



普通高等教育“十一五”国家级规划教材

国家精品课程教材

清华大学计算机系列教材

C++语言程序设计 (第4版) 学生用书

郑莉 董渊 编著



清华大学出版社



普通高等教育“十一五”国家级规划教材

国家精品课程教材

清华大学计算机系列教材

C++语言程序设计

(第4版) 学生用书

郑莉 编著
清华大学图书馆
藏书章

清华大学出版社
北京

内 容 简 介

本书是与《C++ 语言程序设计(第4版)》(清华大学出版社, ISBN 9787302227984)配套的学生用书。本书首先给出了一个简要的“学习指南”,其余章节与主教材《C++ 语言程序设计(第4版)》相对应,每章内容分为3部分:第一部分是要点导读,主要是为自学读者指明学习重点,建议学习方法。第二部分是实验,每章都有一个精心设计的实验,与《C++ 语言程序设计(第4版)》相应章内容配合,使读者在实践中达到对主教材内容的深入理解和熟练掌握。每一个实验都包括实验目的、实验任务和实验步骤。第三部分是习题解答,给出了《C++ 语言程序设计(第4版)》各章习题的参考答案。本书附录中还给出4个非常实用的附录——Visual C++ 开发环境简介、ASCII码表、STL中通用算法函数原型及说明、STL中类属算法函数原型及说明。本书适合作为高等院校程序设计课程的教辅书。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

C++ 语言程序设计(第4版)学生用书/郑莉等编著. —北京:清华大学出版社,2011.6
(清华大学计算机系列教材)

ISBN 978-7-302-25352-5

I. ①C… II. ①郑… III. ①C语言—程序设计—高等学校—教材 IV. ①TP312

中国版本图书馆CIP数据核字(2011)第070546号

责任编辑:谢琛 薛阳

责任校对:焦丽丽

责任印制:何芊

出版发行:清华大学出版社

地 址:北京清华大学学研大厦A座

<http://www.tup.com.cn>

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62795954,jsjic@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015,zhiliang@tup.tsinghua.edu.cn

印 装 者:北京鑫海金澳胶印有限公司

经 销:全国新华书店

开 本:185×260

印 张:14.5

字 数:337千字

版 次:2011年6月第1版

印 次:2011年6月第1次印刷

印 数:1~8000

定 价:23.00元

产品编号:039452-01

出版说明

根据国务院关于高等学校教材工作分工的规定,我部承担了全国高等学校、中等专业学校工科电子类专业教材的编审、出版的组织工作。由于各有关院校及参与编审工作的广大教师共同努力,有关出版社的紧密配合,从1978年至1985年,已编审、出版了两轮教材,正在陆续供给高等学校和中等专业学校教学使用。

为了使工科电子类专业教材能更好地适应“三个方向”的需要,贯彻“努力提高教材质量,逐步实现教材多样化,增加不同品种、不同层次、不同学术观点、不同风格、不同改革试验的教材”的精神,我部所属的七个高等学校教材编审委员会和两个中等专业学校教材编审委员会,在总结前两轮教材工作的基础上,结合教育形势的发展和教学改革的需要,制订了1986—1990年的“七五”(第三轮)教材编审出版规划。列入规划的教材、实验教材、教学参考书等近400种选题。这批教材的评选推荐和编写工作由各编委会直接组织进行。

这批教材的书稿,是从通过教学实践、师生反映较好的讲义中经院校推荐,由编审委员会(小组)评选择优产生出来的。广大编审者、各编审委员会和有关出版社为保证教材的出版和提高教材的质量,做出了不懈的努力。

限于水平和经验,这批教材的编审、出版工作还会有缺点和不足之处,希望使用教材的单位,广大教师和同学积极提出批评建议,共同为不断提高工科电子类专业教材的质量而努力。

电子工业部教材办公室

《C++ 语言程序设计(第4版)》学习指南

《C++ 语言程序设计(第4版)》是针对初学程序设计语言的读者编写的入门教材,预期的读者主要有这样几类:初学程序设计的自学读者、以此为C++ 课程教材或参考书的在校学生、以此为参考资料的C++ 程序员、C++ 课程教师。针对教师,另外编写了配套的教师用书。对有经验的程序员而言,自然也不必遵循什么学习指导。因此主要是对前两类读者提出一些学习建议,这里首先给出学习本套教材的总体建议,在本书的后续各章中还会有详细的导读。

1. 主教材《C++ 语言程序设计(第4版)》的学习方法

自学读者,在阅读教材时,应该边阅读、边实践。有条件的,应该坐在计算机前,边阅读边亲自编写每一个例题程序,如果对于某些概念、语法存有疑问,应该立即编写程序予以验证。在完全理解了主教材内容以后,再开始做实验和习题。

对于在校学习C++ 课程的学生,应按照教师讲课的进度,提前预习教材。所谓预习,并不是要完全看懂,如果都看懂了,就不必听课了。预习的目的是大致浏览一下新的内容,了解哪些是难点、重点,将疑问记下来,听课时就比较主动。上课之后要及时复习,然后再写作业。复习时要边看书边看笔记,这时一定要认真阅读书上的内容,要完全领会真懂。教师可能不会在课堂上讲解书中的每一个例题,对于教师课上讲的例题和书上的例题,课后复习时都要阅读、上机实践,达到完全理解,要能够自己独立编写例题程序,还要尝试用不同的方法解决问题。做到这些以后,再开始写作业。

对于以上建议有的读者会不以为然,但这是大多数初学者达到事半功倍的途径。清华大学的C++ 课就一直是这样要求学生的,也曾经有学生对上述要求不同意:认为要求太麻烦了,又要预习又要复习,我们有那么多课程要学,没有时间。但是后来的无数事实证明,想省时间的多半欲速则不达,甚至出现“夹生饭煮不熟”的情况,而一步一个脚印往前走的,走得最从容,总体上花的时间也最短。当然,这只是针对大多数读者的一般建议,每个人还要根据自己的情况选择合适的方法。

2. 学生用书的使用方法

每学习一章主教材内容,都应该及时通过实验和习题巩固知识、提高实践能力。学生用书中的实验,是针对主教材每一章的重点内容设计的最基本的实践任务,有详细的实验指导,很容易入手,应该首先完成。完成实验之后,可以根据自己的时间和教师的要求,选择部分或全部习题来做。

本书给出了全部习题的答案,这是为了方便没有教师指导的自学读者。但是不少读者在没有深入思考之前就急于看答案,这是有害无益的,这样做不仅不能真正提高自己的编程能力,还会扼杀自己的创造性思维能力。有些学生在临近考试的时候,就来诉苦:书上的例题和习题解答都能看懂,可是自己写程序就不会下手。仔细一问,这些都是平时急于看习题

答案的。自学读者纷纷来信欢迎习题解答,而大多数教师都不希望学生看到习题解答。这个矛盾困扰了我们很久,始终没有找到两全的解决方案。所以本书中给出习题解答,但同时建议学生尽量独立完成习题。

当然,有些章的习题较多,如果你没有时间全部做完,也可以将一部分习题解答作为例题来学习。

3. 关于编程能力的困惑

学完本套教材之后,许多读者都会遇到这样的困惑: C++ 语言学会了,但是面对实际问题还是不知道该如何写程序,这是为什么? 每个学期末都有很多学生来问这种问题,于是每学期最后一节课,都要举这样的例子: 我们都是以中文为母语的,对中文的掌握可谓精通了。但是你是否有能力用中文写出某个项目的实施方案、某个企业的发展规划? 恐怕大多数学生做不到,因为只掌握语言是不够的,还需要有相关的专业知识和工作经验。

编写程序的道理也是相同的,就是要用程序语言将需要解决的问题和解决问题的方案描述清楚。仅仅掌握C++ 语言是不够的,还需要学习解决各类问题的专门方法。为此很多程序语言教材(包括本套教材)都声称: 不仅介绍语言本身而且介绍分析问题和解决问题的方法。但是事实上,这些都只是介绍分析问题和解决问题方法的皮毛。如果只读一本薄薄的书,就什么程序都会写了,那学校里还要开设那么多基础课和专业课干什么? 比如“高等数学”、“计算方法”、“数据结构”、“软件工程”等等。学会一门高级语言只是掌握了一种描述工具,要真正具备较强的分析问题和解决问题的能力,要学的东西还很多,除了认真学习、勤奋实践以外没有捷径可走。所以初学者要给自己定一个现实的目标,本套书主教材的第9~10章介绍一些基本的对群体数据的管理方法和类库中相应的算法,就是为了使读者掌握一些基本的解决问题的方法,能够运用C++ 语言编写程序解决一些简单问题,并为读者今后继续学习相关课程打开一扇窗户。

前 言

计算机程序设计是一门实践性很强的课程,因此仅仅通过阅读教科书或听课是不可能完全掌握的,学习程序设计的最重要环节就是实践。对于自学读者来说,更多一重困难,就是在学习和实践过程中缺乏指导。

凡是学习程序设计的人,往往有这样的感觉:看书或听课时,对老师讲的和书上写的内容基本上能够理解,但是当需要自己编程时却又无从下手。相信每一个讲授程序设计课程的教师都有过这样的经历:有些问题,尽管我们在课上再三强调,反复举例,学生还是不能完全理解,上机时更是错误百出。应该说,这是学习过程中的必然现象。

要想能够把书本上的知识变为自己所具有的能力,所需要的是实践、实践、再实践。在实践环节中,起主导作用的是学习者自己,旁人是无法代劳的,也不能期望有什么一蹴而就的捷径。但是由于学生在实践过程中不能随时随地得到指导,因此花费时间较多,总感觉程序设计课作业负担太重,有的学生甚至因为花四五个小时调不通一个简单的程序而失去学习兴趣。像C++这样面向对象的程序设计语言学习起来尤其如此。

本书是在原《C++ 语言程序设计(第3版)学生用书》的基础上修订编写的。本书作为与《C++ 语言程序设计(第4版)》配套的学生用书,目的就在于为读者的学习提供一些指导,为提高读者的编程能力助一臂之力。使读者在实践的过程中少些曲折和彷徨,多些成功的乐趣。

本书首先给出一个简要的“学习指南”,其余章节与主教材《C++ 语言程序设计(第4版)》相对应,每章内容分为三部分:第一部分是“要点导读”,主要是为自学读者指明学习重点,建议学习方法。第二部分是“实验指导”,每章都有一个精心设计的实验,与《C++ 语言程序设计(第4版)》相应章内容配合,使读者在实践中达到对主教材内容的深入理解和熟练掌握。每一个实验都包括“实验目的”、“实验任务”、“实验步骤”和“答案与提示”,实验的完整参考程序可从网上下载(<http://rjjsjc.cic.tsinghua.edu.cn>)。第三部分是习题解答,给出了《C++ 语言程序设计(第4版)》各章习题的参考答案,共214道习题。每个题目可能有多种解法,这里我们仅给出一种参考解法。大部分题目是编程题,我们在解答中给出了主要程序段的源程序清单,有时不是完整的程序,如果需要运行这些语句,只需将它们插入调试程序即可。

这些习题解答和实验内容不仅可以指导读者上机练习,也可以由教师选做例题在课上演示,使教学内容更加丰富。如果读者没有足够的时间一一做完全部习题和实验,可以将剩下的题解作为例题阅读也不失为一种好的选择。

本书中的全部程序都在 Windows 环境下 Visual C++ 中测试通过。

为了给自学读者更多的帮助,作者将自己在清华大学的教学资源(包括例题源代码、电子教案等)向中国国内读者免费开放。网址是 <http://rjjsjc.cic.tsinghua.edu.cn>。

参与本书编写工作的还有：何江舟、张瑞丰、孟鸿利、田荣牌、李玉山、薛海伟。

感谢读者选择使用本书，欢迎您对本书内容提出意见和建议，我们将不胜感激。作者的电子邮件地址：zhengli@mail.tsinghua.edu.cn，来信标题请包含“C++ book”。

作 者

2010年12月于清华大学

目 录

第 1 章 绪论	1
要点导读	1
实验 1 C++ 开发环境应用入门(2 学时)	1
习题解答	10
第 2 章 C++ 简单程序设计	13
要点导读	13
实验 2 C++ 简单程序设计(4 学时)	14
习题解答	20
第 3 章 函数	40
要点导读	40
实验 3 函数的应用(2 学时)	40
习题解答	42
第 4 章 类与对象	52
要点导读	52
实验 4 类与对象(4 学时)	52
习题解答	55
第 5 章 数据的共享与保护	67
要点导读	67
实验 5 数据的共享与保护(2 学时)	67
习题解答	69
第 6 章 数组、指针与字符串	77
要点导读	77
实验 6 数组、指针与字符串(4 学时)	78
习题解答	79
第 7 章 继承与派生	97
要点导读	97
实验 7 继承与派生(4 学时)	97
习题解答	99

第 8 章 多态性	113
要点导读	113
实验 8 多态性(2 学时)	113
习题解答	114
第 9 章 群体类和群体数据的组织	130
要点导读	130
实验 9 群体类和群体数据(4 学时)	131
习题解答	132
第 10 章 泛型程序设计与 C++ 标准模板库	154
要点导读	154
实验 10 标准模板库的应用(2 学时)	154
习题解答	155
第 11 章 流类库与输入/输出	175
要点导读	175
实验 11 流类库与输入/输出(2 学时)	175
习题解答	177
第 12 章 异常处理	187
要点导读	187
实验 12 异常处理(2 学时)	187
习题解答	188
附录 A Visual C++ 开发环境简介	194
附录 B GNU make 及 Makefile 文件的编写	201
附录 C ASCII 码表	203
附录 D STL 中类属算法函数原型及说明	205

第 1 章 绪 论

要点导读

本章作为全书的开篇,旨在使读者初步了解面向对象的程序设计语言的由来,初步了解面向对象的程序设计思想的基本特点,概要性地了解面向对象的软件开发方法,为后续章节的学习奠定基础。

为什么需要首先有一个初步和概要性的了解呢?一方面,这是为了在以后的学习中具体接触到每一个新的概念、语法时都能够清楚地认识到,它在面向对象的方法中、在C++语言中的地位和作用是什么。另一方面,是希望读者在一开始就能够认识到,面向对象的思想与人类所习惯的思维方式是一致的,虽然C++语言比起面向过程的语言(如C语言)来要复杂许多,但是C++设计者的目的是为了把事情变得更简单,而不是故弄玄虚将事情搞得更复杂。事实上,正是由于C++语法的复杂性,使得它的表现能力更强,程序员用C++来写程序的时候能够更容易、更灵活地实现各种功能。

读者在阅读本章第1.1~1.3节时会感觉很多问题理解不透,这是正常的。因为需要学完本教材的全部内容,才能对C++语言和面向对象的方法有一个全面的认识。而本章在一开始就给出了一个全面介绍,虽然尽量使用通俗的语言,但是肯定仍有一些问题是读者现在不能完全理解的。对此读者不必深究,对第1.1~1.3节的内容阅读后有个大致的了解就行。

第1.4节介绍了信息的表示与存储,这是程序设计的基本知识,是必须掌握的基础。建议读者认真学习、完全掌握。不过有些读者可能会觉得这些知识在编程中并没有直接使用,不学这一节好像也不影响学习编程。但是没有这些基础知识,会影响读者对程序的理解。当然,如果觉得枯燥,也可以先略过这一节,待以后遇到疑问时,再来学习。因此有的教师在讲课时也略过这一节,留给学生自学,我本人就是这样做的。

第1.5节简单介绍了程序的开发过程和一些术语,不必死记硬背,最好结合实验来体会。

本章的主要实验任务是学会使用两种C++开发工具,本书的实验用的是Visual Studio 2008开发环境(Windows)和Eclipse开发环境(UNIX/Linux)。认真完成这一实验很重要,了解开发环境的基本功能,是完成以后各章实验的基础。

实验 1 C++ 开发环境应用入门(2 学时)

本实验分为两个部分,读者可以根据自己使用的开发环境,选择其中的一个部分。

第一部分: Visual Studio 2008 C++ 开发环境应用入门

一、实验目的

- (1) 了解 Visual Studio 2008 的特点。
- (2) 熟悉 Visual Studio 2008 的开发环境。

(3) 学习用 Visual Studio 2008 编写标准的 C++ 控制台程序。

二、实验任务

使用 Visual Studio 2008 建立一个非图形化的标准 C++ 程序,编译、运行例 2.1,即以下程序:

```
#include <iostream >
using namespace std ;
int main()
{
    cout<<"Hello!\n";
    cout<<"Welcome to c++!\n";
    return 0;
}
```

三、实验步骤

(1) 启动 Visual Studio 2008 开发环境。

从“开始”菜单中选择“程序”|Microsoft Visual Studio 2008|Microsoft Visual Studio 2008,显示 Visual Studio 2008 开发环境主窗口。

(2) 创建一个项目。

① 单击 File 菜单中的 New 选项中的 Project,显示 New Project(新建项目)对话框(如图 1-1 所示)。

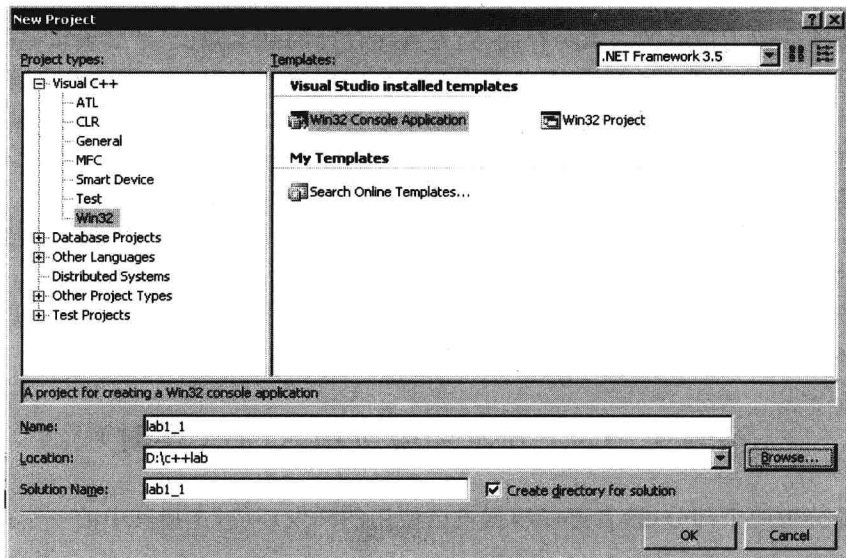


图 1-1 New Project 对话框

② 展开 Visual C++ 树状节点单击 Win32,选择 Win32 Console Application(Win32 控制台应用程序)。在 Location(位置)文本框中指定一个路径 D: \C++ lab,在 Name(名称)文本框中为项目输入一个名字“lab1_1”,单击 OK(确定)按钮。

③ 在弹出的 Win32 Application Wizard-lab1_1 对话框中,单击左侧的 Application Settings,并选择 Console application 单选项和 Empty Project 复选框,然后单击 Finish(完成)按钮(如图 1-2 所示)。

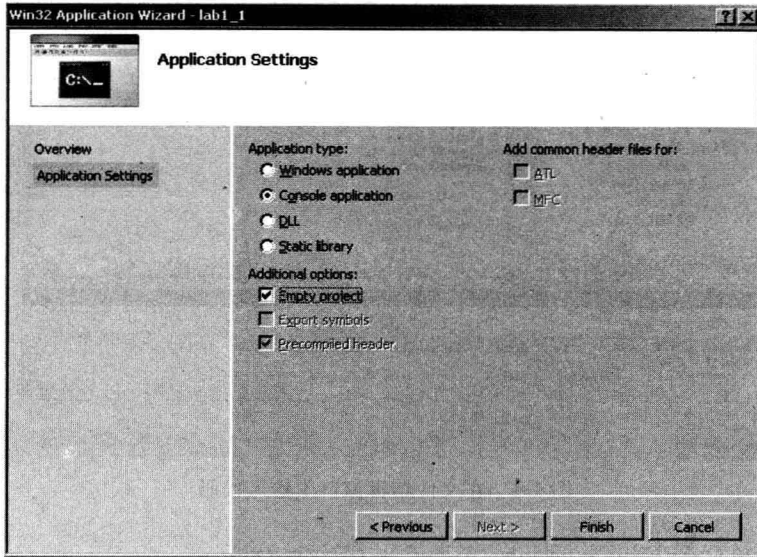


图 1-2 Win32 Application Wizard-lab1_1

④ 最后在 New Project Information 对话框中单击 OK 按钮,完成项目的建立。

(3) 建立C++ 源程序文件。

① 选择菜单命令 Project|Add New Item,弹出 Add New Item 对话框。

② 如图 1-3 所示,单击左侧树状结构中的 Code 节点,然后在右侧选择C++ File (.cpp),并填入文件名称“lab1_1”,单击 OK 按钮,完成新建C++ 源程序文件。

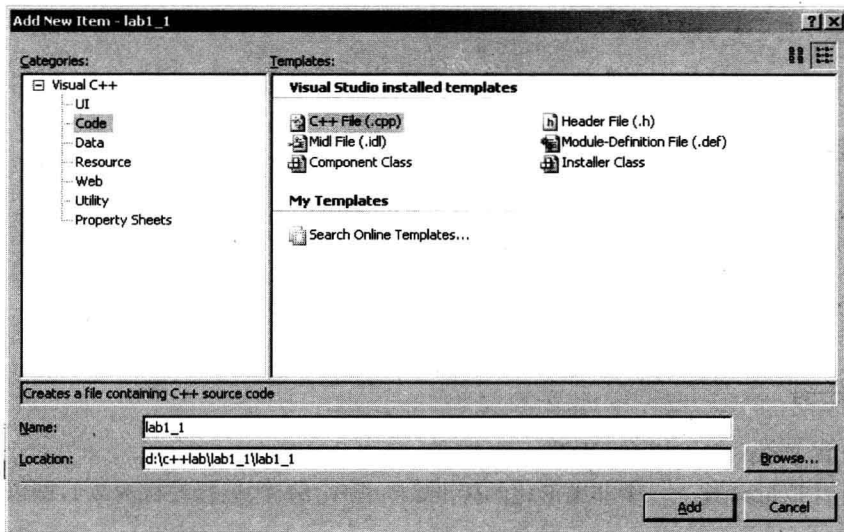


图 1-3 新建C++ 源程序文件

(4) 编辑C++ 源程序文件内容。

① 在文件编辑窗口中输入代码(如图 1-4 所示)。

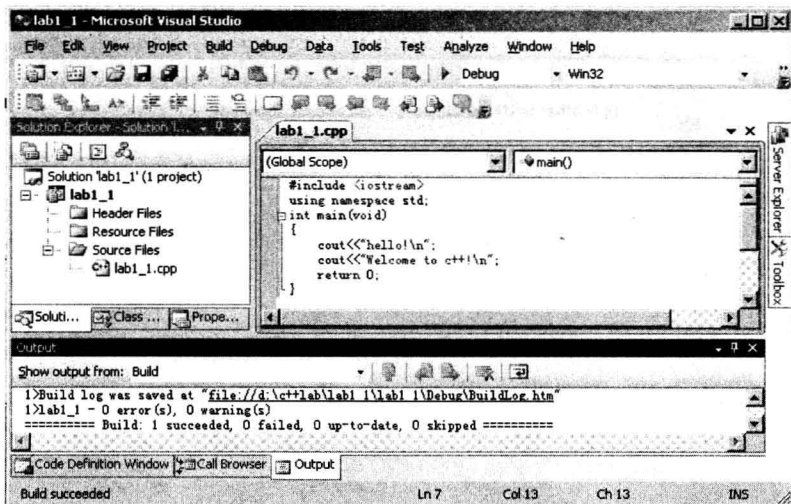


图 1-4 在文件编辑窗口中输入代码

② 选择菜单命令 File|Save lab1_1. cpp 保存这个文件。

(5) 建立并运行可执行程序。

① 选择菜单命令 Build|Build Solution 建立可执行程序。

如果正确输入了源程序,此时便成功地生成了可执行程序 lab1_1. exe。如果程序有语法错误,则屏幕下方的状态窗口中会显示错误信息。根据这些错误信息对源程序进行修改后,重新选择菜单命令 Build|Build Solution,建立可执行程序。

② 选择菜单命令 Debug|Start Without Debugging 运行程序,观察屏幕的显示内容。

(6) 关闭工作空间。

选择菜单命令 File|Close 关闭工作空间。

第二部分：Eclipse 开发环境应用入门

一、实验目的

- (1) 熟悉 UNIX/Linux 环境下开发 C/C++ 的方法。
- (2) 了解 Eclipse 的特点。
- (3) 熟悉 Eclipse 开发环境。
- (4) 学习用 Eclipse 编写标准的C++ 控制台程序。

二、实验任务

- (1) 安装 Eclipse SDK 和 CDT 插件。
- (2) 使用 Eclipse 建立一个非图形化的标准C++ 程序,编译、运行教材例 2.1,即以下程序:

```
#include <iostream.h>
int main(int argc, char * argv[])
```

```
{
    cout<<"Hello!\n";
    cout<<"Welcome to c++!\n";
    return 0;
}
```

三、实验步骤

(1) 安装 Eclipse IDE for C/C++ Developers。

Eclipse 通常被人们用作开发 Java 程序的 IDE。实际上,更准确地说,Eclipse 是一个平台,它的核心(Core)是很小的,其他绝大部分功能都是通过插件的形式组织上去的,包括 Eclipse 自带的 JDT(Java Development Tools)开发环境,此外 Eclipse 是一个跨平台的开发环境,可以运行于 Windows 环境,也可以运行于 UNIX/Linux 环境。

正是由于 Eclipse 的这种插件组织以及跨平台的特性,使得 Eclipse 的应用非常广泛,可以被用来作为多种计算机语言的开发环境,如 C/C++、Fortran 等。

本次实验以 Linux 环境下 Eclipse 的使用为例,开发环境的版本为 Eclipse IDE for C/C++ Developers 1. 2. 2,其中包含了 Eclipse Platform 3. 5. 2; Eclipse C/C++ Development Tools(CDT)6. 0. 2,由于 Eclipse 各个版本之间的变动比较大,因此请尽量使用与本书相同的版本,如图 1-5 和图 1-6 所示。

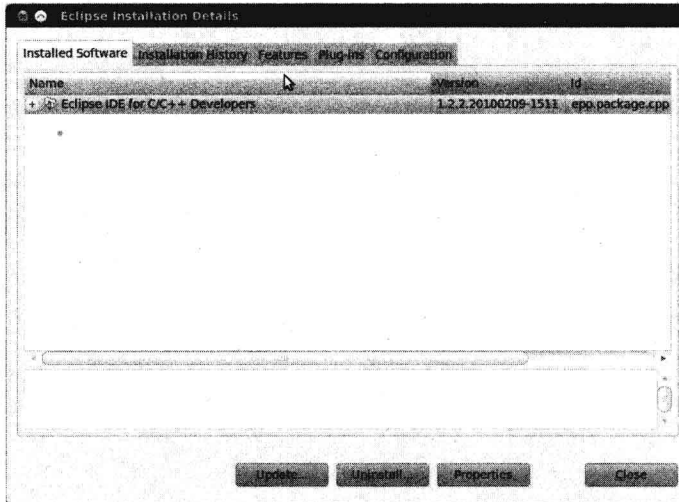


图 1-5 Eclipse IDE for C/C++ Developers 的版本号

下面介绍 Eclipse IDE for C/C++ Developers 1. 2. 2 的安装方法,如果读者已经安装过 Eclipse SDK 但尚未安装 CDT,可参考 Eclipse 官方网站的相关说明安装 CDT 插件,以下假设读者尚未安装任何 Eclipse 相关的软件包。

Eclipse IDE for C/C++ Developers 可以直接从 Sun 的官方网站上下载(<http://www.eclipse.org/downloads>)。Eclipse 属于绿色软件,它不需要安装,下载成功后直接把文件 eclipse-cpp-galileo-SR2-linux-gtk.tar.gz 解压缩到一个合适的目录就可以了,如/root/installed。然后运行/root/installed/eclipse/目录下的 eclipse 二进制文件就可以了。

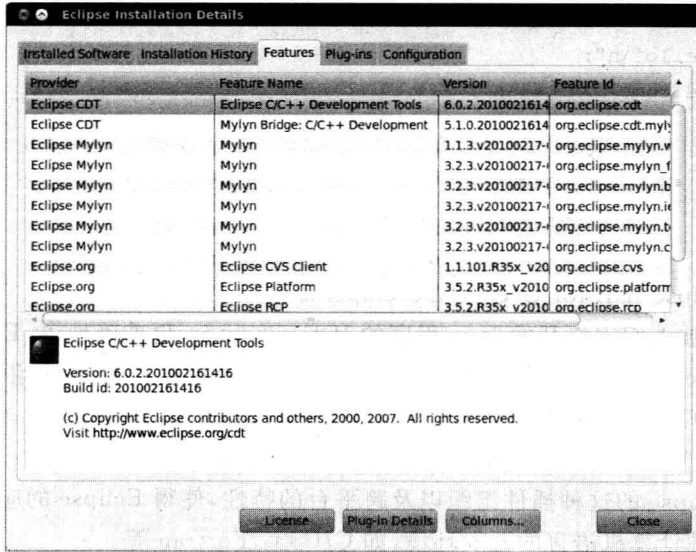


图 1-6 CDT 以及 Eclipse Platform 的版本号

这时 Eclipse 会提示用户选择 Workspace, 随便输入一个, 如 /root/Workspace。接着会看到如图 1-7 所示的欢迎界面。



图 1-7 Eclipse 的欢迎界面

以 Ubuntu 为例, 打开终端运行“sudo apt-get install sun-java6-jre sun-java6-jdk sun-java6-plugin”, 根据提示便可以安装, 如果运行这条命令时出现了类似“现在没有可用的软件包”的问题, 那么可能是 Ubuntu 版本较新, 新版本的 Ubuntu 默认去掉了 partner 软件源, 在 System(系统)中的 Administration(系统管理)中单击 Software Source(软件源), 选择 Other Software(其他软件)选项卡, 勾选 <http://archive.canonical.com/lucid> partner, 关闭窗口时系统会提示更新软件源, 更新完毕后, 再次在终端运行上述命令, 根据提示安装 Sun

JRE,安装 JRE 成功后在终端运行“sudo update-alternatives--config java”,根据提示选择 Sun 提供的 JVM,将 JVM 切换到 Sun JVM。

(2) 创建一个C++ 项目。

① 单击 File 菜单中的 New 子菜单,选择 Project,打开如图 1-8 所示的对话框,选择 C++ Project。

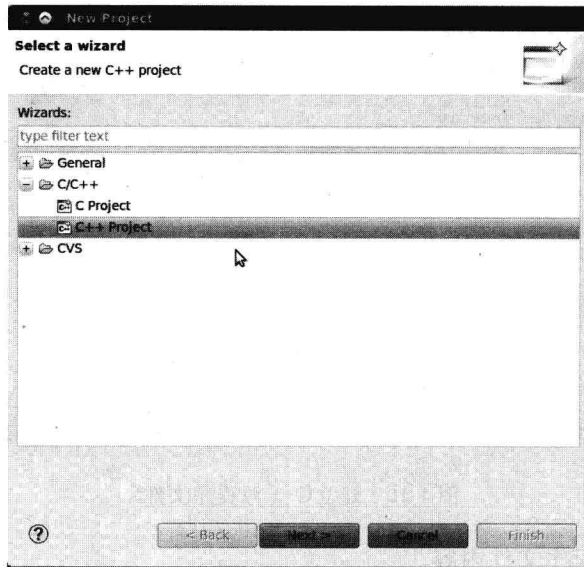


图 1-8 新建 Standard Make C++ Project

② 接着输入一个项目名称,如“Hello”,如图 1-9 所示。然后直接单击 Finish 按钮即可。这时 Eclipse 会问你是否打开 C/C++ 视图(Perspective),选择“是”,并让它“Remember my decision”。

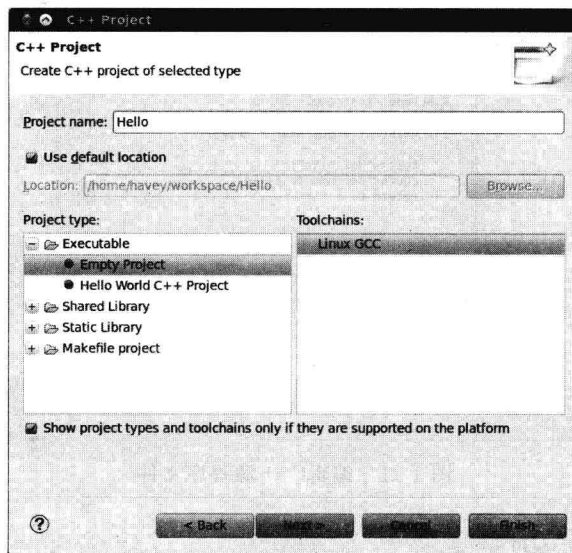


图 1-9 新建一个 Hello 项目