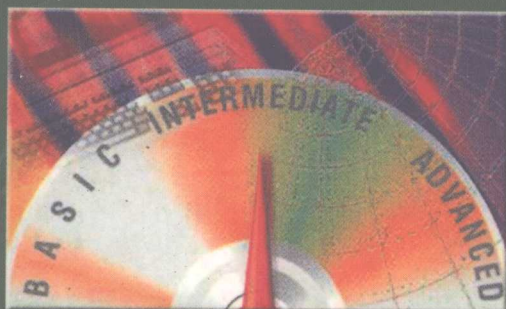


印度优质 IT 职业教育教学用书



C 语言编程

Comp-U-Learn Tech India Ltd.

贾素玲 王 强 郑晋梅 韩小汀 编译



高等教育出版社
HIGHER EDUCATION PRESS

内容提要

印度优质 IT 职业教育教学用书

C 语言编程

Comp-U-Learn Tech India Ltd.

贾素玲 王强 郑晋梅 韩小汀 编译

封面设计
封面文字

行式照出
址 社
何制照融
册 总

册 登
册 册
册 册
本 天
册 册
册 册

册 册
册 册
册 册
册 册

高等教育出版社

内容提要

本书是引进的印度优质 IT 职业教育教学用书,是 Comp-U-Learn Tech India Ltd. 的培训课程用书,与 COMP-U-LEARN 软件开发职业资格证书相配套。

本书共有 12 章,主要包括:C 语言的来源与简介、数据类型和运算符、C 语言中的控制语句、数组、C 语言中的函数、指针、字符串、数据结构、C 语言中的文件处理、存储类型和预处理程序、位运算符等。同时本书还配有相应的指导手册。

本书适用于高等职业学校、高等专科学校、成人高校、本科院校举办的二级技术学院,也可供示范性软件职业技术学院、继续教育学院、民办高校、技能型紧缺人才培养、培训使用,还可供本科院校师生、计算机从业人员和爱好者参考使用。

图书在版编目(CIP)数据

C 语言编程/Comp-U-Learn Tech India Ltd.;贾素玲等编译. —北京:高等教育出版社,2005.4

ISBN 7-04-017175-9

I. C... II. ①C... ②贾... III. C 语言 - 程序设计 - 高等学校:技术学校 - 教材 IV. TP312

中国版本图书馆 CIP 数据核字(2005)第 018105 号

策划编辑 冯 英 责任编辑 彭立辉 封面设计 张 楠
版式设计 王艳红 责任校对 殷 然 责任印制 韩 刚

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 010-58581000

经 销 北京蓝色畅想图书发行有限公司
排 版 高等教育出版社照排中心
印 刷 北京市鑫霸印务有限公司

购书热线 010-58581118
免费咨询 800-810-0598
网 址 <http://www.hep.edu.cn>
<http://www.hep.com.cn>
网上订购 <http://www.landaco.com>
<http://www.landaco.com.cn>

开 本 787×1092 1/16
印 张 17
字 数 370 000

版 次 2005 年 4 月第 1 版
印 次 2005 年 4 月第 1 次印刷
定 价 27.70 元

本书如有缺页、倒页、脱页等质量问题,请到所购图书销售部门联系调换。

版权所有 侵权必究

物料号 17175-00

The Team

Production/Series Supervision

Jambu Krishnamurthy

Vice President-Product Development

The Head

Sridhar Subramaniam

Head-Training Solutions

编审委员会

主任 陈 明
副主任 董 葛 尹 洪 贾素玲
委员 (中文姓氏按拼音顺序排序)
Vikram Gupta (印度)
Roxanne Ang (加拿大)
冯 英 贾俊义 刘 超
倪金生 任树森 万世基
王克勤 王 强 王人骅
王树英 许 珂 严 亮
杨小平 张学群 郑晋梅
朱永庆

前言

这是一套难得的 IT 教材，因为它是我们师徒几个千辛万苦从西天取回的“真经”。

众所周知，印度是当今世界计算机软件出口王国，但上帝为何如此偏爱，把这个光环戴在了她的头上，人们并不十分清楚。一位印度朋友告诉我，印度软件业的发展之所以令世人赞叹，其原因除政府长期实施一系列扶持政策外，主要是得益于行之有效的培训教育。这一点，在我们对印度考察时得到充分的印证。为贯彻全国职教会关于“积极引进国外优质职业教育资源”的精神，教育部所属中国华育发展总公司和北京华育发国际技术培训有限公司，于 2003 年组织专家对印度进行了为期两周的考察学习，先后走访了印度软件行业协会、4 所大学以及 7 家软件培训机构。通过实地考察、分析和比较，结合中国软件产业发展情况及软件教育培训现状，专家认为 Comp-U-Learn Tech 印度有限公司 (Comp-U-Learn Tech India Ltd.) 的培训课程比较适合我国职业技术学院计算机软件专业教学，经研究最终决定引进 Comp-U-Learn 系列教材。

Comp-U-Learn Comp-U-Learn 于 1993 年在美国密歇根州成立，是北美开展计算机软件职业培训的先导者，在培训中形成了独特的 ICAM 教学理念，已有上万名的学员在财富 500 强、IT 行业和其他公司从事相应的工作。Comp-U-Learn Tech 印度有限公司也有超过 2 万 5 千多名学生受益于 Comp-U-Learn 的 IT 培训计划。Comp-U-Learn 的教材和课件由美国统一研发，并在全球推广。Comp-U-Learn 的软件开发职业资格证书在美国、印度、加拿大等国家业内得到广泛认可。

课程设置 本套教材的形成，一是遵循教育部《两年制高等职业教育计算机应用与软件技术专业领域技能型紧缺人才培养指导方案》的精神，以培养技术应用人才为目标，力求在课程和课时设置上与高职院校的教学计划无缝衔接；二是以企业对人才的需求为依据，把软件工程的思想完全融入教材体系中，将基本技能培养和主流技术相结合，课程设置中重点突出、主辅分明、结构合理、衔接紧凑。据此，我们对 Comp-U-Learn 的数本教材进行了精选和组合，最终完成以 Java 开发为主线的职业教育课程设置。

按照循序渐进的方式，课程设置分为 6 个部分，其中计算机基础类有《计算机基础》，语言基础类有《C 语言编程》、《Java 编程基础》、《HTML 设计》，数据库基础类有《基于 Oracle 的数据库系统》，分析工具有《UML 系统分析设计》，Java 进阶类有《高级 Java 编程》、《JSP 开发技术》、《JavaScript 程序开发》、《XML 核心技术》，企业级开发类有《EJB 企业级应用开发》、《J2EE 技术实践》，共计 12 本教材。这样的课程设置，构成了一个完整的 Java 培训体系结构，它从最基本的《计算机基础》入手，渐进到当今最流行的《J2EE 技术实践》，可使初学计算机的人在学完本套教材后，成为熟练使用各种 Java 工具的企业级开发专业人才。

教材特点 凡阅读过 Comp-U-Learn 教材的朋友,都会有相似的感觉,无不佩服作者面对复杂技术的实践体系,运用巧妙的方式而设计的编排模式。每章开始均列有学习目标和本章内容,学习目标可作为评估学生掌握本章知识的参考标准;本章内容实际上为教学重点,既有助于教师备课,又有助于学生掌握要点。每章的结尾都配备有复习题,使得学生能够在学习的基础上进一步思考,巩固前一阶段所学的知识,以便得到知识的升华。每本教材最后还添加了附录和词汇表,以便学生扩展相关主题的知识面,帮助学生梳理知识结构,同时也为学生展示了新的探索和发展空间。

侧重培养学生的实战操作能力是该系列教材的另一特点。在“学”和“思”的基础上,本教材还增加“练”的环节,除《J2EE 技术实践》外,每册教材都配有单独的案例指导手册,并附有完整的解决方案,旨在通过项目实践,增强学生的编程能力,使知识从书本中释放出来,转化为专业技能。

此教材还有一个特点是课件配套齐全。不仅有学生用书、案例指导手册,还配备有教师授课用 PPT、考试题库和对应的职业资格认证。PPT 可用来帮助教师教学和引导学生深刻理解每章主题、轻松完成相关实例、准确掌握概念定义。职业资格认证对应教学的内容,分为 CUL 软件编程助理工程师、软件编程工程师和软件开发工程师。学生考试合格后,一旦取得证书,就等于获得了迈入 IT 行业的绿色通行证(详细信息可登录 www.culnet.net.cn 或 www.ceditt.com)。

译者 为使这套引进的教材能原汁原味地呈现给读者并更好地服务于职业教育,我们组建 CED-CUL 学术委员会,制订了针对性的编译方案。本套系列教材是由北京航空航天大学的贾素玲教授、王强副教授主持编译。他们从事多年的信息系统开发,在信息技术领域有着广泛的涉猎和独到的见解。他们严谨的学术态度和精深的专业造诣,保证了本套系列教材的翻译质量。从译稿到审稿,无不凝聚着几位教授的心血。

参加该系列教材翻译工作的老师还有:许珂、陈当阳、罗昌、张成、张剑、韩小汀、卢根、姚琪琳、张国强、王卫星、柴庆慧、朱磊、姜浩、张小强、孙海涛、武心清、孔令鹏、傅玉、郝娟君、姚冠扬、潘雪梅、赵宏伟等。首都师范大学计算机教育硕士赵俊莉对全套教材进行了校阅,对可能存在的问题进行了精心的修饰。他们一丝不苟的工作精神和相互间的密切配合,特别值得赞扬。在此,对他们卓有成效的工作表示感谢!

本套教材的翻译工作得到了教育部高教司、教育部职成司、北京航空航天大学、中国软件行业协会教育与培训委员会、北京软件产业促进中心等单位领导和相关人士的指导,同时还得到了 Comp-U-Learn Tech 印度有限公司亚洲首席代表 Vikram Gupta 先生的大力配合和支持,在此一同对他们表示真诚的谢意!我要特别感谢中国社会科学院王树英教授、教育部教育发展研究中心培训中心朱永庆教授对引进这套系列教材所付出的辛苦。感谢北京大学计算机软件工程硕士许珂和太原理工大学高级工程师郑晋梅在课程设置工作中做出的贡献。感谢中国地质大学陈明教授、北京航空航天大学王人骅教授、刘超教授、首都经济贸易大学张学群教授、中国人民大学杨小平教授对编译工作的指导,感谢张璠、邱钦伦两位硕士真诚的关心和支持。这里我

们还要特别感谢高等教育出版社为这套系列教材的出版给予的大力支持。

在编译过程中，我们虽然力求准确，尽量保持原文的特色，但限于时间仓促，难免存在失当，恳请各位同仁和读者不吝赐教。

希望每位学员能从本套教材中汲取丰富的营养，预祝每位学员能尽快加入令人羡慕的 IT 行业中！一旦进入了这个五彩缤纷的世界，将会感谢本套系列教材给你带来的一切。

同学们，这里是信息化高速公路起跑线，这里是人生旅途的转折点，让我们共同点击“开始”吧！



北京华育发国际技术培训中心

二〇〇五年元月

I 开始之前

随着时代的发展,人、国家和社会都发生了很大变化,这些变化改变了人们的行为并重新定义了规则。这些变化虽然在自然界中非常少,但是却改变了自然界。它们不仅改变了自然界的行爲,也改变了人们的思维方式。就像计算机对于商业一样,C语言在计算机程序语言中起着重要的作用。

C语言诞生于贝尔实验室,由Dennis Ritchie开发。C语言从诞生以来已走过了一段漫长而伟大的历程。

今天,凡是C语言存在的地方,就连Ritchie本人也会为其如此广泛的应用而感到震惊。与初始C语言相比,现在C语言的形式已经有了很大的发展。

由于C语言具有功能强大等特点,因此它的出现使其他传统语言受到了挑战。从MS-DOS、UNIX到Windows,每一个操作系统都是用C语言写的,C语言在信息高速路上几乎建立了每一个里程碑。

Clipper、DBASE3+的数据库都是用C语言书写的,我们用到的每一个编译器也都是用C语言写成的,每个微处理器都是因为C语言才使它们得以工作。以C++为例,如果C语言的革命没有发生,那么它就不可能诞生,Java也是同样的道理。新时期两个最有影响力的革命是——OOPS和C。

如果不能够进入C的世界尽情遨游,它的力量和光辉就不能使人们折服。但是一旦进入了这个世界,就会感谢C语言所带来的一切;它可以使人们在信息高速路上狂奔。

II 如何使用本书

本书在复杂的技术主题方面提供了清晰明了、条理清楚的陈述说明。考虑到读者群大多为初学者,我们采用了最有效的方式对书中内容的结构和模式进行了编排。

本书共分12章,列出了本书的大纲及内容。每一章包含12个主题并根据每章开始列出的主题来进行课堂教学。这样有助于读者在每堂课前了解该章所学的内容;而且各个主题都罗列在每章一开始的“本章内容”标题下。

本书每一章开始都有目标,这些目标列在每一章首页上,这样有助于读者了解对每一章所学知识的掌握程度,更好地理解每一章的相关主题。此外,一些被引用的例子有详细的说明,这样方便读者更好地掌握概念的定义。建议本书读者试着独自完成所学章节中的实例,以便更准确地理解概念。本书在每一章的最后还提供了复习题,建议读者在课后认真完成。

此外,本书还配有指导手册,针对每一章给出了案例题目,希望读者自行完成该练习。如果在完成题目时遇到困难,可以参考给出的答案。

为了得到相关主题的更多信息,本书还添加了附录。该部分提供主题外的额外内容,通过扩展相关主题的内容,增强读者的编程能力。

本书最后的索引帮助读者了解常用技术术语的意义及其应用。

III 本书编写目标

本书旨在教读者编程。初学计算机语言的读者往往比较习惯于先学习 C 语言,这本书在讲述 C 语言的同时还指导读者如何写一个标准的程序。

完成这本书的学习以后,读者将能写出具有良好设计和良好形式的 C 语言程序。而且,也将能够理解更高级的概念,如指针和链表等。

IV 本书使用惯例

每一章有 12 个主题,称之为 12 点编程(12PP),在每一章中,每一主题(节)都用较大的字体进行强调。

需要注意的问题列在框内,它是了解主题所必须知道的信息,而不是内容的一部分。

一系列应做和不应该做的事项放在每一章 12 点的最后一点。

每一章最后进行总结,用特定的表显示。

每一章用“复习题”作为结束,有助于概括本章内容。

目

I 开始之前	I
II 如何使用本书	I
III 本书编写目标	II
IV 本书使用惯例	II

第一章 C 语言的来源与简介	1
1.1 关于 C 语言	3
1.2 C 语言的发展	3
1.3 程序语言概述	3
1.4 C 语言的结构	4
1.5 编译一个 C 程序	4
1.6 编译器和解释器	6
1.7 通过伪代码建立逻辑	6
1.8 一个简单的 C 程序	7
1.9 C 语言中用到的字符集	8
1.10 C 语言中的关键字	9
1.11 运算符的优先级	9
1.12 应该做的与不应该做的	10
复习题	10

第二章 数据类型和运算符	13
2.1 基本的数据类型	15
2.2 各种基本数据类型的限定符	16
2.3 C 语言中的变量	16
2.4 类型声明	17
2.5 输出函数	18
2.6 输入函数及格式限定符	18
2.7 算术运算符	19
2.8 一元运算符	20
2.9 关系和逻辑运算符	21

录

2.10 C 语言中的常量	22
2.11 一个 C 程序	23
2.12 应该做的与不应该做的	24
复习题	25

第三章 C 语言中的控制语句	27
3.1 为什么要控制语句	29
3.2 if 语句	30
3.3 if else 语句	31
3.4 for 语句	34
3.5 while 循环语句	35
3.6 do while 语句	36
3.7 break 语句	37
3.8 continue 语句	38
3.9 switch 语句	38
3.10 goto 语句	40
3.11 三目运算符	41
3.12 应该做的与不应该做的	42
复习题	42

第四章 数组	45
4.1 数组简介	47
4.2 数组的优势	47
4.3 数组的类型	48
4.4 数组声明	48
4.5 数组初始化	48
4.6 从数组中获取数据	50
4.7 使用数组的简单例子	51
4.8 数组的存储方式	52
4.9 多维数组	52
4.10 字符数组	55

C 语言编程

4.11 数组溢出	57
4.12 应该做的与不应该做的	57
复习题	58

第五章 C 语言中的函数

5.1 函数介绍	61
5.2 函数的优势	61
5.3 声明函数	61
5.4 调用函数	63
5.5 变量	63
5.6 向函数中传递参数	64
5.7 使用函数的简单程序	67
5.8 函数嵌套	68
5.9 向函数中传递数组	68
5.10 递归函数	70
5.11 传值调用与传址调用	71
5.12 应该做的与不应该做的	73
复习题	74

第六章 指针

6.1 指针简介	77
6.2 初识指针	77
6.3 指针操作	78
6.4 动态存储分配	80
6.5 指针和数组	81
6.6 指向指针的指针	82
6.7 指向函数的指针	83
6.8 返回指针的函数	84
6.9 指针和二维数组	85
6.10 指针数组	86
6.11 对于指针声明的补充说明	86
6.12 应该做的与不应该做的	87
复习题	87

第七章 字符串

7.1 字符串简介	91
7.2 字符串	91

7.3 字符串的内部表示形式	92
7.4 更多内容	93
7.5 strlen()	94
7.6 strcpy()	95
7.7 strcat()	96
7.8 strcmp()	97
7.9 向函数中传递字符串	97
7.10 二维字符数组	98
7.11 指针数组	100
7.12 应该做的与不应该做的	101
复习题	101

第八章 数据结构-I

8.1 结构体介绍	105
8.2 基础结构体	105
8.3 声明并访问结构体	105
8.4 结构体的内部表示	107
8.5 结构体数组	108
8.6 向函数传递结构体	109
8.7 结构体中的指针	110
8.8 嵌套结构体	111
8.9 共用体	112
8.10 枚举数据类型	112
8.11 typedef 声明	113
8.12 应该做的与不应该做的	114
复习题	114

第九章 数据结构-II

9.1 简介	119
9.2 单链表	119
9.3 双链表	122
9.4 循环链表	124
9.5 栈	126
9.6 队列	127
9.7 二叉树	128
9.8 排序技术-I	131
9.9 排序技术-II	134
9.10 检索技术	138

9.11 哈希表	139
9.12 应该做的与不应该做的	140
复习题	140
案例学习	141

第十章 C 语言中的文件处理 147

10.1 文件处理简介	149
10.2 文件的打开	149
10.3 文件的读取	150
10.4 文件的关闭	150
10.5 错误检测	151
10.6 文件的打开方式	151
10.7 文件的写入	152
10.8 argc 和 argv	153
10.9 字符串输入	154
10.10 格式化的磁盘 I/O	155
10.11 错误检测函数	155
10.12 应该做的与不应该做的	156
复习题	157

第十一章 存储类型和预处理

程序 159

11.1 简介	161
11.2 自动存储类型	161
11.3 寄存器存储类型	162
11.4 静态存储类型	163
11.5 外部存储类型	164
11.6 宏	165
11.7 带参数的宏	166
11.8 #include 指令	166
11.9 #if 和 #elif 指令	167

11.10 #undef 和 #pragma 指令	168
11.11 类型转换	169
11.12 应该做的与不应该做的	170
复习题	170

第十二章 位运算符 173

12.1 简介	175
12.2 位逻辑运算符	175
12.3 反码	176
12.4 右移运算符	177
12.5 左移运算符	177
12.6 按位与运算符	178
12.7 按位或运算符	179
12.8 异或运算符	180
12.9 文件包含	181
12.10 C - ISAM	181
12.11 位段	182
12.12 应该做的与不应该做的	183
复习题	184

术语表 185

附录 191

附录 A ASCII 码表	191
附录 B 标准库函数	192
附录 C 控制语句表	196
附录 D COMP - U - LEARN 证书样本	198

词汇表 199

第一章

C 语言的来源与简介

完成本章学习,你将能了解:

-  C 语言的发展
-  程序语言概述
-  C 语言的结构
-  编译器和解释器
-  C 中的字符集、关键字和运算符的优先级

本章内容

- 关于 C 语言
- C 语言的发展
- 程序语言概述
- C 语言的结构
- 编译一个 C 程序
- 编译器和解释器
- 通过伪代码建立逻辑
- 一个简单的 C 程序
- C 语言中用到的字符集
- C 语言中的关键字
- 运算符的优先级
- 应该做的与不应该做的

1.1 关于 C 语言

C 语言是许多其他高效率程序语言的基础。该语言是面向过程语言的先驱,而且在该领域占有绝对统治地位。C 语言最重要的特性是它既适合编程新手,同时也能激起老手的兴趣。这就是为什么在 C 语言已经发展了 30 年后,人们仍然对其不断进行研究的原因。

多年来,C 语言一直是人们关注的焦点,其原因之一就是其所提供的多样性。例如,用 C 语言编写的程序仅做很小的改动或者在完全不改动的情况下即可从一台机器移植到另外一台机器上,非常方便。在高级语言之中,C 语言是一种主要的结构化程序语言。

C 语言通常被描述为一门中级语言,它的书写风格与其他的高级语言类似,如 FORTRAN、COBOL、PASCAL 语言。它们之间的不同在于 C 语言允许与计算机内部进行隐秘的联系,同时 C 语言也提供了无数的运算符和数据类型,使程序员在编程的时候有更多的选择,这样就可以以尽可能少的努力解决更多的问题。

1.2 C 语言的发展

C 语言能避免被新一代计算机语言冲击的原因是它技术上的先进性,它由 AT&T 的贝尔实验室的一个名叫 Dennis Ritchie 的程序员开发。C 的概念起源于一个原始的 C 的形式,它的名字叫做 Combined Programming Language (BCPL),即联合程序语言,该语言由贝尔实验室的 Ken Thompson 开发,它被称之为 B。C 语言被认为是一个改进的,更具适应性的 B 的继承。C 比 B 高级的原因在于它提供了更广范围的数据类型,包括的基础类型如字符型、整型浮点型等。

C 语言的简洁性和一致性归结于一个事实,就是它是一种人的语言。这可能是解释它的可靠性、简单性和易用性的原因。虽然它已经与 UNIX 操作系统联系起来,但是它并没有专用于任何一个操作系统和环境。

正如自然界的生存法则一样,所有非凡的事情必有某些失常之处,C 语言也没有违背这个法则。当遇到文件处理的时候,出现了一些退步。C 语言另一个小的局限是没有任何输入输出的操作命令,这些功能的完成必须借助于使用标准 C 的函数。尽管有这些瑕疵,但是因为它友好的环境和可靠性,C 语言仍然很流行。这就是为什么程序员喜欢这门语言的原因。

1.3 程序语言概述

当两个或者更多的人相互对话的时候,会在他们之间发生一件很普遍的事情——语言。

语言与呼吸的空气和吃的食物一样重要,因为它是一种人们不能不用的东西。同理,因为计算机是一个机器,它不能理解人们的语言,因此必须有能与其交互的方式。它需要一个指导其工作的媒介,人们必须能够懂得这种方式,且能够通过这种方式告知相关的意图。

有众多的语言可以用于计算机编程,最原始的应该是机器语言。机器语言是一个含义模糊的、系统的分类,它可以直接操纵计算机的硬件。对计算机运行一个专门的操作时,必须植入一系列的声明。现在极少用到机器语言,因为一系列的因素,使其工作起来非常费功夫,而且每一个计算机有它自己的指令系列。目前,用于与计算机沟通的语言多为高级语言。高级语言优于机器语言的原因在于高级语言中一个单独的语句可以代替机器语言中一系列的指令。高级语言具有很高的可移植性,例如,在一台机器上运行的程序可以只做很少的改动或者完全不用改动就可以在另外一台机器上运行。

在高级语言中,以语言的形式给出计算机指令,或者在另外一种简单的形式下执行一系列的指令。

因此,程序可以被定义为用一套声明和函数运行一系列特定操作的过程。

1.4 C 语言的结构

在对 C 语言有一个基本了解后,现在来了解一下 C 语言的结构。对 C 语言的结构进行深入了解,对于写好 C 程序是非常必要的。

一个典型的 C 程序包含 3 个主要的部分:预处理指令、全局声明、函数。

C 语言虽然有 3 个主要的分类,但结构也不能被严格地限制在这种分类。用预处理指令开始的程序,能够用在全局变量声明,然后细化每个函数体内的代码。如果你对 C 语言编程还不是很熟悉,那么使用这个过程,会使人们更加清楚 C 语言的编程过程。

1. 预处理指令

在一个 C 程序的第一行是预处理指令。它连接到标准库,从标准库里可以使用一些预先定义好的函数。例如,为了打印一些特殊的数据,不需要再专门为其写一个单独的函数,只要连接标准库就可以使用这个函数(因为打印函数已经存在库里面)。这个调用过程的顺利完成需要预处理指示的帮助。

2. 全局声明

通过定义一个变量,可以在程序中使用该变量来存储数据,编译器将根据这个变量的属性来工作,例如变量的名字、类型和初始值等。全局变量的意思是它可以用在程序的任何地方,是全局性的。C 语言可以在有全局声明的情况下执行。

3. 函数

函数是一个程序的命令,每一个 C 程序至少要有有一个 `main()` 函数。正如先前所说的,程序的所有代码都在程序的内部。这样,程序必须以 `main()` 函数开始。

1.5 编译一个 C 程序

编写、编译和运行是一个 C 程序包含的不同步骤,下面将分别进行讨论。在开始讨论这些

步骤之前,必须告诉人们的是 C 程序能在不同的操作系统上运行,比如 DOS 和 UNIX。在不同环境下运行一个程序时包含不同的步骤,下面将要讨论在不同的平台上运行程序时所需要的不同步骤。

1. 在 Turbo C 中的运行步骤

(1) 打开一个 C 程序

当系统准备好后,选择主菜单中的“文件”→“NEW”命令,打开文本编辑窗口。无论什么时候第一次进入 Turbo C 的文本编辑器,系统都会打开一个默认的文件,可在这个默认文件上编辑 C 程序。

(2) 编译一个 C 程序

写完程序代码以后,可先进行编译看是否有错误。编译器执行了编译过程以后,把程序指令转化为机器语言。如果程序中有错误,编译器会立即停止。

(3) 错误检查

只有在代码有错误的时候,这个步骤才会出现。这些错误有可能是语言的语法错误或语义错误,也有可能是别的错误。错误所在的行和错误的类型会在屏幕上显示出来。当程序的错误被纠正以后,程序将继续编译。这个过程会持续到程序不再有错误为止。

(4) 运行程序

若通过了编译,则说明程序没有错误,运行程序以后就会得到一个显示结果。

(5) 显示结果

只有当程序编译和执行以后才会输出该程序的结果。

2. 在 UNIX 系统下的运行步骤

UNIX 操作系统支持不同的用于编译一个 C 程序的命令。因此,编译 Turbo C 的步骤在此不再适用。

(1) 打开一个文件

下面的命令用于打开一个新的文件,它打开一个新的编辑器,程序员在编辑器中编写代码。

```
$ vi first.c (输入回车)
```

(2) 编译一个文件

写完程序代码后,在等待运行时给出编译命令。如果在编译时发现原程序有任何错误,这些错误都会被指出,改正错误以后,再次执行编译命令。该命令如下:

```
$ cc first.c (输入回车)
```

(3) 显示结果

当程序的错误都被改正后,通过了系统的编译时,为了把运行结果显示在屏幕上,还需要执行以下命令:

```
$ a.out (输入回车)
```

以上的 3 个步骤对于打开、编译和执行一个 C 程序来说是强制性的。