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">悉尼智能科技学院</td><td>课程代码</td><td colspan="5">3060512007</td></tr>
<tr><td>中文名称</td><td colspan="8">C++语言程序设计(双语)</td></tr>
<tr><td>英文名称</td><td colspan="8">C++LanguageProgramming</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">应用统计学</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、 张建波、王子健，《C++语言程序设计》，自编（未出版）。
2、 张建波、王子健，《C++语言程序设计实验指导书》，自编（未出版）。</td></tr>
<tr><td>参考书及教学资源</td><td colspan="8">1、Bjarne Stroustrup, A Tour of C++, 电子工业出版社，2023 年
2、Ivor Horton, Peter Van Weert，Beginning C++20: From Novice to Professional, 清华大学版社，2022 年
3、Y. Daniel Liang, Introduction to Programming with C++, 机械工业出版社, 2015 年
4、PaulDeitel, Harvey Deitel, C++ How to Program，电子工业出版社，2019 年
5、郑莉等，C++语言程序设计（英文版），清华大学出版社，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

本课程为应用统计学本科专业的基础课，是大学期间学的第一门计算机编程语言类课程。本课程全面、系统地介绍了 C++语言的基本特征和基础语法，主要包括数据类型与表达式、控制结构、函数、数相、类和对象、模板、继承与派生、文件操作等内容。通过该课程学习，为学生能够使用一种开发工具进行软件开发，为学生后续专业课程的学习、创新创业活动、课程设计和毕业设计的开展奠定坚实基础。

English Description

C++ Language Programming is a foundational course for undergraduates majoringinApplied Statistics, and is the first computer programming language studied during the university years. This course provides a comprehensive and systematic introduction to the basic features and syntax of C++ language, including data types and expressions, control structures, functions, arrays, classes and objects, templates, inheritance and derivation, file operations, and soon. AfterC++LanguageProgramminglearning, students will be able to start programming in C++ language, which lays a solid foundation for their subsequent professional course learning, innovation and entrepreneurship activities, course design, and graduation design, etc.

三、课程目标

1、知识目标：

1-1：掌握编程的一般过程，了解 C++程序的基本构成。

1-2：理解数据的描述，掌握简单数据的基本运算，掌握几种简单的用户自定义类型。

1-3：掌握数据的输入和输出，程序的控制结构。

1-4：理解函数的作用，掌握函数的定义与调用，了解系统函数库的组织与调用。

1-5：理解数组的概念，了解数组的作用，掌握数组定义与使用。

1-6：理解字符串的概念，了解字符串的存储机制，掌握字符串的定义与使用。

1-7：了解面向对象的基本特征，理解类和对象的概念，掌握类的定义、并用类创建对象。

1-8：掌握作用域与可变性、对象的声明周期、静态成员、友元和数据的共享与保护。

1-9：理解编译预处理和多文件结构的实现。

1-10：理解指针的概念，了解指针的作用，掌指针的定义与使用。

1-11：掌握动态内存分配方法，理解深拷贝和浅拷贝。

1-12：理解模板的概念，能实现函数模版和类模版。

1-13：理解代码重用的思想，掌握继承的概念，能实现类的继承与派生。

1-14：了解多态的概念和类别，掌握多态的实现。

2、能力目标：

2-1：会使用集成开发环境，能够编辑、编译并运行程序。

2-2：能完成简单的输入和输出，能找出简单的错误并做出修改。

2-3：能用控制结构完成简单的科学运算任务。

2-4：能编写并调用函数。

2-5：能定义、初始化数组，并输入输出数组元素，会把数组作为函数参数完成函数调用。

2-6：理解字符串和字符数组的区别与联系，能使用 string 类处理字符串。

2-7：能声明类，并能用类定义对象，能通过对象调用成员函数。

2-8：能用 const 进行数据的共享与保护。

2-9：会用指针处理单个数据和数组中的数据，能用指针作为函数参数完成函数调用，能用指针完成对象的相关操作。

2-10：能用函数模板处理泛化问题。

3、德育/素养目标：

3-1：培养具有不畏困难、锲而不舍、敢于尝试、不惧失败、迎难而上的精神，并在学习过程中培养自己的细心和耐心的勇气和精神。

3-2：培养服务意识，具有“以人为本”的服务精神。

3-3：培养遵守法律、懂规则、讲规则的新时代公民。

3-4：了解主要矛盾和次要矛盾，在面对复杂问题的时候要实事求是、抓住主要矛盾。

3-5：培养有条理和计划，做到心中有数、有条不紊、循序渐进地完成一项工作。

3-6：面向人生抉择时要慎重思考，选择能顺应时代发展的、同时也适合自己的方向。

3-7：培养具有不怕吃苦、从点滴做起，注重积累的能力。

3-8：培养学生积极向上，能与优秀的人学习和交流，努力让自己变得更加优秀。

3-9：培养学生具有分而治之思维模式，具有要有大局意识，具有把握全局、统筹规划、团队合作和责任担当意识。

3-10：守正与创新：既与中华民族几千年来坚守正道、革故鼎新的文化传统相承接，又与我们党一贯坚持的解放思想、实事求是、与时俱进、求真务实的理论品格相贯通。

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 (H/M/L)

3．设计/开发解决方案：能够设计针对复杂实际问题的解决方案，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。	3.2 能够对不同设计方案进行比较和优化，在工作各环节中具有创新意识和批判意识，善于发现、分析、系统表述和解决实际问题。	1-1 到 1-14 2-1 到 2-10 3-1 到 3-10	H
	3.3 能够在设计和开发的各个环节中综合考虑社会、健康、安全、法律、文化以及环境等因素。	1-1 到 1-14 2-1 到 2-10 3-1 到 3-10	L
4．研究：能够基于科学原理并采用科学方法对复杂实际问题进行研究，包括设计实验、分析与解释数据、并通过信息综合得到合理有效的结论。	4.2 能够结合本专业知识对实验数据进行分析与解释，设计并优化实验方案，并通过信息综合得到合理有效的结论。	1-1 到 1-14 2-1 到 2-10	H
5．使用现代工具：能够针对复杂实际问题，开发、选择与使用恰当的技术、资源、现代信息技术工具，包括对复杂实际问题的预测与模拟，并能够理解其局限性。	5.2 熟悉解决本专业相关复杂实际问题所需的技术和资源，能够运用现代信息技术进行文献检索和资料查询，获取专业解决方案。	1-1 到 1-14 2-1 到 2-10	M
	5.3 能够针对本专业相关复杂实际问题，选择与使用恰当的技术、资源、现代信息技术工具。	1-1 到 1-14 2-1 到 2-10	M

四、主要教学内容

（一）理论教学

第 1 章：Introduction（支撑课程目标 1-1、2-1、2-2、3-1、3-2、3-3、3-4）

【1】知识点：

(1) The development and history of programing language.

(2) The characters of object-oriented languages.

(3) The development procedure of programming.

(4) The frame of C++ program.

(5) The programming style and documentation.

(6) Programming errors.

【2】学习目标：

理解：The development and history of programing language; The characters of object-oriented languages; The development procedure of programming.

掌握：The frame of C++ program; The programming style and documentation; Programming errors.

【3】重点内容：

(1) The frame of C++ program.

(2) The programming style and documentation.

(3) Programming errors.

【4】难点内容：

(1) The frame of C++ program.

(2) The programming style and documentation.

(3) Programming errors.

【5】思政案例：

(1) In recent years, the C++language has consistently ranked among the top ten global programming languages, and learning it well is of great significance. However, beginners may find it difficult to get started and find it quite boring. I hope everyone has the courage to learn without fear of difficulties and persevere.

(2) The code style should be well-organized, and if you should go back, go back. Code is not only used to tell computers how to run, but also to show people, which is "people-oriented".

(3) Similar to human language, computer language also has syntax, semantics, nouns, polysemous words, etc. Unlike human language, humans have a certain ability to correct errors when understanding human language. However, computer languages such as C++have strict grammar rules, so we hope everyone has a sense of rules, ranging from complying with the rules and regulations of the class and school to complying with national laws and regulations.

第 2 章：Elementary of C++ Programming（支撑课程目标 1-2、2-1、2-2、3-4）

【1】知识点：

(1) The characters of C++ language.

(2) Primitive data types.

(3) Data representations.

(4) Operators and expressions.

(5) Operator precedence and associativity.

(6) Data type conversions.

(7) User-defined data type.

【2】学习目标：

了解：The characters of C++ language.

理解：Operator precedence and associativity; Data type conversions; User-defined data type.

掌握：Primitive data types; Data representations; Operators and expressions.

【3】重点内容：

(1) Primitive data types.

(2) Data representations.

(3) Operators and expressions.

(4) Operator precedence and associativity.

(5) Data type conversions.

【4】难点内容：

(1) Data representations.

(2) Operators and expressions.

(3) Operator precedence and associativity.

(4) Data type conversions.

【5】思政案例：

Operators in C++have priority, just like mathematical calculations. In fact, doing things is also like this, especially when many things are pressed together, one should learn to analyze and plan, grasp the main contradiction, and first do important and urgent things.

第 3 章：Statements（支撑课程目标 1-3、2-2、2-3、3-4、3-5、3-6、3-7）

【1】知识点：

(1) Input and Output Statements.

(2) Control Statements.

【2】学习目标：

掌握：Input and Output Statements; Control Statements.

【3】重点内容：

(1) Input and Output Statements.

(2) Control Statements.

【4】难点内容：

(1) Input and Output Statements.

(2) Control Statements.

【5】思政案例：

(1) Sequential structure refers to the sequential execution of one sentence (simple sentence, complex sentence) in a program. This is similar to doing things in our daily work. In order to complete a task, we should think about what to do first and then what to do. In other words, there should be regulations and plans for doing things, so that we can have a clear understanding, orderly and gradual progress in learning or work.

(2) Selection structure refers to the process of selecting and executing different statements based on the current state when a program reaches a certain step. Our growth is similar, and we often encounter various choices on the path of life. We should choose carefully, have the correct values, and not make choices that go against our conscience and morality. We should also be responsible for the choices we make.

(3) A loop structure refers to a code block (one or more statements) that needs to be executed repeatedly. In normal times, the seemingly mundane daily study life may lead to significant differences in learning outcomes after four years, so it is important to start accumulating knowledge for a better future from now on.

第 4 章：Functions（支撑课程目标 1-4、2-4、3-9）

【1】知识点：

(1) Defining a Function.
(2) Calling a Function.
(3) Function Prototype.
(4) *void* Functions.
(5) Other calling modes of function.
(6) Passing Parameters between Functions.
(7) Inline Functions.
(8) Default Formal Parameters Functions.
(9) Overloading Functions.

【2】学习目标：

理解：Function Prototype; *void* Functions; Other calling modes of function; Inline Functions.

掌握：Defining a Function; Calling a Function; Passing Parameters between Functions; Default Formal Parameters Functions; Overloading Functions.

【3】重点内容：

(1) Defining a Function.
(2) Calling a Function.
(3) Passing Parameters between Functions.
(4) Default Formal Parameters Functions.
(5) Overloading Functions.

【4】难点内容：

(1) Passing Parameters between Functions.
(2) Default Formal Parameters Functions.
(3) Overloading Functions.

【5】思政案例：

Functions are a commonly used processing module. When applying functions, we should have a divide and conquer mentality, have a sense of overall situation, grasp the whole situation, coordinate planning, team cooperation, and sense of responsibility.

第 5 章：Common Used System Functions（支撑课程目标 1-4、2-4）

【1】知识点：

(1) Math functions.

(2) Time functions.

(3) Random functions.

(4) Character functions.

(5) Formatting of output.

【2】学习目标：

理解：Math functions; Time functions.

掌握：Random functions; Character functions; Formatting of output.

【3】重点内容：

(1) Math functions.

(2) Time functions.

(3) Random functions.

(4) Character functions.

(5) Formatting of output.

【4】难点内容：

(1) Character functions.

(2) Formatting of output.

第 6 章：Single-Dimensional Arrays（支撑课程目标 1-5、1-6、2-5、2-6、3-1、3-8）

【1】知识点：

(1) Declaring, initializing, and accessing Array ant its elements.

(2) Processing Arrays: copying, searching, sorting, etc.

(3) Passing Arrays to Functions.

(4) Character Array and C-String.

(5) *string* Type.

【2】学习目标：

理解：Processing Arrays: copying, searching, sorting, etc.; Character Array and C-String.

掌握：Declaring, initializing, and accessing Array ant its elements; Passing Arrays to Functions; *string* Type.

【3】重点内容：

(1) Declaring, initializing, and accessing Array ant its elements.

(2) Processing Arrays: copying, searching, sorting, etc.

(3) Passing Arrays to Functions.

(4) Character Array and C-String.

(5) *string* Type.

【4】难点内容：

(1) Passing Arrays to Functions.

(2) Character Array and C-String.

(3) *string* Type.

【5】思政案例：

Arrays are essentially storing multiple data of the same type together, which means "birds of a feather flock together, people flock together". In practical life, we hope that everyone can regularly stay awake and make friends with excellent and positive people, so that they can also become excellent.

第 7 章：Multi-dimensional Arrays（支撑课程目标 1-5、2-5）

【1】知识点：

(1) Declaring, Initializing, and accessing 2-D Arrays and its elements.

(2) Processing Two-Dimensional Arrays.

(3) Passing Two-Dimensional Arrays to Functions.

(4) Multidimensional Arrays.

【2】学习目标：

理解：Declaring, Initializing, and accessing 2-D Arrays and its elements; Processing Two-Dimensional Arrays; Multidimensional Arrays.

掌握：Processing Two-Dimensional Arrays.

【3】重点内容：

(1) Declaring, Initializing, and accessing 2-D Arrays and its elements.

(2) Processing Two-Dimensional Arrays.

(3) Passing Two-Dimensional Arrays to Functions.

(4) Multidimensional Arrays.

【4】难点内容：

(1) Declaring, Initializing, and accessing 2-D Arrays and its elements.

(2) Processing Two-Dimensional Arrays.

(3) Passing Two-Dimensional Arrays to Functions.

(4) Multidimensional Arrays.

第 8 章：Class and object（支撑课程目标 1-7、2-7、2-8）

【1】知识点：

(1) Basic Features of OO.

(2) Defining Classes and Constructing Objects.

(3) Constructors.

(4) Separating Class Definition from Implementation.

(5) Preventing Multiple Inclusions.

(6) Inline Member Functions.

(7) Constant Members: Data and Functions.

(8) Static Members: Data and Functions.

(9) Destructors.

(10)Passing Objects to Functions.

(11)Returning an Object from Functions.

(12)Array of Objects.

(13)Object Composition.

(14)Friend functions and friend classes.

(15)*struct* and *union* Types.

【2】学习目标：

了解：Basic Features of OO.

理解：Separating Class Definition from Implementation; Preventing Multiple Inclusions; Passing Objects to Functions; Returning an Object from Functions; Array of Objects; Object Composition; Friend functions and friend classes; *struct* and *union* Types.

掌握：Defining Classes and Constructing Objects; Constructors; Inline Member Functions; Constant Members: Data and Functions; Static Members: Data and Functions; Destructors.

【3】重点内容：

(1) Defining Classes and Constructing Objects.

(2) Constructors.

(3) Separating Class Definition from Implementation.
(4) Preventing Multiple Inclusions.
(5) Inline Member Functions.
(6) Constant Members: Data and Functions.
(7) Static Members: Data and Functions.
(8) Destructors.
(9) Passing Objects to Functions.
(10)Returning an Object from Functions.
(11)Array of Objects.
(12)Object Composition.
(13)Friend functions and friend classes.
(14)*struct* and *union* Types.

【4】难点内容：

(1) Constructors.
(2) Constant Members: Data and Functions.
(3) Static Members: Data and Functions.
(4) Passing Objects to Functions.
(5) Returning an Object from Functions.
(6) Array of Objects.
(7) Object Composition.
(8) Friend functions and friend classes.

第 9 章：Program Structure（支撑课程目标 1-8、1-9、2-8）

【1】知识点：

(1) Scope and Visibility of Identifiers.
(2) Lifetime of Objects.
(3) Preprocessor Directives.
(4) Multi-file Programs.

【2】学习目标：

掌握：Scope and Visibility of Identifiers; Lifetime of Objects; Preprocessor Directives; Multi-file Programs.

【3】重点内容：

(1) Scope and Visibility of Identifiers.
(2) Lifetime of Objects.
(3) Preprocessor Directives.
(4) Multi-file Programs.

【4】难点内容：

(1) Scope and Visibility of Identifiers.
(2) Lifetime of Objects.

第 10 章：Pointer（支撑课程目标 1-10、1-11、2-9、3-1）

【1】知识点：

(1) Pointer Basics.
(2) Constant Pointers.
(3) *void* Pointers.
(4) Multiple Pointers.
(5) Pointers and Arrays.

(6) Dynamic Memory Allocation.

(7) Pointers and Functions.

(8) *this* Pointer.

(9) Copy Constructor.

(10)Shallow Copy and Deep Copy.

【2】学习目标：

理解：Pointer Basics; *void* Pointers.

掌握：Constant Pointers; Multiple Pointers; Pointers and Arrays; Dynamic Memory Allocation; Pointers and Functions; *this* Pointer; Copy Constructor; Shallow Copy and Deep Copy.

【3】重点内容：

(1) Pointer Basics.

(2) Constant Pointers.

(3) *void* Pointers.

(4) Multiple Pointers.

(5) Pointers and Arrays.

(6) Dynamic Memory Allocation.

(7) Pointers and Functions.

(8) *this* Pointer.

(9) Copy Constructor.

(10)Shallow Copy and Deep Copy.

【4】难点内容：

(1) Constant Pointers.

(2) Multiple Pointers.

(3) Pointers and Arrays.

(4) Dynamic Memory Allocation.

(5) *this* Pointer.

(6) Copy Constructor.

(7) Shallow Copy and Deep Copy.

【5】思政案例：

Pointers are a manifestation of the powerful features of C++, but they are particularly prone to errors. I hope everyone has the willpower to try, not afraid of failure, and to face difficulties, and to cultivate their attention to detail and patience in the learning process.

第 11 章：Templates（支撑课程目标 1-12、1-13）

【1】知识点：

(1) Function Template.

(2) Case Study: A Generic Sort.

(3) Class Template.

(4) Improving Class Template.

(5) The *vector* Class.

(6) Supplement: Returning a Reference from Function.

【2】学习目标：

理解：Case Study: A Generic Sort; Improving Class Template; The *vector* Class.

掌握：Function Template; Class Template; Supplement: Returning a Reference from Function.

【3】重点内容：

(1) Function Template.

(2) Class Template.

(3) Supplement: Returning a Reference from Function.

【4】难点内容：

(1) Function Template.

(2) Class Template.

(3) Supplement: Returning a Reference from Function.

第 12 章：Inheritance（支撑课程目标 1-13、3-10）

【1】知识点：

(1) Inheritance Basic.

(2) Inheritance Modes.

(3) Type Compatibility Rules.

(4) Constructors and Destructors.

(5) The Virtual Base Classes.

【2】学习目标：

理解：Inheritance Basic; The Virtual Base Classes.

掌握：Inheritance Modes; Type Compatibility Rules; Constructors and Destructors.

【3】重点内容：

(1) Inheritance Modes.

(2) Type Compatibility Rules.

(3) Constructors and Destructors.

【4】难点内容：

(1) Type Compatibility Rules.

(2) Constructors and Destructors.

【5】思政案例：

(1) The essence of inheritance is a development that mainly includes three aspects: absorption, addition, and modification.

(2) These three aspects are actually a manifestation of "upholding integrity and innovation", namely "developing while inheriting, innovating while upholding integrity", "maintaining direction while upholding integrity, sailing far in innovation", "only by upholding integrity can we not lose direction and avoid disruptive mistakes, and innovation can we grasp the times and lead the times".

第 13 章：Polymorphism（支撑课程目标 1-13、1-14）

【1】知识点：

(1) Operator Overloading.

(2) Virtual Functions and Dynamic Binding.

(3) Virtual Destructor.

(4) Pure Virtual Functions and Abstract Classes.

【2】学习目标：

理解：Pure Virtual Functions and Abstract Classes.

掌握：Operator Overloading; Virtual Functions and Dynamic Binding; Virtual Destructor.

【3】重点内容：

(1) Operator Overloading.

(2) Virtual Functions and Dynamic Binding.

(3) Virtual Destructor.

(4) Pure Virtual Functions and Abstract Classes.

【4】难点内容：

(1) Virtual Functions and Dynamic Binding.
(2) Virtual Destructor.
(3) Pure Virtual Functions and Abstract Classes.

（二）实验教学

The experiment is based on Dev-C++as Integrated Development Environment, with data representation and processing as the center, focusing on data types and representations, data processing and processing, and combining the two to process data in an object-oriented manner, thus introducing object-oriented programming. The experimental project adopts a teaching model that focuses on students' practical operations and is supplemented by teacher instruction. It fully utilizes various software resources, explores students' potential, lays a solid foundation in programming, and enhances their programming abilities. The project names and class hour allocation related information for experimental teaching are shown in Table 2.

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	Introduction to IDE	2	1	Veri.	Com.
2	Data Representation, Operator, Expression	2	1	Veri.	Com.
3	Statements	4	1	Comp.	Com.
4	Functions	2	1	Comp.	Com.
5	System Functions	2	1	Veri.	Com.
6	1-D Array	1	1	Desi.	Com.
7	String	1	1	Desi.	Com.
8	*m*-D Array	0	1	Desi.	Opt.
9	Class and Object	2	1	Desi.	Com.
10	Program Structure	2	1	Veri.	Com.
11	Pointer	4	1	Desi.	Com.
12	Polymorphism	2	1	Desi.	Com.

NOTE:
(1) Veri.: Verification Experiment
(2) Desi.: Designing Experiment
(3) Comp.: Comprehensive Experiment
(4) Com.: Compulsory Experiment
(5) Opt.: Optional Experiment

实验项目一：Introduction to IDE（支撑课程目标 2-1、2-2、3-1、3-2、3-3、3-4）

【1】实验内容：
(6) Basic usage of IDE.
(7) Simple Input and Output Statements.
(8) Simple Debug.

【2】实验目的：
(1) Familiar to IDE.
(2) Able to edit, compile and run simple programs.

(3) Able to debug simple programs.

【3】教学要求：

(1) The method of programming, i.e., editing, compiling and running the simple C++ program.

(2) Basic usage of input and output statements.

(3) The basic technique of debugging simple program.

【4】实验方式：Programming.

【5】实验场地与设备：Laboratory, Computer.

【6】思政案例：

(1) The C++language is developed from C language. For many years, it has been ranked among the top ten global programming languages, and learning it well is of great significance. But getting started is difficult, and you may feel bored at the beginning. I hope everyone has the courage to learn without fear of difficulties and persevere.

(2) The code (programming) style should be well-organized, because code is not only used to tell computers how to run, but also to show people, which is "people-oriented".

(3) Similar to human language, computer language also has syntax, semantics, nouns, polysemous words, etc. Unlike human language, humans have a certain ability to correct errors when understanding human language. However, computer languages such as C++have strict grammar rules, so we hope everyone has a sense of rules, ranging from complying with the rules and regulations of the class and school to complying with national laws and regulations.

实验项目二：Data Representation, Operators, Expressions（支撑课程目标 2-1、2-2、3-4）

【1】实验内容：

(1) Data type.

(2) Data Representation.

(3) Operator and Expression.

(4) Operator Precedence and Associativity.

(5) Data Type Conversions.

【2】实验目的：

(1) Able to express some information using C++ data.

(2) Able to solve some simple problems using C++ expression.

【3】教学要求：

(1) Solve some mathematical problems using C++ expressions.

(2) Some special situations about expression.

【4】实验方式：Programming.

【5】实验场地与设备：Laboratory, Computer.

【6】思政案例：

In all computer languages, operators have priority and associativity, and C++is no exception. This is the same as mathematics. In fact, doing things in life is also like this, especially when many things are pressed together, we should learn to analyze and plan, grasp the main contradiction, and first do important and urgent things.

实验项目三：Statements（支撑课程目标 2-3、3-4、3-5、3-6、3-7）

【1】实验内容：

(1) I/O (Input and Output) Statements.

(2) Control statements.

【2】实验目的：

(1) Able to program using I/O statement.

(2) Able to program using basic control statements to solve simple problems.

【3】教学要求：

(1) Introduce the usage of I/O Statements.

(2) Introduce the usage of C++ control statements.

【4】实验方式：Programming.

【5】实验场地与设备：Laboratory, Computer.

【6】思政案例：

(1) Sequence, selection, and loop structures are the three common structures in all computer programming languages and are the three major structures in structured programming design.

(2) Many aspects of life are steady, organized, and continually progressing, such as everyone's learning (sequential structure). However, life is also full of decisions, and we often encounter various choices. We should choose carefully, with correct values, and not make choices that go against our conscience and morality. We should also take responsibility for our decisions, such as choosing a major or taking the postgraduate entrance examination (selection structure). Even seemingly simple tasks require consistent effort, and small changes can accumulate to create significant differences (loop structure).

实验项目四：Functions（支撑课程目标 2-4、3-9）

【1】实验内容：

(1) Declaration and calling of functions.

(2) Debug using Step-Into.

【2】实验目的：

(1) Able to design and write functions.

(2) Able to call the function to solve some problems.

【3】教学要求：

(1) Declaration of functions.

(2) Calling ways of functions.

(3) Passing parameters.

【4】实验方式：Programming.

【5】实验场地与设备：Laboratory, Computer.

【6】思政案例：

Functions are a commonly used processing module. When calling functions, we should have a divide and conquer mentality, have a sense of overall situation, grasp the whole situation, coordinate planning, team cooperation, and sense of responsibility.

实验项目五：System Functions（支撑课程目标 2-4）

【1】实验内容：

(1) The organization of system functions.

(2) The calling of system functions.

【2】实验目的：

(1) Understand the system functions in C++.

(2) Able to the call system functions solve some problems.

【3】教学要求：

(1) Including header files.

(2) Calling system functions.

【4】实验方式：Programming.

【5】实验场地与设备：Laboratory, Computer.

实验项目六：1-D Array（支撑课程目标 2-5、3-1、3-8）

【1】实验内容：

(1) Define and initialize 1-D arrays.
(2) Accessing the elements of 1-D arrays.

【2】实验目的：

(1) Able to solve some problems using 1-D array.
(2) Able to process the data in 1-D array.

【3】教学要求：

(1) Declaration, initialization of 1-D arrays.
(2) Accessing the elements of 1-D arrays.

【4】实验方式：Programming.

【5】实验场地与设备：Laboratory, Computer.

【6】思政案例：

The essence of arrays is to store multiple data of the same type together, which can be likened to the proverb "birds of a feather flock together." In real life, it is important to surround yourself with positive and excellent people in order to become excellent yourself.

实验项目七：String（支撑课程目标 2-6）

【1】实验内容：

(1) The usage, input, and output of string (taken as character array and *string* object).
(2) The simple C-String functions.

【2】实验目的：

(1) Able to store strings into both character arrays and *string* objects.
(2) Able to commit simple string functions to process C-String and *string* objects.

【3】教学要求：

(1) Make the difference between *string* objects and *char* arrays (C-String).
(2) The storage structure of string
(3) The usage of string functions to process C-String and *string* objects.

【4】实验方式：Programming.

【5】实验场地与设备：Laboratory, Computer.

实验项目八：*m*-D Array（支撑课程目标 2-5）

【1】实验内容：

(1) The definition, initialization, and usage of2-D array.
(2) Methods of taking arrays as function’s parameters.

【2】实验目的：

(1) Able to store 2-D data into 2-D arrays.
(2) Able to take arrays as function’s parameters.

【3】教学要求：

(1) The definition, initialization, and usage of 2-D arrays.
(2) The method of passing array as parameter.

【4】实验方式：Programming.

【5】实验场地与设备：Laboratory, Computer.

实验项目九：Class and Object（支撑课程目标 2-7）

【1】实验内容：

(1) Declaration of class.
(2) Members of class.
(3) Instantiation of object.
(4) Calling of member function.
(5) Methods of taking objects as function’s parameters.

【2】实验目的：

(1) Able to declare classes.
(2) Able to define objects by classes.
(3) Able to call member functions by objects.
(4) Able to pass objects to a function.
(5) Able to return an object from functions.

【3】教学要求：

(1) Declaration of class.
(2) Write and combine member data and member function as a class.
(3) Instantiation of objects using classes.
(4) Calling of member functions via objects.
(5) Passing objects to a function and return an object from functions.

【4】实验方式：Programming.

【5】实验场地与设备：Laboratory, Computer.

实验项目十：Programming Structure（支撑课程目标 2-8）

【1】实验内容：

(1) Scope and visibility of identifier.
(2) Lifetime of object.
(3) Protection mechanism of sharing data, i.e., *const* object, *const* member, and *const* reference.

【2】实验目的：

(1) Able to analyze the scope, visibility, and lifetime of some identifiers.
(2) Able to edit the whole class using protection mechanism of C++.

【3】教学要求：

(1) The scope, visibility, and lifetime of identifiers.
(2) The protection mechanism of C++.

【4】实验方式：Programming.

【5】实验场地与设备：Laboratory, Computer.

实验项目十一：Pointer（支撑课程目标 2-9、3-1）

【1】实验内容：

(1) The definition and initialization of pointer.
(2) Array processing by pointer.
(3) The operation of pointers.
(4) The relationship between pointer and function.
(5) The relationship between pointer and object.

【2】实验目的：

(1) Able to process data (including array) by pointer indirectly.
(2) Able to pass pointers to functions as parameters.
(3) Able to processing object by pointers.

【3】教学要求：

(1) The definition and initialization of pointers.
(2) The way to process single data and 1-D array via pointers.
(3) The operation of pointers.
(4) The method that passing pointers as function’s parameters.
(5) The method that processing objects through pointer.

【4】实验方式：Programming.

【5】实验场地与设备：Laboratory, Computer.

【6】思政案例：

Pointers are a manifestation of the powerful features of C++, but they are particularly prone to errors. I hope everyone has the willpower to try, not afraid of failure, and to face difficulties, and to cultivate their attention to detail and patience in the learning process.

实验项目十二：Polymorphism（支撑课程目标 2-10）

【1】实验内容：

(1) Pure Virtual Functions and Abstract Classes.

(2) Polymorphism.

【2】实验目的：

(1) Understand the polymorphism mechanism.

(2) Able to use polymorphism thinking.

【3】教学要求：

(1) Pure Virtual Functions and Abstract Classes.

(2) Polymorphism.

【4】实验方式：Programming.

【5】实验场地与设备：Laboratory, Computer.

五、教学安排与学时分配

This subject mainly focuses on classroom teaching, supplemented by experimental teaching. Classroom teaching revolves around the basic characteristics, concepts, and practical operations of C++language. The subject revolves around the fundamental task of cultivating morality and nurturing talents, with the overall goal of cultivating socialist builders and successors with comprehensive development in morality, intelligence, physical fitness, aesthetics, and labor. The basic usage methods of C++are the main thread, and teaching activities are fully utilized in offline classrooms, online teaching resources, and ideological and political cases. The role of theoretical knowledge transmission, experimental practice operation, and value guidance is played, cultivating patriotism and the concept of building a strong country, consolidating basic knowledge, improving practical skills and innovative spirit, and establishing a correct outlook on life, world, and values. The allocation of offline teaching hours is shown in Table 3.

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	Introduction	4	2	0	0	0
第 2 章	Elementary of C++ Programming	8	2	0	0	0
第 3 章	Statements	4	4	0	0	0
第 4 章	Functions	4	2	0	0	0
第 5 章	System Functions	2	2	0	0	0
第 6 章	Single-Dimensional Arrays	4	2	0	0	0
第 7 章	Multi-dimensional Arrays	2		0	0	0
第 8 章	Class and Object	8	2	0	0	0
第 9 章	Program Structure	4	2	0	0	0
第 10 章	Pointer	4	4	0	0	0

第 11 章	Template	2	0	0	0	0
第 12 章	Inheritance and Derivation	4	0	0	0	0
第 13 章	Polymorphism	6	2	0	0	0
小计		56	24	0	0	0
总计	80					

六、教学方法

1. Teach basic principles, apply theory to practice, emphasize the basic characteristics of C++language, the basic ideas, and methods of C++programming, and cultivate students' innovative abilities.

2. The teaching method combines multimedia courseware with blackboard writing, offline self-learning and computer operation, and online Q&A with traditional Q&A to improve classroom vitality and Q&A ability.

3. Provide theoretical questions and experimental projects in advance, encourage students to learn on their own, stimulate their learning enthusiasm, cultivate their self-learning ability, problem-solving ability, and teamwork ability.

七、成绩评定

1. The exam form for this subject is closed examination and will be evaluated on a hundred marks system. Combine attendance, regular assignments, experiments, mid-term tests, and final exams to comprehensively evaluate grades.

(1) Attendance (5%): Students are required to participate in all stages of this subject and are not allowed to be absent.

(2) Homework (15%): Students are required to complete online assignments on time.

(3) Experiment (20%): Students are required to complete all experimental tasks.

(4) Mid-term testing (10%).

(5) Final exam (50%): Closed examination on Computer, 100 minutes.

2. The assessment qualification will be cancelled for those who lack more than one-third of the required experiments, assignments, or are absent for more than one-third of the planned class hours, or whose experimental assessment results are not qualified.

八、评价及持续改进

1. Syllabus

1、教学大纲

Check Period: 4 years, Revised Period: 4 years.

考核周期：4 年，修订周期：4 年。

Improvement Measures: the subject coordinator is responsible to organize all members in subject team for the syllabus discussion and modification, and the revised version shall be submitted to director of the major then the deputy dean (teaching affairs) for reviewing and approval, and finally be executed.

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. Evaluation

2、成绩评定考核

Check Period: 1 years. Evaluation method can be improved based on the comprehensive analysis of students' final scores.

考核周期：1 学年，评价依据：学生最终成绩综合分析。

Improvement Measures: based on the distribution of students' final scores, make adjustments and optimizations of subject content, and improve teaching methods.

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： 张建波，王子健

审核： 薛昌涛

批准： 史闻博

日期： 2023.07.05