0
第 13 章	输入输出流	4	2	0	0	0
小计		56	24	0	0	0
总计	80					

六、教学方法

1、讲授基本原理，理论联系实际，强调程序设计过程中的基本原理，培养学生创新能力。

2、采用多媒体课件、网络教学和传统教学相结合的教学方式，赋予课堂生机。

3、理论教学、讨论形式相结合，调动学生的学习积极性，培养学生的自学能力、团队协作能力和解决实际问题的能力。

七、成绩评定

1、本课程考试形式为闭卷，按百分制评定。结合平时作业、课堂提问、课堂交流和期末考试成绩综合评定成绩。

（1）平时成绩占 30%，主要包括课堂交流、作业、线上资料学习等综合评定。

（2）实验成绩占 10%，主要为上机实验练习、作业和测评。

（3）期末成绩占 60%，主要为试卷成绩（闭卷测试）。

2、所缺实验、作业等超过应完成量 1/3 者，或缺课学时超过计划学时 1/3 者，或实验考核成绩不合格者，取消该课程考核资格。

八、评价及持续改进

1、教学大纲

考核周期：4 年，修订周期：4 年。

改进措施：课程负责人组织课程团队所有教师讨论后提出改进意见，经教研室主任审核后由教学院长批准。

2、成绩评定考核

考核周期：1 学年，评价依据：学生最终成绩综合分析。

改进措施：根据学生最终成绩的分布，进行相应的内容调整和优化，并改进教学方法。

撰写：　　李国瑞、吕艳霞

审核：　　袁晓铭

批准：　　史闻博

日期：　　2023.07.05