



普通高等院校计算机专业（本科）实用教程系列

C++语言 基础教程

徐孝凯 编著



123456789123123

123456789123123



清华大学出版社
<http://www.tup.tsinghua.edu.cn>



普通高等院校计算机专业(本科)实用教程系列

C++ 语言基础教程

徐孝凯 编著

清华大学出版社

(京)新登字 158 号

内 容 简 介

本书是普通高等院校计算机专业(本科)实用教程系列丛书之一,主要介绍C++语言中常量、变量、表达式、语句、函数等语法成分的定义和使用,数组、字符串、指针、结构、联合、类、文件等数据类型的定义和访问,累加、统计、查找、排序、矩阵运算、数据输入、数据输出等一系列典型程序的设计方法,C++程序结构和 Visual C++ 6.0集成开发环境的使用等内容。本书包含丰富的例题和练习题,并将配有教学辅导和习题解答光盘,以有利于读者自学。

本书是以读者第一次系统学习计算机高级程序设计语言为对象编写的,具有概念定义明确、内容连贯有序、结构层次分明、文字叙述流畅、章节安排合理、深入浅出、方便自学等特点,通过学习本书可以使读者很好地掌握C++语言的基本语法规则,并具备分析和设计一些典型应用程序的能力。本书适合作为普通高等院校各专业开设程序设计语言课程的教材和教学参考书。

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

书 名: C++语言基础教程

作 者: 徐孝凯

出 版 者: 清华大学出版社(北京清华大学学研大厦,邮编:100084)

<http://www.tup.tsinghua.edu.cn>

印 刷 者: 清华大学印刷厂

发 行 者: 新华书店总店北京发行所

开 本: 787×1092 1/16 印张: 21 字数: 492 千字

版 次: 2001年8月第1版 2001年9月第2次印刷

书 号: ISBN 7-302-04632-8/TP·2745

印 数: 5001~11000

定 价: 28.00 元

序 言

时光更迭、历史嬗递。中国经济带着她足以令世人惊叹的持续高速发展驶入了一个新的世纪,一个新的千年。世纪之初,以微电子、计算机、软件、通信技术为主导的信息技术革命给我们生存的社会带来的变化令人目不暇接。软件是优化我国产业结构、加速传统产业改造和用信息化带动工业化的基础产业,是体现国家竞争力的战略性新兴产业,是从事知识的提炼、总结、深化和应用的高智型产业;软件关系到国家的安全,是保证我国政治独立、文化不受侵蚀的重要因素;软件也是促进其他学科发展和提升的基础学科;软件作为 20 世纪人类文明进步的最伟大成果之一,代表了先进文化的前进方向。美国政府早在 1992 年“国家关键技术”一文中提出“美国在软件开发和应用上所处的传统领先地位是信息技术及其他重要领域竞争能力的一个关键因素”,“一个成熟的软件制造工业的发展是满足商业与国防对复杂程序日益增长的要求所必需的”,“在很多国家关键技术中,软件是关键的起推动作用(或阻碍作用)的因素”。在 1999 年 1 月美国总统信息技术顾问委员会的报告“21 世纪的信息技术”中指出“从台式计算机、电话系统到股市,我们的经济与社会越来越依赖于软件”,“软件研究为基础研究方面最优先发展的领域”。而软件人才的缺乏和激烈竞争是当前国际的共性问题。各国、各企业都对培养、引进软件人才采取了特殊政策与措施。

为了满足社会对软件人才的需要,为了让更多的人可以更快地学到实用的软件理论、技术与方法。我们编著了《普通高等院校计算机专业(本科)实用教程系列丛书》。本套丛书面向普通高等院校学生,以培养面向 21 世纪计算机专业应用人才(以软件工程师为主)为目标,以简明实用、便于自学、反映计算机技术最新发展和应用为特色,具体归纳为以下几点:

1. 讲透基本理论、基本原理、方法和技术,在写法上力求叙述详细,算法具体,通俗易懂,便于自学。

2. 理论结合实际。计算机是一门实践性很强的科学,丛书贯彻从实践中来到实践中去的原则,许多技术理论结合实例讲,以便于学习和理解。

3. 本丛书形成完整的体系,每本教材既有相对独立性,又有相互衔接和呼应,为总的培养目标服务。

4. 每本教材都配以习题和实验,在各教学阶段安排课程设计或大作业,培养学生的实战能力与创新精神。习题和实验可以制作成光盘。

新世纪曙光催人向上,催人奋进。江总书记在十五届五中全会上的讲话:“大力推进国民经济和社会信息化,是覆盖现代化建设全局的战略举措。以信息化带动工业化,发挥优势,实现社会生产力的跨越式发展。”指明了我国信息界前进的方向。21 世纪日趋开放的国策与更加迅速发展的科技会托起祖国更加辉煌灿烂的明天。

孙家广

2001 年 3 月

· I ·

普通高等院校计算机专业(本科)实用教程系列丛书

编 委 会

主 任 孙家广(清华大学教授,中国工程院院士)
成 员 (按姓氏笔划为序)

王玉龙(北方工业大学教授)

艾德才(天津大学教授)

刘 云(北方交通大学教授)

任爱华(北京航空航天大学副教授)

幸云辉(北京邮电大学教授)

张海藩(北京信息工程学院教授)

徐孝凯(中央广播电视大学副教授)

徐培忠(清华大学出版社编审)

樊孝忠(北京理工大学教授)

丛书策划 徐培忠 徐孝凯

前 言

C++ 语言是目前世界上最流行和实用的一种计算机高级程序设计语言,它具有丰富的数据类型和各种运算功能,带有庞大的函数库和类库,既支持面向过程的程序设计,又支持面向对象的程序设计,因此是目前进行计算机软件开发的主要工具之一。

同其他所有计算机程序设计语言相比,C++ 语言具有非常明显的优势,正在成为普通高等院校开设程序设计课程的首选语言,因此在这套计算机专业系列教材中把它列为惟一必修的计算机语言课程是正确和明智的选择。

在这套系列教材中,C++ 语言基础课程具有非常重要的地位和作用,它将为数据结构、操作系统、数据库、软件工程、面向对象程序设计、计算机网络等所有后续课程打下坚实的计算机语言和程序设计的基础。C++ 语言的知识将贯穿于所有这些课程之中,使得你的软件开发水平得到不断的提升,最终能够达到用面向对象的观点和程序结构解决实际应用中的软件开发问题。

同社会上已经出版的各种C++ 语言相比,本书具有如下特点:

1. 采用最新、最流行、通用性最强的 Microsoft Visual C++ 6.0 为依据,对C++ 语言的基本内容作了详细的介绍,改变了计算机语言教材落后和脱离现实的状况。

2. 对C++ 语言中的每一种数据类型、运算符、表达式、语句等基本内容,不仅从概念叙述上做到条理清楚、层次分明,而且精心选择针对性强的典型语句或程序段加以解释和阐述,使你能够从概念、理论到应用的结合上加深理解和认识。

3. 本书从训练和提高初学者分析和编写一般应用程序的能力出发,结合介绍分支、循环、函数调用等语句,并介绍数组、字符串、指针、结构、联合、类、文件等数据类型,给出了具有各种实际应用价值的一般典型程序的设计方法,通过这种基本训练后,你不仅能够学会阅读和分析现有的程序,而且能够具有解决实际问题的编程思路和方法,编写出符合规范和性能良好的程序。

4. 每一章后面给出了具有各种题型的大量练习题,以便从各种不同的角度加强你对所学知识的训练和提高。

5. C++ 语言系统庞大,知识点之间像一张网,错综复杂,如何能够按照一般的认知规律,把所有知识点按章节划分为前后有序的一个线性结构,由浅入深、由易到难、循序渐进地组织内容,并使之前后呼应、条理清楚、方便自学,是编写教材成功与否的关键。本书在这方面作了很大的努力,使你能够较轻松地学好本门课程,掌握C++ 语言的基本内容,为学好后续课程打下良好的基础。

6. 本书中所有的C++ 语句、程序段、函数、程序等都在 Visual C++ 6.0 集成开发环境下运行通过,确保它们是正确无误的。

7. 本书由一人创作完成,确保了体系的完整性,前后内容的一致性,编写风格的统一性,避免了由多人创作可能带来的种种弊端。

8. 本书将配有教学辅导和练习解答光盘,主要供教师参考和自学者使用,仍由清华大学出版社出版,不久面市。

本书虽然是为普通高等院校计算机专业开设程序设计语言课程而编写的教材,由于学习它只需要初等数学的知识和对计算机的初步了解,学习目标是掌握C++语言的基本语法规则和分析与编写解决简单通用问题的程序,所以,同样适合作为其他各专业开设C++语言课程的教材。

本书共分为10章,依次为C++语言概述、数据类型和表达式、流程控制语句、数组和字符串、指针、函数、结构与联合、类与对象、类的继承与多态性、C++流等。各章之间内容连贯有序,衔接自然,成为一个有机的整体。

C++语言课程是一门实践性极强的课程,只有多阅读别人的程序,多练习编写自己的程序,多上机调试和运行程序,才能够获得真正的知识。书中所有例题和习题的程序及编程题均可以作为上机题使用。

本课程总学时应安排在80~100之间,其中讲授与上机学时之比应为3:2左右。

承蒙北京大学计算机系孙家骕教授在百忙之中认真审阅了全部书稿,给予了高度评价,并提出了一些修改意见,对此表示衷心感谢!

尽管本人做了很大努力,但由于水平所限,错误和不足之处在所难免,敬请专家和读者批评指正。本人电子邮件地址为:xuxk@crtvu.edu.cn

徐孝凯

2001年4月

目 录

第一章 C++ 语言概述	1
1.1 引言	1
1.2 C++ 字符集	2
1.3 C++ 单词	3
1.4 C++ 语句	5
1.5 C++ 函数	7
1.6 C++ 程序	10
1.7 VC++ 6.0 集成开发环境简介	12
习题一	16
第二章 数据类型和表达式	19
2.1 数据类型	19
2.2 常量	21
2.2.1 整型常量	22
2.2.2 字符常量	23
2.2.3 逻辑常量	24
2.2.4 枚举常量	24
2.2.5 实型常量	25
2.2.6 地址常量	26
2.3 变量	26
2.4 运算符	30
2.5 函数	38
习题二	42
第三章 流程控制语句	50
3.1 if 语句	50
3.2 switch 语句	53
3.3 for 语句	58
3.4 while 语句	67
3.5 do 语句	72
3.6 跳转语句	77
习题三	81

第四章 数组和字符串	90
4.1 数组的概念	90
4.2 数组的定义	91
4.2.1 一维数组	91
4.2.2 二维数组	95
4.2.3 使用 typedef 语句定义数组类型	99
4.3 数组的应用	101
4.3.1 数值计算	101
4.3.2 统计	105
4.3.3 排序	106
4.3.4 查找	109
4.4 字符串	112
4.4.1 字符串概念	112
4.4.2 字符串函数	115
4.4.3 字符串应用举例	118
习题四	121
第五章 指针	128
5.1 指针的概念	128
5.2 指针变量	129
5.3 指针运算	134
5.4 指针与数组	138
5.4.1 指针与一维数组	138
5.4.2 指针与二维数组	140
5.5 引用变量	141
5.6 动态存储分配	143
习题五	146
第六章 函数	151
6.1 函数的定义	151
6.1.1 定义格式	151
6.1.2 定义格式举例	151
6.1.3 有关函数定义的几点说明	152
6.2 函数的调用	155
6.2.1 调用格式	155
6.2.2 调用过程	156
6.2.3 函数调用举例	158
6.3 变量的作用域	162
6.3.1 作用域分类	162

6.3.2 程序举例	164
6.4 递归函数	169
6.5 函数重载	173
6.6 函数模板	174
6.7 函数指针	178
习题六	180
第七章 结构与联合	190
7.1 结构的定义	190
7.1.1 结构定义格式	191
7.1.2 定义格式举例	191
7.1.3 结构使用说明	192
7.2 结构变量的定义和初始化	193
7.2.1 用结构类型名定义变量	193
7.2.2 定义结构类型的同时定义变量	195
7.2.3 定义无名结构类型的同时定义变量	195
7.3 结构成员的访问操作	196
7.4 使用结构的程序举例	197
7.5 结构与函数	206
7.6 结构与链表	210
7.7 结构与操作符重载	213
7.8 联合	221
7.8.1 联合的定义和访问	221
7.8.2 使用联合举例	223
习题七	228
第八章 类与对象	232
8.1 类的定义	232
8.1.1 类的定义格式	232
8.1.2 定义格式举例	233
8.1.3 有关说明	235
8.2 构造函数	240
8.2.1 无参构造函数和带参构造函数	240
8.2.2 拷贝构造函数	245
8.2.3 赋值重载函数	247
8.2.4 构造函数中的初始化表	247
8.3 析构函数	248
8.4 友元函数和友元类	250
习题八	256

第九章 类的继承与多态性	266
9.1 类的继承	266
9.1.1 派生类定义的格式	266
9.1.2 格式举例	267
9.1.3 应用举例	274
9.2 类的虚函数与多态性	277
9.3 类的静态成员	281
9.4 类模板	283
习题九	285
第十章 C++ 流	290
10.1 C++ 流的概念	290
10.2 输入输出格式控制	292
10.2.1 ios 类中的枚举常量	292
10.2.2 ios 类中的成员函数	293
10.2.3 格式控制操纵符	296
10.3 文件操作	298
10.3.1 文件的概念	298
10.3.2 字符文件的访问操作	302
10.3.3 字节文件的访问操作	309
10.4 字符串流	315
习题十	317
附录 ASCII 代码表	322

第一章 C++ 语言概述

1.1 引言

C++ 语言是目前世界上最流行和应用范围最广的一种计算机高级程序设计语言。它从早期的 C 语言逐渐发展演变而来, C++ 对 C 语言不是简单的扩充, 而是从面向过程的语言发展为既面向过程又面向对象的语言, 以适应软件开发技术从面向过程转向面向对象的客观发展的需求。

面向过程是求解问题的一种传统方法, 它把整个问题按功能划分为若干个相对独立的小问题, 每个小问题又可以按功能划分为若干个相对独立的更小问题, 依此类推, 直到最低一层的问题较容易用一种计算机语言编写的程序模块实现为止。在面向过程的程序设计中, 每个程序模块都具有一定的相对独立的功能, 通过较小的程序功能模块的组合就可以形成较大的程序功能模块, 最后形成一个完整的程序。在采用面向过程的方法进行程序设计时, 整个程序的功能是通过程序模块之间的相互调用完成的, 若问题比较复杂, 程序结构即模块之间的调用关系很容易变得复杂和混乱, 并且也容易增加模块之间的依赖性以及调试和修改程序的难度。

面向对象是求解问题的一种新的思路和方法, 它把求解问题中的所有事物(即独立个体)都看做为各自不同的对象, 进而把具有共同特征的对象归属为一个类, 由此得到若干个不同的类, 每个类是对该类事物(对象)的抽象描述, 通过相同或不同类对象之间的相互作用和通信使问题得以解决。

面向过程方法基于对实现某一功能所进行的操作过程的描述, 当功能发生变化时, 哪怕是稍微变化, 都需要重新修改和调试对应的程序模块; 面向对象方法基于对类和对象的描述, 包括对对象属性和进行所有操作的描述, 当对象发生变化时, 可以让原对象保持不变, 再另外建立新对象, 让它继承原对象, 并根据发生的变化定义出新的部分, 此时只需要编写和调试这新增的部分, 对原对象中的代码不需要做任何改变而全部继承下来。这种改良措施更符合人们的思维习惯和解决问题的方法, 同面向过程方法相比, 可以大大减少软件编写和调试的工作量, 提高软件的复用性、可靠性和可扩充性, 缩短软件更新周期。

C++ 语言是对 C 语言的继承、丰富和发展, 既适合开发面向过程的程序, 又适合开发面向对象的程序。无论是利用 C++ 语言开发面向过程的程序, 还是开发面向对象的程序, 都需要掌握 C++ 语言中的一些基本内容, 如数据类型、常量、变量、表达式的含义与使用, 函数的定义与调用, 文件操作, 各种语句的格式与功能, 程序的基本结构等。若在此基础上再深入学习和掌握有关“类”的知识, 就可以利用面向对象的程序设计方法进行软件开发。

由于 C++ 语言系统庞大, 对于初学计算机高级程序设计语言的读者来说, 最好把它分为两个阶段, 作为两门课程学习, 第一阶段主要学习 C++ 语言基础知识, 第二阶段主要学习

与“类”有关的进行面向对象的程序设计知识,这后一阶段最好不要就语言学语言,而要结合数据结构或软件工程课程一起学习效果最佳。

C++ 语言有标准版本,但各软件公司开发的C++ 语言版本并不是严格遵守它,而是与它兼容且稍有修改和扩充。现在普遍使用的是 Microsoft 公司的 Visual C++ 6.0 版本和 Borland 公司的C++ Builder 3.0 及以上版本,它们都是在 Windows 操作系统环境下运行的可视化集成开发工具。本书以 Visual C++ 6.0 为蓝本,向读者介绍C++ 语言的基本内容和进行一般算法设计的知识,使读者能够进行模块化和结构化的面向过程的程序设计,以及较简单的面向对象的程序设计,为学习后续数据结构课程和进行更复杂的算法设计奠定基础。

自然语言(如汉语、英语等)是人们进行交流的工具,它的基本符号是字或字符,由字(字符)构成基本词法单位——单词,由单词构成句子,由句子组成文章。计算机语言同自然语言一样,也具有字、词、句、章的体系结构。如对于C++ 语言来说,它具有规定的字符集,由一个或若干个字符按照词法规则构成C++ 单词,由一个或若干个单词按照语法规则构成C++ 语句,而完成某一功能的一条或若干条语句被定义为一个程序模块——函数,由一个或若干个函数以及一些其他的语法成分构成一个独立的程序文件,由一个或若干个程序文件构成一个完整的C++ 程序。

1.2 C++ 字符集

下面分类列出在C++ 语言中规定的全部字符。

1. 大、小写英文字母(52 个)

A~Z, a~z

注意: 在 C/C++ 语言中,同一字符的大写和小写被视为不同的字符,这与其他语言中的规定不同。如 Abc 和 abc 或 ABC 是完全不同的标识符。

2. 十进制数字符号(10 个)

0~9

3. 标点符号(8 个)

, 逗号	// 数据之间的分隔符
; 分号	// 简单语句结束符
' 单引号	// 字符常量起、止标记符
" 双引号	// 字符串常量起、止标记符
: 冒号	// 语句标号结束符或条件运算符
空格	// 语句中各成分之间的分隔符
{ 左花括号	// 复合语句的开始标记
} 右花括号	// 复合语句的结束标记

4. 单字符运算符(19个)

- (左圆括号 // 简称左括号,同右圆括号配对使用,用于表达式和函数运算
-) 右圆括号 // 简称右括号
- [左中括号 // 左、右中括号必须配对使用,用于数组元素访问
-] 右中括号
- + 加号或正号
- 减号或负号
- * 乘号或间接访问运算符
- / 除号
- % 取整余数
- . 小数点或结构成员访问符
- < 小于号或左尖括号
- = 赋值号
- > 大于号或右尖括号
- ! 感叹号 // 逻辑非运算符
- ~ 波折号 // 按位取反运算符
- & // 取地址或按位与运算符,同时又是引用说明符
- ^ 尖字符 // 按位异或运算符
- | 竖线 // 按位或运算符
- ? 问号 // 条件表达式运算符

5. 特殊用途符号(3个)

- # 井字符 // 预处理命令行的开始标记
- \ 反斜线 // 转义字符序列的开始标记
- _ 下划线 // 只用于标识符中

注意:在字符串常量(简称字符串)中能够使用任何 ASCII 码字符。如 \$ 不属于 C++ 字符,但可以使用在 C++ 字符串中。另外,在汉字操作系统的支持下,任何汉字区位码字符均可以使用在 C++ 字符串中。如 "汉字"、"24.56 \$"、"a + b ="等都是合法的字符串。

1.3 C++ 单词

由 C++ 字符按照一定的组词规则可以构成各种 C++ 单词,可以把 C++ 单词分为以下五类。

1. 保留字

保留字是 C++ 系统预定义的、由小写英文字母组成的单词、词头或词组,VC++ 6.0 中

的全部保留字如下:

auto	bool	break	case	char	class	const
continue	default	delete	do	double	else	enum
extern	false	float	for	friend	goto	if
inline	int	long	new	operator	private	protected
public	register	return	short	signed	sizeof	static
struct	switch	template	this	true	typedef	union
unsigned	virtual	void	volatile			

每个保留字都被系统赋予了一定的含义,具有了相应的功能,所以用户不能再将它们作为非保留字使用。

另外,在预处理命令中,其命令关键字虽然不算作C++保留字,但也最好把它们看做为C++保留字,不要使用它们作为其他用途,以免引起混乱。这些命令关键字有 include、define、ifdef、ifndef 等。

2. 标识符

标识符是用户在程序设计中给特定量所起的名字。在C++语言中规定:每个标识符必须是由英文字母、十进制数字符号和下划线组成的一串字符,并且第一个字符必须是英文字母或下划线。每个标识符中的字符数可以任意,但只有前 32 个字符有效。通常,一个标识符由 1~32 个字符所组成。如 a、ab、size、Max、x1、y25、fun_1 等都是合法的标识符,而 3xy、“work”、lable:、Hi-4、list length 等都是非法的标识符,因为第一个标识符以数字开头,其余四个标识符中均使用了非法字符。

给一个特定量命名一个标识符时,为了便于记忆和阅读,最好使用该特定量的英文或汉语拼音作为标识符,有时将第一个字母大写,有时使用下划线连接两个英文或拼音单词。如可以用 wages、wage、Wage、Gongzi 等表示工资,用 Name、xingming、XM 等表示姓名,用 maxWage、MaxWage、max_wages 等表示最高工资。

在C++程序设计中,变量、对象、符号常量、用户定义的数据类型、函数等的名字都是需要用户定义的标识符。

3. 常量

常量分为数值常量、字符常量和字符串常量三类。日常使用的十进制常数可以直接作为C++数值常量使用。如 32、-128、3.26、+100、-50.718 等都是合法的C++数值常量,简称常数。

字符常量就是单个 ASCII 码字符,表示时为了把它同单个字符的标识符或数值相区别,必须用单引号括起来。如 'a'、'B'、'+'、';'、'5' 等都是字符常量。使用以单引号括起来的转义字符序列也可以表示字符常量,特别是用来表示像回车、换行等控制字符常量,这些将在以后介绍。

字符串常量就是由 ASCII 码字符和汉字区位码字符组成的一串字符,同样,为了把它同其他语法成分相区别,表示时必须用双引号括起来。如 "a + b =", "main:", "x, y, z =", "Not found", "length of table", "1. 输出线性表的长度"等都是字符串常量,简称为字符串。

由以上讨论可知:a、'a'和"a"是完全不同的,它们分别为一个标识符、一个字符常量和一

个字符串。

4. 运算符

运算符是对数据进行运算的符号。C++ 运算符有单字符运算符,如 +、-、*、/等,有双字符运算符,如 <=、!=、->、++、&&、*= 等,也有三字符运算符,如 <<=、>>= 等,另外还有三个保留字运算符 new、delete 和 sizeof。以后将详细列出所有 C++ 运算符,并一一介绍其具体功能。

由运算符和操作数可以构成各种表达式,对表达式进行计算的结果通常得到一个确定的值。如 $25 * 4 + 15$ 就是一个数值表达式,其求值结果为 115。当然一个单独的操作数也可以看做为一个表达式,因为它本身就是一个值或对应一个值。如一个常数 25,一个变量 x,一个字符 '!',一个字符串 "apple" 等操作数都是表达式,不过它们为最简单的表达式。

5. 标点符号

C++ 字符集中列出的每个标点符号可以单独作为 C++ 单词使用,作为一个语法成分出现在语句中。

1.4 C++ 语句

由 C++ 单词按照一定的语法规则排列起来就形成语句。虽然每种语句的语法规则不同,但除了复合语句外,最后都必须以分号结束。下面的每一行都是一条 C++ 语句,当然其前面的编号除外。

- (1) int x;
- (2) x = 20 * 35 - 6;
- (3) if(x >= 100) cout << x;
- (4) break;
- (5) typedef int DataType;
- (6) void Sort(int aa[], int nn);
- (7) Sort(a, n);

在第一条语句中,含有 4 个单词(即语法成分),依次为表示整型的保留字 int、空格、表示变量的标识符 x 和分号。在第二条语句中含有 8 个语法成分,其中为 1 个变量标识符,3 个运算符,3 个常数和 1 个分号。第三至第七条语句中,分别含有 10 个、2 个、6 个、15 个和 7 个单词,请读者验证。

按照语句功能,可以把 C++ 语句分为以下八类:

1. 类型定义语句

类型定义语句又称类型说明语句,用来定义系统预定义类型之外的、用户需要使用的数据类型。如结构、联合、枚举和类类型都需要用户结合应用情况具体定义。

2. 变量定义语句

变量定义语句又称变量说明语句,用来定义程序中需要使用的属于某个类型的变量。如上述第一条语句就定义了整型变量 `x`,其中 `int` 表示系统预定义的整数类型,`x` 为一个变量标识符,以后可以用它来表示(即保存)一个整数。在上述第二条语句中,用 `x` 保存一个整数 694,它是赋值号右边表达式 $20 * 35 - 6$ 的值。

3. 函数原型语句

它是用以声明一个函数存在并指定调用格式的语句。通常情况下,函数原型语句出现在一个程序或程序文件的开始部分,以便在其后的函数中能够调用该函数,而该函数原型语句所对应的函数定义可以出现在整个程序中的任何位置,甚至可以不在同一个程序文件中。如上述第六条语句就是一条函数原型语句,调用该函数时要使用两个参数,一个为整型数组,另一个为整型数,该函数执行后不返回任何值。

4. 表达式语句

任何一个C++ 表达式后加上一个语句结束符分号就成为一条语句,称此为表达式语句。最常用的表达式语句为赋值表达式语句和函数调用表达式语句。上述第二条为赋值表达式语句,第七条为函数调用表达式语句。

5. 复合语句

用花括号括起来的语句序列合起来称为一条语句,即复合语句。如 `{int x=3,y=4; y=x+2*y;}` 就是一条复合语句,它包含有两条简单语句。一条复合语句中可以包含任意多条语句,包括不含任何语句,并且每条语句可以是任何种类的语句,包括仍可以是复合语句。如 `{},{},{x=40;}` 等都是合法的复合语句,其中第一条中不含有语句,第二条中含有一条空语句,即只有分号的语句被称为空语句,它也是一条合法的语句,第三条中含有一条简单语句。

注意: 复合语句是以左花括号作为开始标记,右花括号作为结束标记的,其后不需要使用分号,若误用分号,编译系统则认为是后接一条空语句。如“`{int a; a=1;};`”为两条语句,前者为复合语句,后者为空语句。

由于复合语句也是一条语句,所以,以后所说的语句均指所有种类的语句,当然包括复合语句在内。

6. 选择语句

选择语句具有根据条件选择是否执行某条语句的功能,或者具有根据条件从多个入口点中选择某个入口点执行的功能。如上述第三条语句就是一条选择语句,它根据条件 `x >= 100` 是否成立来决定是否执行把 `x` 的值输出到屏幕上的操作,若条件成立则输出,否则不输出。