中山大学软件学院

软件工程专业本科生课程教学大纲

Course Profile for Undergraduates Majoring in Software Engineering

最近更新 / Revision： 2010.04.07

<table>
<tr><td>课程编号
Course Code</td><td>SE-123</td><td>课程名称
Course Title</td><td colspan="2">程序设计（II）：C++语言与面向对象程序设计
Programming in High Level Languages (II):
Object-Oriented Programming in C++</td></tr>
<tr><td>课程类别
Course Type</td><td>专业必修课
Compulsory</td><td>学分
Credits</td><td colspan="2">2 + 1 = 3</td></tr>
<tr><td rowspan="3">周学时
Hours per Week</td><td rowspan="3">2 + 2 = 4</td><td rowspan="3">总学时
Total Hours</td><td>理论环节
Lecture Sessions</td><td>34</td></tr>
<tr><td>实验环节
Laboratory Sessions</td><td>34</td></tr>
<tr><td>学时合计
Total Hours</td><td>68</td></tr>
<tr><td>开课学期
Semester</td><td colspan="4">第一学年春季学期
Spring semester, 1st year</td></tr>
<tr><td>课程负责人
Course Coordinator</td><td colspan="4">李文军教授
Prof. Wen-jun LI
Email : lnslwj@mail.sysu.edu.cn
Homepage: http://ss.sysu.edu.cn/~lwj/</td></tr>
<tr><td>主要授课教师
Principal Instructors</td><td colspan="4">李文军教授、周晓聪副教授、张锋博士、孙雪冬博士、潘炎博士、徐亚波博士、蒋爱军博士
Prof. Wen-jun LI, A/Prof. Xiao-cong ZHOU, Dr. Feng ZHANG, Dr. Xue-dong SUN, Dr. Yan PAN, Dr. Ya-bo XU, Dr. Ai-jun JIANG</td></tr>
<tr><td>教学助理配置
Teaching Assistants</td><td colspan="4">每班配置 1 名 TA，负责批改编程作业并指导实验过程。
1 TA per class. The TA assists the instructor in grading programming projects, and tutoring at lab-time.</td></tr>
<tr><td>课外交流时间
Office Hours</td><td colspan="2">主讲教师：每周 2 小时。
Instructor: 2 hours per week.</td><td colspan="2">教学助理：无。
TA: N/A.</td></tr>
<tr><td>课程主页
Homepage</td><td colspan="4">http://my.ss.sysu.edu.cn/courses/cpp/
User Name: sser Password: ToBeNo.1</td></tr>
<tr><td>讨论社区
Forum</td><td colspan="4">PENDING</td></tr>
<tr><td>课程描述
Course Description</td><td colspan="4">Programming in High Level Languages (II) will focus on Object-Oriented Programming (OOP) and C++ language. The students will learn object-oriented and generic programming paradigms after the acquisition of Structured Programming (SP) paradigm through the study of its prerequisite, Programming in High Level Language (I), which focuses on Programming in C.

At the completion of the course, the students will obtain the knowledge of the principles, techniques, methodologies and styles for programming in C++, be equipped with the thinking in Object-Oriented Programming and Generic Programming. The students will also develop their practical skills in using at least one Integrated Development Environment (IDE) of C++ programming. Their solid programming knowledge gained in this course will be able to support them to take on the subsequent courses and future programming tasks in real-world applications and academic research.

The following topics will be covered in the course: history, programming styles and</td></tr>
</table>

	characteristics of C++ language; classes and objects; compile-time polymorphism, including function overloading, operator overloading and templates; run-time polymorphism, including inheritance, virtual functions, abstract base classes and run-time type identification; inheritance and multi-inheritance; template and STL; I/O streams; exception handling; basics of design patterns. 本课程以 C++语言为载体，详细介绍面向对象程序设计方法、技术与工具，是“程序设计 (I)”的重要后续课程，也是软件工程专业本科生的一门重要专业主干课程。 本课程的教学目标包括：指导学生在掌握了结构化程序设计方法和 C 语言的基础上，进一步学习面向对象程序设计方法和 C++语言；逐步掌握面向对象程序设计和泛型程序设计的核心思想，初步掌握使用 C++语言实现两种设计范例的技术和方法；熟悉至少一个可视化集成开发环境（IDE）的使用，为从事面向对象软件开发打下基础；并结合软件学院本科生培养的特点，采取双语教学模式，进一步培养学生外文专业书籍阅读和思维习惯，提高通过英语获取专业知识、解决问题的能力。 本课程的主要教学内容包括：C++语言的历史、风格与特点；类与对象的思想；编译时多态性：函数重载、操作符重载、类属（模板）机制；运行时多态性：继承、虚函数、抽象基类、运行时类型识别；继承与多继承；模板、STL；C++语言的 I/O 流处理；异常处理机制；设计模式基础；面向对象程序设计方法学基础、泛型程序设计方法学基础等。
先修课程 Prerequisites	1. SE-122 Programming in High Level Languages (I): Programming in C 程序设计（I）：C 语言程序设计
后续课程 Successive Courses	1. SE-129 Software Engineering Training: Elementary 软件工程实训（初级） 2. SE-221 Data Structures and Algorithms 数据结构与算法
理论教学部分 Lecture Sessions	
教材 Textbook	H. Deitel and P. Deitel. **C How to Program, 4th Ed.** Prentice Hall, 2004, ISBN 978-0-131-42644-3 影印版：《C 大学教程》（第 4 版），北京：清华大学出版社，2007，ISBN 978-7-302-15582-9
教学参考书 References	[1] S. Lippman, J. Lajoie and B. Moo. **C++ Primer, 4th Ed**. Addison-Wesley Professional, 2005, ISBN 0-201-72148-1 影印版：《C++ Primer 英文影印版》（第 4 版），图灵原版计算机科学系列，北京：人民邮电出版社，2006，ISBN 7-115-15169-5 中译版：李师贤、蒋爱军、梅晓勇、林瑛译，《C++ Primer 中文版》（第 4 版），图灵程序设计丛书，北京：人民邮电出版社，2010，ISBN 7-115-22017-2 [2] B. Stroustrup. **The C++ Programming Language: Special Edition, 3rd Ed**. Addison-Wesley Professional, 2000, ISBN 0-201-70073-5 影印版：《C++程序设计语言（特别版）》，国外优秀信息科学与技术系列教学用书，北京：高等教育出版社，2001，ISBN 7-040-10095-9 中译版：裘宗燕译，《C++程序设计语言（特别版）》，计算机科学丛书，北京：机械工业出版社，2002，ISBN 7-111-10202-9

[3] B. Stroustrup. **The Design and Evolution of C++**. Addison-Wesley Professional, 1994, ISBN 0-201-54330-3
影印版:《C++语言的设计和演化》,经典原版书库,北京:机械工业出版社,2002,ISBN 7-111-09592-8
中译版:裘宗燕译,《C++语言的设计和演化》,计算机科学丛书,北京:机械工业出版社,2002,ISBN 7-111-09098-5

[4] B. Eckel. **Thinking in C++, Volume 1: Introduction to Standard C++, 2nd Ed**. Prentice Hall, 2000, ISBN 0-139-79809-9
中译版:刘宗田等译,《C++编程思想,第 1 卷:标准 C++导引》(第 2 版),计算机科学丛书,北京:机械工业出版社,2002,ISBN 7-111-10807-8
B. Eckel and C. Allison. **Thinking in C++, Volume 2: Practical Programming**. Prentice Hall, 2003, ISBN 0-130-35313-2
影印版:《C++编程思想第 2 卷:实用编程技术》,经典原版书库,北京:机械工业出版社,2004,ISBN 7-111-12188-0
中译版:刁成嘉译,《C++编程思想,第 2 卷:实用编程技术》,计算机科学丛书,北京:机械工业出版社,2005,ISBN 7-111-17115-2

[5] D. Vandevoorde and N. Josuttis. **C++ Templates: The Complete Guide**. Addison-Wesley Professional, 2002, ISBN 0-201-73484-2
影印版:《C++ Templates》,原版风暴系列,北京:中国电力出版社,2004,ISBN 7-508-31924-9
中译版:陈伟柱译,《C++ Templates 中文版》,C 和 C++经典著作,北京:人民邮电出版社,2008,ISBN 7-115-17181-8

[6] N. Josuttis. **The C++ Standard Library: A Tutorial and Reference**. Addison-Wesley Professional, 1999, ISBN 0-201-37926-0
影印版:《C++标准库教程》,大学计算机教育国外著名教材系列,北京:清华大学出版社,2006,ISBN 7-302-12826-X
中译版:侯捷、孟岩译,《C++标准程序库——自修教程与参考手册》,侯捷译作系列,武汉:华中科技大学出版社,2002,ISBN 7-560-92782-3

[7] E. Gamma, R. Helm, R. Johnson and J. Vlissides. **Design Patterns: Elements of Reusable Object-Oriented Software**. Addison-Wesley Professional, 1994, ISBN 0-201-63361-2
影印版:《设计模式:可复用面向对象软件的基础》,经典原版书库,北京:机械工业出版社,2002,ISBN 7-111-09507-3
中译版:李英军等译,《设计模式:可复用面向对象软件的基础》,计算机科学丛书,北京:机械工业出版社,2004,ISBN 7-111-07575-7

[8] T. Misfeldt, G. Bumgardner and A. Ray. **The Elements of C++ Style**. Cambridge University Press, 2004, ISBN 0-521-89308-9
双语版:罗小平译,《C++编程风格:英汉对照》,图灵程序设计丛书,北京:人民邮电出版社,2008,ISBN 7-115-17605-9

[9] 李师贤、李文军、周晓聪等,《面向对象程序设计基础(第二版)》(面向 21 世纪课程教材),北京:高等教育出版社,2005,ISBN 7-040-16650-X

教学方法 Approach	1. Bilingual teaching: English and Chinese. 2. Interactive and heuristic discussions in the classroom. 3. Emphasis on labs/practices as well as lectures.
理论教学内容 Lectures	1. **Introduction to C++ (4 hr)** history of C++; C++ programming styles and characteristics; from C to C++; Visual C++ 6.0 IDE; C++ features. 2. **Classes and Objects (6 hr)** structures and classes; classes definitions; classes access control and class scope; classes, objects and abstract data types; object creation, object initialization and object destruction; default constructor, converted constructor, copy constructor; encapsulation. 3. **Compile-Time Polymorphism: Function and Operator Overloading (6 hr)** function overloading; operator overloading basics; unary and binary operator overloading; class members and friends: two operator overloading approaches; overloading examples of major operators; overloading and compile-time polymorphism. 4. **Run-Time Polymorphism: Inheritance and Dynamic Binding (6 hr)** inheritance basics; inheritance and multi-inheritance; virtual base classes and virtual inheritance; virtual functions, pure virtual functions, abstract base class; static binding vs. dynamic binding; run-time polymorphism; run-time type identification; design patterns basics. 5. **Templates, STL and Generic Programming (6 hr)** templates: function templates and class templates; STL programming basics: containers, algorithms, iterators, adapters and allocators; generic programming and template, STL; compile-time polymorphism and templates. 6. **Exception Handling (2 hr)** concepts of exception handling; user-defined exceptions; standard exception handling; exception handling vs. error handling. 7. **I/O Streams (2 hr)** I/O stream class library; I/O stream basics: standard, file and string; formatted I/O streams. 8. **Multi-paradigm Programming in C++ (2 hr)** evolution of programming methodologies: procedural programming, object-oriented programming and component-based programming; history of object-oriented programming languages: Simula, Smalltalk, C++, Eiffel, Java and C#; characteristics of C++ language: encapsulation, inheritance and polymorphism; software reuse in C++: inheritance, polymorphism and templates; cutting edge of software reuse: design patterns; generic programming in C++. **Totally 34 hours.**
布置作业 Assignments	No written assignments.

实验教学部分 **Laboratory Sessions**	
实验课时间 Lab Hours	**34** hours
实验课地点 Lab Venue	专用实验室。 In the laboratory.
实验课方式 Lab Approach	The laboratory session is divided into 2 parts: on-line programming tests and off-line programming projects. On-line programming tests will be conducted aperiodically in lab time. Off-line programming projects may be done in lab-time or in out-of-class time, but must be submitted before the deadlines. All the programming projects must be submitted electronically (by FTP) and all the submitted work must be done by that student. The programming projects will be implemented in ANSI/ISO C++. The programs will be compiled and tested using Visual C++ 6.0 (or Dev C++) running on Microsoft Windows 2000 and above. The major duties of instructors and TAs in the lab are to help the students to design and write programs properly, and gradually cultivate their abilities of programming independently. In on-line programming test time, instructors and TAs will monitor and track the students' performance. Discussions with classmates or friends are encouraged, but only in the stage of reaching the understanding of each assigned problem. Every program must be written as your own work, although you may ask instructors or TAs for help in debugging. Do not, under any circumstances, copy any segment of another's program. Writing code for use by another or using another's code in any form violates the University's academic regulations, and will be punished harshly.
课外实验辅导 Additional Tutoring	N/A.
实验教学内容 Practices / Labs	● **Compulsory Off-Line Programming Projects** 1. C++ Programming Basics and IDEs 2. Programming with Classes and Objects 3. Programming with Operator Overloading 4. Programming with Loop Structures 5. Inheritance, Virtual Base Classes, Virtual Functions and Dynamic Binding 6. Templates, STL and Generic Programming 7. Exception Handling, C++ I/O Streams ● **Optional Projects** ● **Compulsory On-Line Programming Tests** Items chosen from ACM/ICPC training websites.
实验平台 Platforms	● Operating System: Microsoft Windows

	• IDE: Microsoft Visual C++ 6.0 (or later, or Dev C++).	
实战技能培养 Skill Development	• Skills to create, debug and test a program written in the C programming language. • Familiar with an Integrated Development Environment (IDE) for C programs, such as Microsoft Visual C++.	
总评成绩计算 Course Assessment		
评分标准 Grading	Weekly Programming Assignments	30%
	Final Online Test	30%
	Final Examination	40%
	Total	100%
备注 Note	The final online test will be organized by the School of Software, following the model of ACM/ICPC. Students who do not pass the final online test will definitely fail in the course. 期末上机考试将由软件学院统一组织，采取 ACM/ICPC 模式。未通过期末上机考试则本课程成绩将不及格。	