

C++

面向对象程序设计

主 编 任 哲



高等教育出版社
HIGHER EDUCATION PRESS

C++ 面向对象程序设计

任 哲 主编

高等教育出版社

图书在版编目(CIP)数据

C++面向对象程序设计/任哲主编.—北京:高等教育出版社,2003.5

ISBN 7-04-012166-2

I.C... II.任... III.C语言-程序设计-高等学校-教材 IV.TP312

中国版本图书馆CIP数据核字(2003)第030453号

责任编辑 司马镭 **封面设计** 吴昊 **责任印制** 蔡敏燕

书 名 C++面向对象程序设计
主 编 任 哲

出版发行 高等教育出版社
社 址 北京市西城区德外大街4号
邮政编码 100011
总 机 010-82028899
传 真 021-56965341

购书热线 010-64054588
021-56964871
免费咨询 800-810-0598
网 址 <http://www.hep.edu.cn>
<http://www.hep.com.cn>
<http://www.hepsh.com>

排版校对 南京展望照排印刷有限公司
印 刷 商务印书馆上海印刷股份有限公司

开 本 787×1092 1/16
印 张 16.5
字 数 392 000

版 次 2003年6月第1版
印 次 2004年6月第2次
定 价 29.80元(附光盘)

凡购买高等教育出版社图书,如有缺页、倒页、脱页等质量问题,请在所购图书销售部门联系调换。

版权所有 侵权必究

前 言

面向对象程序设计是一种当今普遍应用的程序设计方法,C++就是一门优秀的面向对象的程序设计语言。因此,C++面向对象程序设计已成为高等学校计算机及其相关专业开设的一门专业基础课程。

一、C++是目前最好的面向对象程序设计语言之一

所谓编程序,就是提供一些计算机指令,让计算机来处理某些事情。因此就需要把处理事情的步骤跟计算机“说清楚”,这样就必须有人与计算机进行沟通的语言。

由于计算机是用电子线路构成的,它只懂得用二进制数表达的信息。因此,在计算机应用的早期,计算机程序设计人员只好用0和1组成的信息,把要完成的工作告诉它,这就是最早的计算机语言,也就是通常所说的机器语言。这时的计算机的角色就是一个按机器语言程序去工作的工人。

显然机器语言不便于人们记忆和使用。于是人们就用一些容易记忆的东西来帮助记忆,这样,使用助记符的汇编语言就发展起来了。而把汇编语言翻译成机器语言的工作是由计算机去做的。因此,计算机就有了两个功能:一是把助记符翻译成机器语言的功能;二是按机器语言执行程序的功能。或者说计算机是身兼二职:既是翻译又是工人。

自从有了汇编程序这类翻译程序之后,人们就尽可能把一些底层的工作交给计算机去做,这样形成了高级语言,例如BASIC、FORTRAN、Pascal、COBOL、C等。并且各种语言都在不断地更新版本,其中一个重要的目标就是使计算机语言尽可能地接近人类语言。而C语言以其强大的功能和自由的书写风格,成为目前计算机程序设计的主流语言之一。

自从面向对象的程序设计理论出现之后,C++又适时地在C的基础上构建了面向对象的体系,并充分利用了抽象、封装、继承、多态等特性,使C++更进一步地接近了人类的语言,人们能更加容易地掌握和使用它,并且使用它开发出来的程序更加稳定了。所以说,C++是目前最好的面向对象程序设计语言之一。

二、C++是学习大部分程序开发工具的基础

现在流行的开发工具,诸如Visual C++、Borland C++ Builder、Delphi、Visual Basic等,都是以面向对象为基础的,而Visual C++和Borland C++ Builder其实就是用C++开发Windows应用程序的工具,并且,Windows本身就是用C来编写的操作系统。因此没有C/C++的基础,学习程序开发工具、尤其是想比较深入地使用现代的可视化开发工具是极其困难的。所以有人说:不会C++很难成为优秀的程序设计人员。

三、本书的编写思路及特点

本书是为高等学校工科学生编写的教材,为此在本教材的编写过程中,力图体现以

下特点:

1. 以面向对象为核心讲述程序设计的方法

科学的发展过程不应该是学习的顺序,当然也不应该是组织教学内容的顺序,尤其面对的是计算机程序设计的初学者时更是如此。因此,本书摒弃了先讲解面向过程的设计,后讲解面向对象的程序设计的教学顺序。而是根据初学者对程序设计语言的认识规律来安排教学内容的顺序。因此,本书一开始就介绍了面向对象设计的一些基本概念,使学生初步了解对象、类及C++程序的基本框架,然后在后面的教学中,不断地对这个框架进行充实、细化,从而使学生更全面、更清晰地理解和掌握这些基本概念,并逐步会用这些概念去分析问题和解决问题。

2. 从人类自然语言表述事物的方法入手介绍C++

高级程序设计语言的发展目标之一,就是使其尽可能地接近人类的自然语言。目前,继承、多态机制的引入,已使高级程序设计语言很接近人类的自然语言了,这样就从人类自然语言出发,介绍和学习计算机语言创造了良好的条件。因此,本书从人类自然语言表述事物的方法入手,使用伪代码使学生了解和掌握计算机语言的语法规则和语言的使用规律,从而使看起来很晦涩难懂的计算机语言生动活泼起来。

3. 介绍了函数的使用方法

计算机程序主要有两大要素:数据和操作,在C++中操作是用函数来实现的(广义地看,运算符也是函数),因而对于函数的理解和使用在本书中做了较多的阐述。

4. 重点讲述实用技术

重点讲述了C++的一些新的、实用的技术,而对C++仍然支持的那些由C继承来、但已经逐渐淘汰的内容,譬如:宏、结构、联合、多维数组、指针的一些应用方法等等,在不影响学生掌握C++语言的基础上尽量不讲,或者少讲。

5. 适当介绍新技术

适当地介绍C++的最新技术。C++仍在不断地发展,C++的标准类模板库(STL)就是其重要的成果之一。它可以大大地提高应用程序的开发效率,并且已经被纳入ANSI C++标准,所以有对它做一个介绍的必要。

6. 配套齐全,便于教与学

为方便读者练习及上机实验,本书除每章后附有习题外,还配有实验和习题解答一书,提供了书中所有例题、习题解答的源代码。

参加本书编写的有:任哲教授、姜丰副教授和宋绍成副教授。全书由张长海教授主审。

在本书的编写过程中,作者参考了大量的资料,也使用了其中的一些内容和例题,在此谨对这些书籍的作者们表示诚挚的谢意。

由于作者的水平有限,书中的错误或不妥之处,恳请读者批评指正(作者的电子邮箱是:renzhe71@sina.com)。

作者
2003年3月

目 录

第 1 章 计算机程序设计基础知识	1
第一节 计算机程序的基本概念	1
第二节 计算机应用程序的基本要素和程序设计方法	3
第三节 C++ 语言概述	7
习题	11
第 2 章 实际事物的描述方法——类与对象	12
第一节 类与对象的基本概念	12
第二节 类成员访问权限的控制	17
第三节 C++ 程序的基本结构	18
第四节 常用的系统预定义的类和对象	20
习题	22
第 3 章 数据类型、运算符和表达式	24
第一节 字符集与标识符	24
第二节 基本数据类型	25
第三节 常量与变量及常量的声明	27
第四节 运算符与表达式	33
习题	39
第 4 章 函数	42
第一节 函数	42
第二节 函数的调用	47
第三节 递归函数	53
第四节 函数的重载	55
习题	59
第 5 章 程序流程控制语句	62
第一节 顺序结构	62
第二节 选择结构	63
第三节 循环结构	76
习题	83

第 6 章 类及自定义数据类型	85
第一节 抽象和封装	85
第二节 类的成员函数	89
第三节 类的聚集	97
第四节 类成员的访问控制	99
第五节 类模板	102
第六节 自定义数据类型	104
习题	110
第 7 章 继承	113
第一节 继承和派生的基本概念	113
第二节 继承方式对类成员访问控制的影响	114
第三节 派生类的构造函数	119
第四节 多继承派生类成员的标识问题	121
习题	125
第 8 章 数组	126
第一节 数组的定义及数组元素的引用	126
第二节 数组的存储和初始化	133
第三节 数组作为函数参数	137
第四节 对象数组	139
习题	141
第 9 章 指针	143
第一节 指针的基本概念	143
第二节 指针与数组	148
第三节 对象的指针	155
第四节 动态内存分配	157
第五节 指针与函数	161
第六节 用 const 修饰指针	164
第七节 字符串	165
第八节 关于指针的小结	167
习题	168
第 10 章 多态	169
第一节 用虚函数实现多态性	169
第二节 用运算符重载实现多态性	178
习题	186

第 11 章 C++ 的程序结构	190
第一节 文件结构	190
第二节 标识符的作用域与可见性	193
第三节 生存期	194
第四节 外部存储类型	196
第五节 类的共享数据	196
第六节 C++ 编译预处理命令	201
习题	204
第 12 章 I/O 流	206
第一节 I/O 流的概念	206
第二节 I/O 流类标准对象	207
第三节 文件流类	208
第四节 I/O 文件流类的成员函数	212
第五节 输出格式控制	218
第六节 字符串流类	220
第七节 自定义流类的操作	221
习题	223
第 13 章 异常处理	224
第一节 异常的概念	224
第二节 异常处理的基本思想	224
第三节 C++ 的异常处理机制	225
第四节 异常处理中的构造和析构	228
第五节 异常对象的使用	230
习题	231
附录 1 标准模板类库	232
第一节 群体数据和群体类	232
第二节 STL 及其容器	234
第三节 迭代器与通用算法	237
第四节 常用的通用算法应用举例	243
附录 2 运行时类型识别	250
参考文献	253

第 1 章

计算机程序设计基础知识

本章首先介绍了计算机语言和计算机程序的基本概念,然后简要地介绍面向过程和面向对象的程序设计方法,最后用一个最简单的程序例子,说明 C++ 程序的一般构成及程序的开发过程。

第一节 计算机程序的基本概念

一、计算机语言与计算机程序

众所周知,人类语言(英语、法语、汉语、西班牙语……)是人们在工作和生活中进行信息传递和信息交流的工具。这些人类使用的、自然发展起来的语言叫做人类自然语言。而人与计算机进行信息交流则需要使用计算机语言,计算机语言与人类的自然语言一样,是一套具有语法、词法规则的系统。

在日常的劳动和生活中,当人们向其他人表述某项工作的做法时,通常将工作分成若干个步骤,用语言(当然要使用对方能理解的语言)表达出来。例如表 1-1 就是某电话机的故障排除说明书。

表 1-1 故障现象与排除方法对照表

故障现象	故障排除方法
无声	把插头与接线盒连接好
只能接听不能拨出	检查拨号制式是否正确
杂音	更换连接线或接线盒

上面是用表格的方法说明了排除故障的步骤。如果单纯用文字书写出来则是:
如果(无声)把插头与接线盒连接好;
如果(只能接听不能拨出)检查拨号制式是否正确;
如果(杂音)更换连接线或接线盒。

那么这个指导人们去排除电话机故障的说明书就是一个程序,一个让人执行某种工

作的程序。

人们使用计算机的目的是让计算机代替人做某项工作,为此人们就必须把做这项工作的步骤用计算机所能接受的语言(计算机程序设计语言,例如 C、BASIC、Pascal 等等)描述出来,这种用计算机程序设计语言书写的工作步骤的集合就叫做**计算机程序**。当计算机启动这个程序后,就会按照程序所说明的工作步骤,一步一步地完成人们交给它的任务。所以说:

- 计算机程序设计语言是人们向计算机传递信息的工具。
- 计算机程序是以某种计算机程序设计语言为工具编制出来的动作序列,它描述了程序员要求计算机执行的操作。

二、计算机高级语言和低级语言

1. 高级语言和低级语言

由于计算机在内部是使用二进制数来工作的,或者说计算机的硬件只能识别二进制数码,因此,在计算机中,不管什么信息都用二进制代码来表示。例如,用 0010 1000 表示加法运算,用 1001 1000 表示减法运算。这种由二进制数来表达计算机操作信息的语言叫做**机器语言**。显然这种语言不便于人们的记忆和使用,为解决这个问题,人们使用一些便于记忆的符号(助记符)来表达这些机器指令,如用 ADD 代表加法运算,用 SUB 代表减法运算等。这种助记符形式的语言叫**汇编语言**。尽管使用汇编语言要比使用机器语言要方便得多,但汇编语言的语法及表达方式仍与人类自然语言差距较大,不便于普及和使用。随着计算机技术的发展,一些接近人类自然语言的程序设计语言便发展了起来,如 FORTRAN、BASIC、Pascal 和 C 等语言,这些比较接近人类自然语言的计算机程序设计语言就叫做**高级语言**,而那些计算机比较容易接受的机器语言和汇编语言叫做**低级语言**。

在长期的应用实践中,高级语言也得到了不断地发展、更新和完善,目前只剩下一些比较优秀的语言在广泛地被应用。C++就是这些优秀语言中比较杰出的一种。

2. 汇编程序、编译程序、解释程序

计算机汇编语言和高级语言又叫做**源语言**。用源语言编写的程序叫做**源程序**。计算机是不能直接执行源程序的,必须通过某种翻译程序将其翻译成机器语言程序以形成二进制的“目标代码”才能被计算机所执行。这个“目标代码”也叫做“目标程序”。翻译程序有三种不同类型:汇编程序、编译程序、解释程序。

- **汇编程序**。其任务是把用汇编语言写成的程序翻译成机器语言形式的目标代码。
- **编译程序**。其任务是把高级语言写成的源程序翻译成目标代码。
- **解释程序**。这也是一种翻译程序,同样也是将高级语言源程序翻译成机器指令。它与编译程序的区别在于:它是边翻译边执行的,即翻译一句、执行一句,直至将整个程序翻译执行完毕。解释程序不产生整个的目标程序。这种程序执行效率较低。

如果计算机配备了上述翻译程序,则该计算机就扮演了两个角色:一是翻译角色;二是执行程序,完成用户任务的角色。

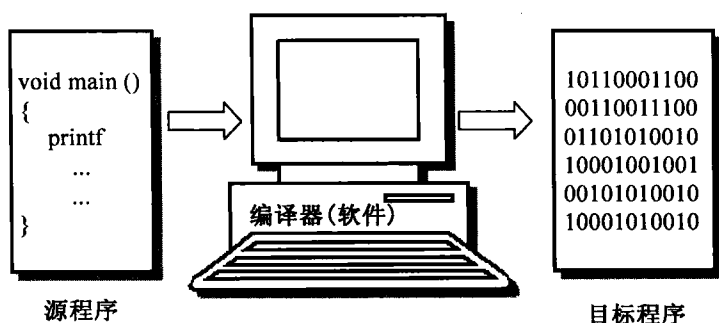


图 1-1 计算机的编译器把源程序转成目标程序

三、编写计算机程序的基本要求

编写一个计算机应用程序,首先要保证程序能正确地完成任务,与此同时,还要求所编写的程序应具有以下特点:

- **可读性。**程序设计结构清晰、可理解性好,程序书写规范,尽量避免复杂的个人程序设计技巧,使其他人能很容易读懂。
- **可维护性。**当计算机程序的应用需求发生变化时,不需要太多的开销就可以扩展和增强程序的功能。
- **可移植性。**程序应该能在各种计算机和操作系统上良好地运行,并且运行结果相同。

第二节 计算机应用程序的基本要素和程序设计方法

人们生活和工作在数据的海洋中,时刻都在同各式各样的数据打交道。许多生活和工作的过程就是一系列对数据进行操作的过程。例如,使用机床进行切削可以改变金属棒的直径,通过加热可以改变水的温度,在太阳下曝晒可以改变粮食的含水量等等。这里的“直径”、“温度”、“含水量”就是数据,“切削”、“加热”、“曝晒”就是对数据的操作。而计算机也是依靠对数据的操作(运算)来解决各种实际问题的,所以**计算机程序的基本要素为数据和操作。**

数据与操作在计算机程序中的关系及其组织方式是一个重要的问题,当然也是计算机工作者一直在研究的重要课题。数据与操作的组织方式不同,程序设计方法也就大不一样。本节,将对数据和操作做进一步的讲解,同时简单介绍两种不同的程序设计方法。

一、计算机程序的基本要素

1. 数据

大千世界,事物多种多样。花是红的,叶是绿的;足球是圆的,橄榄球是长的……,不

同的事物有不同的特征,人们就是利用事物的特征来区别事物和描述事物的。在计算机中,人们是用数据来描述事物属性(特征)的,不同的事物具有不同的属性,于是也就具有不同的数据。例如,张三的身高是 1.73 m,李四的身高是 1.60 m,在计算机中的表达为:张三. 身高=1.73 m;李四. 身高=1.60 m(这描述了张三和李四的区别:张三是高个,李四是矮个)。那么这里所说的身高,就是这两个人的属性之一。当然,描述这两个人的区别还可以有一些其他数据,如:眼睛颜色、头发密度、性格、学习成绩、专业等等。对此,可以用表 1-2 来描述他们的属性。

表 1-2 张三和李四的区别

	身 高	眼睛颜色	头发密度	性 格	学习成绩	专 业
张 三	1.73	黑	稀	倔 强	中	计算机
李 四	1.60	褐	浓	活 泼	良	自动化

再例如,张三的汽车和李四的汽车的区别也可以用数据来表达(表 1-3)。

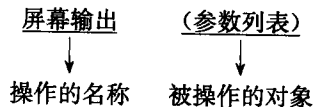
表 1-3 张三和李四汽车的区别

	颜 色	座 位 数	最高时速	ABS
张三的汽车	红	2	200	yes
李四的汽车	白	4	180	no

在上面的例子里,身高、性格、颜色等就是数据,而 1.73 m、1.60 m,倔强、活泼,红、白就是对应的数据值。数据是区别事物的重要依据,因此数据是计算机程序的最基本要素之一。

2. 操作

与解决问题的过程一样,计算机解决问题的过程,就是不断对数据进行运算和处理的过程,这些运算和处理叫做对数据的操作(简称操作)。例如,在上面的例子中,如果希望数据可以在计算机显示器的屏幕上输出,就应该向计算机说明一个操作:



其中,“屏幕输出”是这个操作的名称,表明要干什么。数据名称与被操作的对象都是用文字表达的,为了将两者区别开来,在操作名称后面要加括号“()”。如果这个操作需要初始数据的话,就将这个初始数据添在括号中,表明对哪个(些)数据进行操作。例如:

屏幕输出(颜色)

这项操作的含义就是要把事物的颜色这个属性在屏幕上显示出来。

在计算机程序设计中,操作一般叫做函数。因为它的表达方式与数学中函数的表达方式 $f(x)$ 相近。而实际上,它们的功能也相近。在以后的讲述中,操作和函数不加

区别。

总之,操作也是计算机程序基本要素之一,是对数据进行运算处理的一个完整的程序段。不同的语言中,对操作的称呼有所不同。在C++中,操作叫做“函数”。

二、面向过程程序设计

人们把程序看成是对数据进行的一系列操作过程。当任务比较复杂时,将整个操作过程分解为一些较小的相对独立的操作过程,因此在程序中就是一些数据和操作,所以有人将程序看作是数据+操作(图1-2)。用这种理念进行程序设计的方法叫做面向过程程序设计。

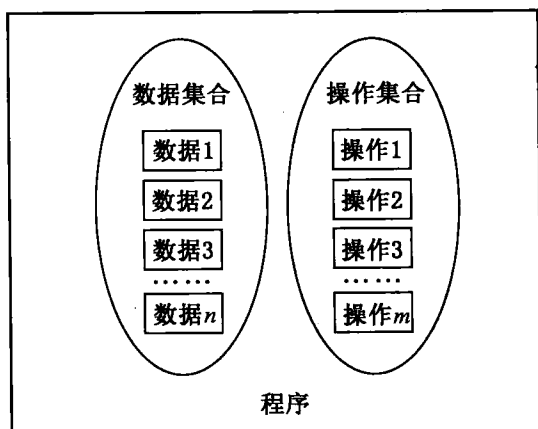


图1-2 面向过程程序设计中数据与操作的关系

随着计算机应用程序的任务日益复杂、规模日益庞大,这种面向过程程序设计方法的一些缺点也就日益突出。这样,就产生了面向对象的程序设计思想。

三、面向对象程序设计

从上面的介绍中可以看出,在面向过程程序设计中,数据与操作是分离的,是各不相同的。程序中的任意一个数据可以使用所有的操作,而任意一个操作都可以对任一数据进行加工。这样,在大规模程序的设计中,极易造成混乱,使数据的安全性,程序的可读性、可维护性都大大地恶化。例如,设计一个程序,在这个程序中,有两个人:张三和李四,他们每人都有一辆汽车,通过重新喷漆可以改变他们汽车的颜色,为了作到这一点,定义一个操作:喷漆(参数)。那么在这个程序中就包含了如图1-3所示的内容。

显然,喷漆这个操作,除了对汽车的颜色这个数据有意义之外,对其他数据没有任何实际意义。但是,在程序中并没有明确操作和数据之间的约束关系,那么在程序设计中就可能出现:

喷漆(李四. 性格)

这种致命的逻辑错误。

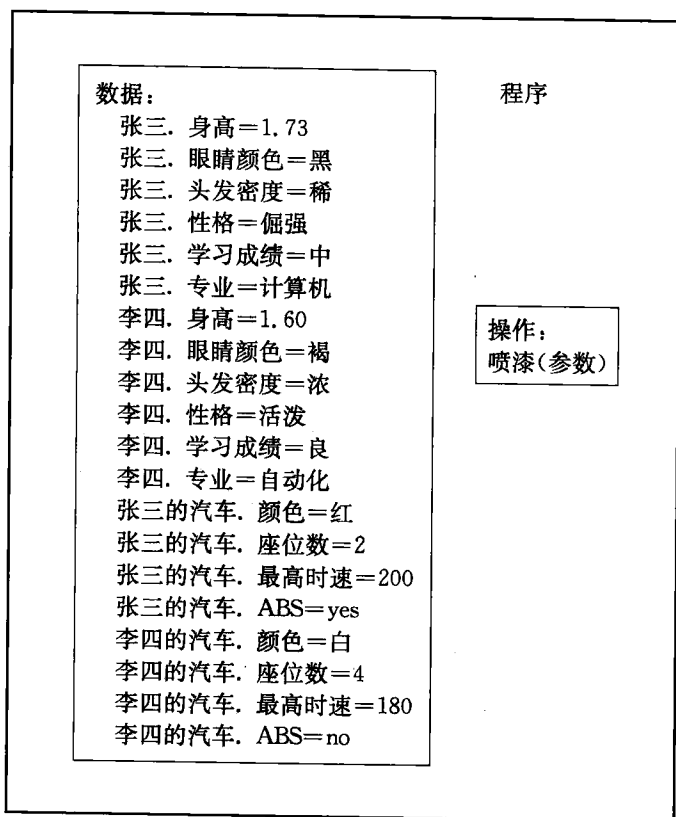


图 1-3 重新喷漆程序所含的内容

解决这个问题的途径之一,是将描述一个特定事物的数据与操作捆绑到一起,作为程序的基本单元。使得特定的操作只面对特定的数据,而特定数据只有与其相关的操作才能处理。也就是说,把这些特定的数据与对这些特定数据的操作看成一个特定的整体。在这个整体中的操作只对这个整体的数据有效,这样就可以大大减少混乱了。这种将数据与对这些数据的操作看成一个整体的方法叫封装。

建筑工地就是利用“封装”技术的一个最实际的例子。在工地上,如果各个工种的施工人员都可以使用工地上的所有各种建筑材料,那么这个工地肯定是一塌糊涂。所以在管理良好的工地上,管理者利用木工车间、金工车间、混凝土车间、瓦工车间这些概念将不同的建筑材料(相当于计算机中的数据)和不同工种的施工人员(相当于计算机中的操作)各自结合起来分别“封装”(归类)到各个车间。以达到减少混乱的目的。

在程序设计中,如果采用了封装技术,则整个应用程序就由若干个整体(对象)组成,程序的执行过程实质上就是这些整体(对象)在计算机中相继表现自己的过程。对于建筑工地这个例子来说,建筑施工不外乎是各个车间在施工过程中不断地表现自己而已。这就是面向对象程序设计的思想(图 1-4)。说到底,面向对象程序设计的思想就是把数据与操作归类管理分工协作的思想。

目前,有很多支持面向对象思想的程序设计语言,C++就是一种应用广泛并全面支持面向对象思想的程序设计语言。

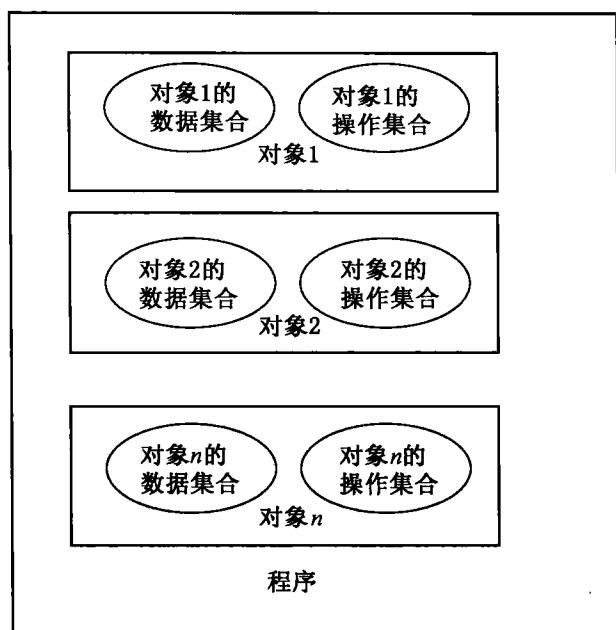


图 1-4 面向对象程序设计的思想

第三节 C++语言概述

一、C++语言的产生和特点

C++语言是由C语言发展演变而来的,C语言是一个优秀的面向过程的程序设计语言,具有语言简洁灵活、运算符和数据结构丰富、程序执行效率高、可以直接访问计算机的物理地址等优点,并且它还有大量的库代码和较多的开发环境,因此C语言得到了极为广泛的应用。

为支持面向对象程序设计,在1980年AT&T贝尔实验室为了模拟的应用开始对C语言进行扩展,于1983年完成并将其命名为C++,C++语言的标准化工作从1989年开始,于1994年制定了ANSI C++标准草案。以后又经过不断的完善,成为目前的C++。

C++的特点主要表现在两个方面:一是全面兼容C;二是支持面向对象的方法。首先,C++是一个更好的C。它保持了C的简洁、高效和接近汇编语言等特点,并对C的类型系统进行了改革和扩充,因此C++比C更安全可靠。由于C++与C保持兼容,这就使许多C代码不经修改就可以为C++使用,用C编写的众多库函数和实用软件可以直接用在C++中。并且由于在C++中引入了类的概念,因此C++全面地支持面向对象的编程方法。

二、一个简单的C++程序

现在,通过一个简单的程序实例简单地说明一下用C++编写的计算机程序结构。

例 1-1 一个简单C++程序。

程序:

```
#include<iostream. h>
void main()
{
    cout<<"Hello!"; //在显示器上输出字符串 Hello!
}
```

这里的 main 是主函数名。在C++程序中,必须且只能有一个名为 main()的主函数,它表示了一个程序执行的起点,两个花括号之间是这个函数的程序部分(一般叫做函数体),可以由多条语句组成,每条语句用分号“;”作为结束符。这个示例程序只有一条语句。语句中的 cout 是一个对象(这是一个C++系统预定义的对象),它代表标准输出设备(计算机的显示器)。符号“<<”是该对象的一个操作,其作用是将操作符“<<”后的内容在显示器上输出。该程序的执行结果为:

Hello!

而#include<iostream. h>指示C++编译器在对程序进行编译时,将文件名为iostream. h文件中的代码程序嵌入到程序中#include指令所在的位置。文件iostream. h中声明了程序所需要的输入输出操作的有关信息。cout和<<的有关信息就是在这个文件中声明的,这个文件是C++系统提供的,用户可以直接使用。

在cout<<"Hello";语句后面由双斜线开始的是注释部分,目的是为了帮助其他用户阅读和理解程序,当然也便于自己日后阅读。

三、程序开发工具的功能

做任何工作都需要工具,编制计算机程序也要使用工具。这个编制计算机程序使用的工具就叫做应用程序开发工具,亦称开发环境(似乎叫开发环境更合适,因为它提供了多种工具)。现在市场上供应的开发工具很多。一个完整的开发工具至少应具备的功能为:源文件编辑功能、程序编译功能、程序文件的连接功能、程序运行调试功能和程序文件管理功能。下面对这些功能做一个简单的介绍。

1. 源文件编辑功能

一般来讲,源文件通常是由程序开发人员用键盘输入到计算机中的,因此开发工具必须具备接受键盘输入且能把输入内容在显示器上显示出来的功能,而且当键入的内容出现错误时可允许用户对其进行修改、编辑。在开发工具中,一般是使用一个文本框来实现这个功能的。

2. 程序编译功能

由于用非机器语言书写的源程序必须变成对应的机器语言程序才能被计算机所执行,因此在开发工具中必须有一个能把用户使用的高级语言程序变成机器语言程序的转

换程序(这个程序一般叫做编译程序)。该功能还应该有在编译过程中发现程序语法错误的能力,这个能力也是评价一个编译器性能的重要指标。

3. 程序文件的连接功能

现代计算机应用程序都是由多个文件组成,这些文件有的是程序开发人员自己开发,有的是系统提供的,只有把这些文件按照程序的要求连接起来才能最终实现计算机的可执行文件(通常其扩展名为: exe)。因此,开发工具必须具有程序文件的连接功能。

4. 程序运行调试功能

一个刚编制的应用程序出现这样或那样的错误是不可避免的,因此开发工具必须具有程序的运行及调试功能,并且能及时地反映调试信息,以使程序设计人员尽快地修正错误。

5. 程序文件管理功能

刚才说到,现代计算机应用程序都是由多个文件组成的,因此,开发工具应该具有使用方便的浏览、创建、删除、改名等文件管理功能。

四、C++程序的开发过程

下面通过例 1-1 的程序简单介绍使用 Visual C++6.0(以下简称 VC 6.0)开发工具开发 C++应用程序的过程。

1. 选择开发工具(环境)

C++开发工具很多,这里,选用应用比较广泛的 VC 6.0。

2. 用 VC 6.0 建立 Workspace

进入 VC 6.0 后,使用 File>New 命令,选择 New 对话框>Projects 选项卡(图 1-5),在该卡上选择 Win32ConsoleApplication(控制台程序)工程,在 Project name 框中添入工程名称,最后按 OK 按钮。

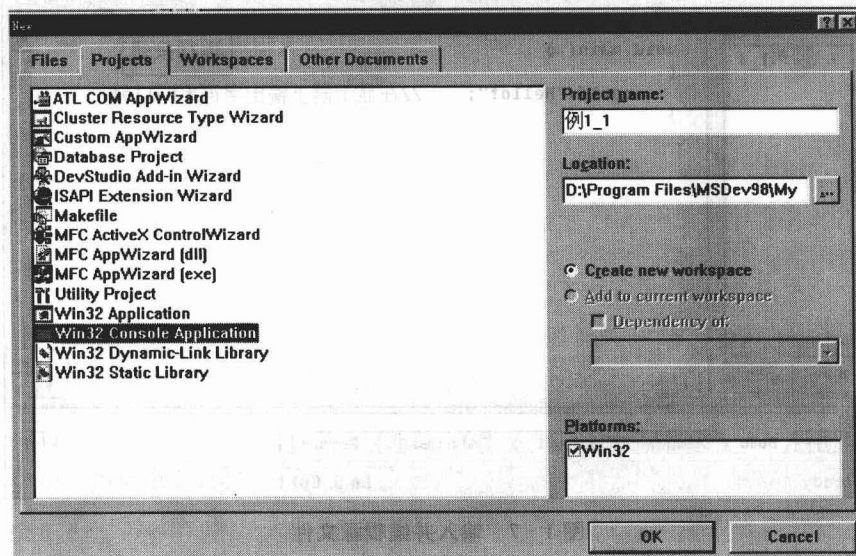


图 1-5 New 对话框的设置