

C 程序设计技术 指导与实例

许志平 编译

- ★ 强调工具和技术
- ★ 大量的实例
- ★ 强调数据结构
- ★ 程序优化
- ★ 位操作符



北京希望电脑公司

7387429
239
73.87429
6239
M 便 覽 公 司

许志平 编译

- ★ 强调工具和技术
- ★ 大量的实例
- ★ 强调数据结构
- ★ 程序优化
- ★ 位操作符

一九九一年五月

版权所有
不许翻印
违者必究

★北京市新闻出版局

准印证号：801157

★订购单位：北京8721信箱资料部

★邮 码：100080

★电 话：2562329

★传 真：01—2561057

★乘 车：320、302、332、路车
到海淀黄庄下车

★办公地点：希望公司大楼一楼往里走
101房间

前 言

C程序设计语言是目前最优秀的程序设计语言之一。它是由Bell实验室的D.Ritchie在1972年设计的。它的设计目的是要在PDP—11小型机上开发UNIX操作系统。起初它与UNIX系统联系很紧密,但现在它已在各种计算机的各种操作系统下得到使用。

C语言的应用是非常广泛的。很多现代的工作站系统和应用软件包都是用C语言写的。另外,C语言还广泛地应用于通讯软件、CAD系统、过程控制、仿真系统、甚至与LISP语言共同用于人工智能领域。这种广泛的应用要求程序员充分地掌握这种语言,而且初出茅庐的学生也要具备这种语言的知识。

关于本书

本书是面向计算机、信息和电子工程领域的工作者。书中假定读者已掌握至少一门其它的程序设计语言的经验,并且具有基本的程序概念(如变量和常量的差别)。本书可以作为独立的C语言教程,也可以作为其它运用C语言的课程的补充材料,如操作系统,数据结构,软件工程,系统程序设计,程序语言的比较,计算机图形学或者计算机体系结构。

本书不仅提供了关于C语言程序设计的具体指导,而且对于计算机工作者,软件工程师,和专业程序员也强调了必要的程序设计经验、技术和风格。作者引用的一些有价值的材料甚至对熟练的程序员也是有幫助的。

本书介绍了C程序设计语言的基本工具和技术,同时也着重讨论了该语言中有一定深度的高级课题。作者根据他多年使用C语言的经验给出了上百个实用性强的例子。掌握一种语言的最好方法就是从实践中获得经验,因此希望读者尽可能地把书中的例子应用到自己的系统中去。

显著特征

· 强调工具和技术

本书不仅告诉你怎样编写程序,主要是教你使用C语言这个工具。书中包括了很多广泛使用的算法,如:冒泡排序,快速排序,树结构周游,字符串处理,递归以及模块化程序设计和调试技术。

· 大量的实例

通过例子可以学习得更有效。作者由浅入深地给出了上百个实用例子。语言的每个特性都用一些小例子来说明,以便读者能够很快掌握。除此之外,书中还分析了大量的实例,这些例子来源于各个方面,用来强调C语言的复杂性。

· 配合ANSI标准

书中的材料和例子都符合ANSI关于C语言的标准。此外,作者还把书中那些与Kernighan和Ritchie标准不同的材料和例子一一标出。

· 通俗地介绍了指针

作者彻底而详细地讨论了指针问题，这是C语言中最难以理解的概念之一。为了强化这一概念，在大量的例子和练习中多次使用指针。

- 强调数据结构

这种强调有双重含义。一是要表现出C语言数据类型的复杂性，另外也介绍了大量通用数据结构的使用，包括：栈结构，缓冲区，队列，链接表和树结构。对于已经熟悉这些数据的读者，这也是一个很好的复习。总之，本书的目的是为具体应用选择最佳的数据结构。

- 引用大量的标准库函数

很多被引用的材料对于日常程序设计是有用的，这在你熟悉了C语言之后的很长一段时间都是如此。本书包含了操作符优先级表，ASCII转换表和不同进位制的转换表。在第十一章中还给出了很多通用的系统函数。

- 给出了问题和练习的答案

每一章中都包括大量复习题来强调概念、原理和术语。奇数编号的问题在附录G中给出答案。而且在每一章结束部分可以找到一些程序设计练习，这使学生们能有机会通过完成这些练习来巩固他们所学的知识。

- 程序优化

使用C语言的程序员所关心的一个主要问题就是程序优化。本书用一章的篇幅来讨论这个重要领域的一般性问题。在很多练习中也都涉及到了这个问题。

- 详细讨论了位操作符

用一章的篇幅来讨论位操作符。位操作对于使用C语言的程序员很重要，这是由于在很多应用中都用它来代替汇编语言。在该章中，首先复习了二进制算术运算和布尔运算，然后用大量实例来说明C语言的布尔操作符。

内容介绍

- C语言介绍

第一章讨论了C语言在众多程序设计语言中的地位。并给出了C语言的长处和短处。

- 建立一个程序

第二章考察了如何用C语言编写程序。分析了建立可执行程序所必需的编译和连接过程，同时给出了各种情况下的例子。

- 语言的约定和规则

第三章介绍了C语言程序的基本结构，并提供了书写基本程序的必要信息。介绍并解释了编码的约定和风格。提到了函数的使用，包括Printf函数。

- 数据类型和操作符

第四章讨论了变量、常量和操作符。深入分析了变量的类别和类型。还讨论了说明常量的不同方法以及可以犯的错误。解释了除位操作符以外的其他操作符。

- 位操作符

第五章讨论位操作符。开始先复习了二进制、八进制、十进制和十六进制的记法。然后给出了布尔运算和真值表。对每个布尔运算符都进行了详细讨论并给出了实例。

- 控制流和循环

第六章包括控制程序流程的不同方法，其中有：条件分枝，循环和无条件转移。着重强

调了编码的清晰性和可读性。讨论了goto, break和continue的不同用法。

- 指针和数组

第七章讨论了指针和数组的使用。指针的使用是最难掌握的概念之一, 书中不仅用文字加以解释, 而且给出了很多例子。此外, 还提供了使用指针的一系列规则。本章有大量的例子供读者练习。

- 函数和程序结构

第八章深入地讨论了函数。详细解释了传值调用和传指针调用的差别, 介绍了定义函数的不同方法以及这些函数与程序结构的关系。讨论了把指针作为参数所带来的优缺点。

- 类型定义语句和结构数接类型

第九章包括typedef和struct语句。读者可从中掌握如何用typedef改善程序的可读性和可移植性。在深入讨论结构时, 特别说明它虽然不是kernighan和Ritchie的标准, 但却为大多数编译程序所接受。还讨论了指向结构的指针。对于联合和枚举类型该章也有所涉及。

- C语言中的数据结构

第十章包括C语言中的数据结构及其用法。读者可以获得编写高效程序所用到的不同数据结构的知识。本章开始分析了基本的动态存储分配的实例, 然后给出更一般化的数据结构在C语言中的定义方法, 最后提供了操纵这些结构的函数。

- 标准系统函数

第十一章给出了最常见的库函数, 对每个函数给出了功能描述、调用格式、合法参数表以及实例。

- 调试要点

第十二章讲述如何调试程序。以编程序为职业的人要花费很大一部分时间来调试程序。本章给出了调试程序的一般手段和C语言中的特殊性。

- 效率和速度

第十三章讨论程序优化。C语言常常用于实时操作并受到内存限制。尽管用C语言书写的程序效率很高, 但读者仍要熟悉优化程序的技术。本章详细讨论了在速度和内存要求之间如何进行协调。

- 附录

附录中提供了一些参考资料和奇数练习题的答案。参考资料中包括操作符优先级表、ASCII字符转换表、不同进制的转换表、XENIX系统的实用程序、XENIX上的INCLUDE文件和C语言编译器的任选项。

目 录

前 言	(1)
第一章 C语言介绍	
1.1 C语言概述	(1)
1.2 C语言的优点	(1)
1.3 C语言的缺点	(2)
1.4 选择程序设计语言	(2)
1.5 选择语言要考虑的因素	(2)
1.6 实例分析	(3)
1.7 实例分析	(4)
1.8 总结	(5)
第二章 建立一个程序	
2.1 必要的工作	(9)
2.2 错误信息	(10)
2.3 用多个源文件建立程序	(11)
2.4 使用函数库	(12)
2.5 连接程序任选项	(13)
2.6 总结	(13)
第三章 语言的约定和规则	
3.1 程序的结构和约定	(15)
3.2 简单的程序结构	(15)
3.3 使用函数	(16)
3.4 编码约定	(17)
3.5 函数	(20)
3.6 无所不在的Printf函数	(21)
3.7 预先介绍一下C语言中的特殊字符	(22)
3.8 总结	(22)
第四章 数据类型和操作符	
4.1 变量	(26)
4.2 存储类别	(26)
4.3 基本数据类型	(28)
4.4 变量名	(28)
4.5 赋值语句	(29)
4.6 常量	(29)
4.7 算术操作符	(30)
4.8 增量和减量操作符	(31)

4.9 关系和条件操作符.....	(33)
4.10 数组和字符串.....	(33)
4.11 初始化.....	(34)
4.12 指针.....	(34)
4.13 条件赋值操作符.....	(35)
4.14 一个例子.....	(35)
4.15 总结.....	(37)
第五章 位操作符	
5.1 在位一级的操作.....	(42)
5.2 带符号整数.....	(43)
5.3 位操作符.....	(44)
5.4 一元位操作.....	(45)
5.5 位域.....	(46)
5.6 总结.....	(46)
第六章 控制流和循环	
6.1 C程序的控制流.....	(51)
6.2 if—else语句.....	(51)
6.3 switch和case.....	(54)
6.4 循环.....	(56)
6.5 while循环.....	(57)
6.6 for循环.....	(58)
6.7 do—while循环.....	(62)
6.8 break语句.....	(63)
6.9 continue语句.....	(64)
6.10 标号和goto语句.....	(64)
6.11 总结.....	(65)
第七章 指针和数组	
7.1 指针.....	(72)
7.2 数组.....	(73)
7.3 指针和数组实例.....	(75)
7.4 指针的算术运算.....	(75)
7.5 数组和指针的记法.....	(76)
7.6 多维数组.....	(78)
7.7 指向指针的指针.....	(78)
7.8 用指针指向内存位置.....	(80)
7.9 总结.....	(80)
第八章 函数和程序结构	
8.1 函数.....	(85)
8.2 参数传递.....	(87)

8.3	返回值	(89)
8.4	递归	(89)
8.5	使用递归实现排序	(90)
8.6	外部变量	(92)
8.7	命令行参数	(92)
8.8	用户接口	(93)
8.9	总结	(94)
第九章 类型定义语句和结构数据类型		
9.1	类型定义和结构语句	(98)
9.2	类型定义	(98)
9.3	结构	(98)
9.4	定义结构	(99)
9.5	结构的存取	(101)
9.6	结构的初始化	(101)
9.7	结构数组	(102)
9.8	结构的赋值	(103)
9.9	嵌套结构	(103)
9.10	联合	(105)
9.11	枚举类型	(106)
9.12	总结	(107)
第十章 C语言中的数据结 构		
10.1	C语言中的数据结 构	(116)
10.2	动态存贮分配	(116)
10.3	栈	(117)
10.4	缓冲区和队列	(119)
10.5	树	(123)
10.6	二叉搜索树	(125)
10.7	总结	(128)
第十一章 标准系统函数		
11.1	通用库函数	(133)
11.2	输入和输出函数	(133)
11.3	格式输入和输出	(133)
11.4	格式化的I/O函数	(134)
11.5	串转换函数	(137)
11.6	文件存取函数	(137)
11.7	串处理函数	(141)
11.8	字符检查函数	(143)
11.9	存贮分配函数	(144)
11.10	其它的函数	(145)

11.11 建立你自己的函数库.....	(145)
11.12 总结.....	(146)
第十二章 调试要点	
12.1 调试.....	(152)
12.2 常见的错误.....	(153)
12.3 结论.....	(154)
12.4 总结.....	(154)
第十三章 效率和速度	
13.1 程序优化.....	(161)
13.2 优化的编译程序.....	(161)
13.3 存储空间的考虑.....	(161)
13.4 运行速度的考虑.....	(162)
13.5 总结.....	(163)
附录A 操作符的优点顺序.....	(168)
附录B ASCII转换图.....	(169)
附录C 基数转换图.....	(172)
附录D 开发支持程序.....	(173)
附录E XENIX的INCLUDE文件.....	(175)
附录F 编译信息.....	(175)
附录G 奇数问题的答案.....	(176)

第一章 C语言介绍

学习目的:

- 介绍C语言
- 描述C语言与其它语言的关系
- 讨论C语言的优缺点
- 对特定问题选择什么样的语言

1.1 C语言概述

为了对一个特定问题选择语言,你必须了解所有可以用的语言的优缺点。由于本书是讨论C语言的,所以只对它给出优缺点的总结。

C语言不是大型和复杂的语言,但它很容易扩展。它具有低级语言的优点。它的数据结构很简单,并与机器硬件相关。这种相关性使C语言可以用于汇编语言的应用领域。C语言还可以构造复杂的数据结构,并把它作为内部已定义好的结构使用。虽然不存在移动或比较复合数据结构的操作,但是很多C语言编译器包括了实施这种操作的函数。C语言还缺少很多基本的输入输出功能,这要靠库函数来提供。

C语言支持控制程序流程的完整的语句集合,包括if-else, while循环, for循环, do循环和switch-case语句。这些语句可以使程序具有优秀的结构并且完全消除了对goto语句的需要。

C语言的这种小型的和简单的结构带来了很多优点。由于它的小型化,编译器可以很容易被移植到新的硬件系统上。由于把输入输出函数从语言中搬到了函数库中,设计者可以使被编译的程序独立于操作系统。为了在一个系统上开发出适应于具有同样硬件结构的另外一个系统上能使用的程序,只需要替换函数库即可。

小型化也带来了高效和快速的执行代码,同时使编译速度非常快。C语言的这些特性成功地使它成为微型机和小型机上的最常见的通用语言之一。

1.2 C语言的优点

下面给出一些C语言在一般应用中的优点:

1. 速度。很多编译程序产生的C语言代码效率都很高。由于语言的小型化,运行系统的小型化和C语言接近于硬件这一事实使很多C语言程序的运行速度接近于用汇编语言书写的相同程序。

2. 存储要求。由于同样的原因, C语言程序非常有效地利用了存储器。由于缺少内部设定的函数使程序避免了引用大量不必要的函数空间。

3. 可移植性。可移植性是度量把一个计算机或操作系统上的程序转换到另外一个计算机或操作系统上的方便程度。用C语言书写的程序很容易在计算机之间移植,尤其是在微型机和小型机上。

4. 结构程序设计。C语言包括了所有结构程序设计中必需的成份。

5. 快速编译。在任何系统上C语言的编译都是很快的。

6. 支持模块化程序设计。C语言支持单独编译和连接。这允许程序员只重新编译程序中被改动过的部分。这种特性对于开发大型程序或者慢速系统上的中型程序是非常重要的。没有模块化程序设计的支持,编译完整的程序所花费的时间就非常可观。

7. 与汇编语言的清晰接口。大部分C语言编译器都给出了调用汇编语言过程的标准方法。通过分别编译和连接,把高级语言和汇编语言结合在一起,大大地扩充了C语言的应用范围。同样,在很多系统中,C语言过程也可以被汇编语言所调用。

8. 特殊函数库。大部分流行的C语言编译器都带有完整的函数库。函数库中包括图形功能,文件处理,数据库支持,屏幕窗口控制,数据访问,通讯控制和通用函数。这些库函数大大地节省了开发时间。

9. 低级存取。C语言允许用户访问机器硬件,如:寄存器变量和位操作。

1.3 C语言的缺点

下面列出C语言的一些缺陷:

1. 不是强类型语言。类型是标志语言如何使用变量的(如整型和浮点型是两种不同的数字类型)。在某些语言中,不允许直接把不同类型的数据相互赋值。这保护了数据的转换机制。在C语言中,可以把整型变量赋给字符型变量。这个特性是语言的优点还是缺点还在争论中,但是编译器允许这种转换是导致程序错误的原因之一,因此可以认为它是缺点。

2. 缺少运行期间检查。这种缺陷可能导致系统出现很多奇怪的和偶发的错误。如果系统可以在运行期间检查诸如数组越界之类的错误,则可能丧失速度和效率。

3. 缺少字符串和记录的处理。这个缺点是象C语言这种小型语言不可避免的最大弱点。如果需要读入,移动,比较和写出复合的记录,就不应该使用C语言,因为它本身就不是为这个目的所设计的。

4. 在大型机应用中使用有限。据说,很多大型机程序员都不懂C语言。也就是说,C语言并不是无处不在。如果在一个团体中只有一个人懂C语言,那么维护这个程序就非常困难。

C语言的优点和缺点导致它很适合于某些应用,而且不适用于另一些应用。它的最值得夸耀的特点是高效率 and 可移植性。

1.4 选择程序设计语言

没有十全十美的语言。不同的问题需要不同的解法。系统分析员的任务就是为一个工程选择最合适的语言。这是完成这个工程所要作出的第一个决策,一旦开始编程序,就无可挽回了。语言的选择可以影响到工程的成败。在本章中我们要给出影响选择语言决策的各种因素。

1.5 选择语言要考虑的因素

在开始一个工程时有很多因素是与选择语言相关的。最好的方法是询问一些有关程序完成后的需求,程序所在的系统的能力。下面给出一些需要提出的问题。

1. 你开发软件所用的系统都支持哪些语言? 支持到什么程度? 原来系统上有用这些语言开发的复杂程度相当的程序吗?

2. 对程序有什么样的移植要求? 将来系统升级时必须对它修改吗? 该程序是否只在一种系统上运行? 有没有用在其它系统上的版本?

3. 开发人员熟悉哪种语言? 有必要为这个项目学习一种新语言吗?

4. 程序的哪一部分与具体语言的特性有关, 而其它语言不支持? 例如, 用 FORT-RAN 语言写的程序适于作复杂计算, 而 COBOL 语言则不行。

5. 需要与汇编语言接口吗? 有些高级语言很难做到这一点。如果必须要有汇编子程序, 就应该选择一种支持它的语言。

6. 该项目对人力或完成时间有要求吗? 最高效的程序当然是用汇编语言写出的, 但这会花费大量的时间和精力。一种作法是, 关键程序用汇编语言写, 其它部分用高级语言写。

7. 你单位中广泛使用这种语言吗? 如果是这样, 用这种语言写的程序将来就很容易维护。

8. 必须要用的函数和子程序可以在函数库中找到吗? 实践证明, 使用函数库要比自己编写相应的程序省时省力。

9. 程序需要运行期间的支持吗? 你需要运行系统识别出哪些操作错误?

10. 有什么样的调试手段? 调试高级语言输出的机器代码与调试高级语言本身有天壤之别。

11. 系统级软件中有数据库管理系统或事务处理系统吗? 如果有的话, 高级语言与它们的接口已经提供了吗?

以上这些因素的重要性对于不同的项目变化很大。经常出现一种因素占压倒一切的地位, 以致于只要考虑这一个因素就行。同样, 有些因素可以完全忽略。

下面让我们检查一下一些项目的说明, 以体会如何回答前面给出的问题。

1.6 实例分析——图形板绘图程序

一个图形板制造商准备宣布一种在 IBM PS/2 上使用的新产品。该产品是一个 20 平方英寸的图形板, 带有一个金属探笔。使用者可以象在纸上一样在图形板上绘图, 或者把图纸放在图形板上进行描述。每个图形板有个控制卡可以插在 IBM 扩展槽上。

厂商感到如果为每个图形板配备一简单的图形程序将会大大提高销售量。该程序必须支持各种图形, 包括直线, 多边形, 圆或彩色填充。程序应该看上去很好用, 并包含现代风格(如下拉菜单)。它还要支持存储和调用已给出的图形。程序要在任何使用 MS-DOS 2.0 以上版本的系统上运行。

本程序要在三个月内完成, 以便配合图形板的销售。由于没有其它的人, 你被指定单独完成这个程序。从制造商那里得不到更进一步的信息, 也没有更多的要求。你可以自由选择任何系统和语言来完成工作。

下面让我们尽可能回答前面提出的问题, 并对开发软件使用的语言作出合理选择。

1. 系统支持什么语言? 在 IBM PC 上可以使用汇编语言、BASIC、C 和 PASCAL。这些语言是很容易得到的。

2. 程序的移植性如何? 程序要求在 IBM PC 上的 MS-DOS 下运行, 要求支持更高的

MS-DOS版本。

3. 你熟悉哪种语言？这只能由你自己回答。

4. 该项目倾向于使用哪种语言？程序应该有与图形板控制卡的接口和高速绘图功能。所以最好用汇编语言。程序还要处理文件和管理高级用户接口，这又需要用高级语言。

5. 需要有与汇编语言接口吗？是。

6. 对人力和时间有限制吗？是的，你需要在三个月内独立完成。这就暗示着要使用高级语言做大部分工作。

7. 这是单位中常用的语言吗？不，单位中没有人进行软件开发。

8. 可用的库函数有助开发吗？C语言函数库可以提供很多帮助。它包括图形方式的设置和基本绘图功能。这些函数可以完成程序要求的大部分功能。

9. 需要运行期间支持吗？调试好的程序很少需要支持。一旦程序发行，在运行期间出错是否给出信息关系并不大。

10. 有什么样的调试手段？IBM PC上可以使用符号调试程序。它允许你在源程序一级调试，而不是按机器代码调试。

11. 有哪些系统程序要与它联用？程序需要DOS的支持，而高级语言使用DOS是没有问题的。

根据这些信息，我们要决定用哪种语言。BASIC对于绘图程序来说太慢了，何况与汇编语言的接口不太清晰。短期内开发成功的规定和与DOS的程序接口又要求不能完全用汇编语言完成。在剩下的几种语言中，C语言是当然的候选者。除了C函数库提供了支持外，C也很容易与汇编语言相连，而C语言的数据结构又与机器硬件连结的很紧密。

很明显，本例的最佳选择是C语言。

1.7 实例分析——零件清单报告

某汽车零件制造厂需要一个当前拖期的用户发货清单。该清单应该在每周一的早晨送到工厂经理和付经理面前。该程序应该在每周末由操作员以批处理方式生成零件清单。

程序运行在IBM 3033机器上。可用的语言包括BASIC、C、COBOL、FORTRAN和PASCAL。当前订单存放在TOTAL数组库文件中，大约有70000个记录。厂家目前已有自己编写的COBOL程序超过60万行。

1. 系统支持什么语言？包括BASIC、C、COBOL、FORTRAN和PASCAL。

2. 程序的移植性如何？不需要。

3. 你熟悉哪种语言？问你自己。

4. 该项目倾向于使用哪种语言？程序可以用COBOL完成，但BASIC、C和PASCAL也可以。COBOL的优点是可以处理文字和记录，此外，COBOL使用者很多，很多程序员都可以使用和维护。

5. 需要有与汇编语言的接口吗？不。

6. 对人力和时间有限制吗？没有提到。可以想象，若不能有所进展，会补充上这个要求的。

7. 什么是单位中常用的语言？COBOL。

8. 可用的库函数有助于开发吗? 该任务似乎无需库函数的帮助, 但厂内过去开发的COBOL子程序会对工作有所帮助。

9. 需要运行期间支持吗? 调试好的程序很少需要支持。

10. 有什么样的调试手段? 无。调试程序时只需要观察结果, 然后修改输入数据。

11. 有哪些系统程序要与它联用? 程序需要与TOTAL数据库管理系统连接。厂家内有COBOL和FORTRAN语言的接口程序。

这个问题不用详细研究。题目要求用批处理方式执行, 所以可以不考虑速度和内容要求。除此之外, 访问数据库这一要求就导致必须选择COBOL。

从前面可以看到, 每个问题都有不同的要求和限制。在某些项目中, 可以尝试使用多种程序设计语言; 而另外一些则无法选择。有时, 对做出决策有帮助的信息很缺乏。也有时, 没有做出决策的必备知识。有时甚至要自己花费一些时间来寻找与决策相关的信息。

你可能已经注意到, 所给出的信息中有些与技术无关。尽管看上去很奇怪, 但很多技术决策都是由技术和非技术两方面因素决定的。

必须注意, 在你决定使用一种语言之前, 需要做出更详细的技术规范说明。这种说明对你下一步编写程序和确定问题是很有帮助的。

绘图程序要求汇编语言是因为要适应用户移动鼠标器时更新画面的需要, 还有与硬件接口的要求。这些功能都可以用C语言完成, 但用汇编语言做起来更简单。对于零件清单的问题, C语言不是好的语言, 因为存在着最佳候选COBOL。

1.8 总结

C语言是小型、简单和通用的程序设计语言。它的优点是:

1. 运行速度快。
2. 占用内存少。
3. 可移植性强。
4. 支持结构化程序设计。
5. 编译速度快。
6. 支持模块化程序设计。
7. 与汇编语言接口简单。
8. 库函数丰富。
9. 与硬件相关性好。

它们的缺点是:

1. 缺少强类型。
2. 缺少运行期间检查。
3. 缺少字符串和记录处理。
4. 在大型机上用的很少。

在选择一种程序设计语言时, 要把候选语言的优点和缺点与问题的要求相互比较, 然后做出决定。

复习题

1. 为什么用C语言编写的程序能有效利用内存?

2. 什么是可移植性?
3. 为什么编译速度很重要?
4. 连接程序的功能是什么?
5. 什么是函数库?
6. 使用函数库的优点有哪些?
7. 函数库的来源有哪些?
8. 什么是类型?
9. 一种语言不能把浮点数赋给整型变量, 怎么称呼它?
10. 强类型语言有什么优点?
11. C语言是强类型语言吗?
12. C语言支持类型吗?
13. 用高级语言写程序总比用汇编语言写快吗?
14. 什么是源程序级调试程序?
15. MS-DOS支持BASIC, 它与C语言和PASCAL语言的主要区别是什么?
16. C语言内部有读写文件和打印的功能吗?
17. C语言支持模块化程序设计吗?
18. 模块化程序设计的主要优点是什么?
19. 如果模块是分别编译的, 怎样把它们合并在一起?
20. C语言很容易移植, 还是不容易移植?

程序设计练习

1. 设计规范和语言的选择

a. 对下面的各项的描述, 仔细分析所给信息, 并列出与选择语言有关的因素。如果信息不足, 就指出所缺乏的内容和可能的来源。

b. 把所给的说明扩展成一般的功能描述, 并标出问题的领域。

c. 列出使用不同语言所带来的优点和缺点。

实例1: 联机安全管理程序

你的公司的中心大型机上有三百多个拨号终端。所有这些线路由一个前端通讯控制机管理。公司要在控制机上用一个程序记录所有的非法拨号请求。程序可以用一条通讯线路与计算机操作员通讯并打印出报告。程序只要收集有关数据即可, 而不必采取任何行动。

通讯控制机由一台小型机担任, 可以使用汇编语言、C、FORTRAN和PASCAL。

实例2: 熔炉控制

某公司要用IBM PC机对热处理熔炉进行过程控制。程序执行时要驻留在内存中, 有些特殊程序如设备测试或校对, 可以单独执行。

PC机有512K内存、10兆硬盘、CGA显示器、实时时钟、12个模拟输入口用于温度和压力的测试、一个模拟输出口用来控制温度、64个110伏数字输入输出控制阀和安全门、还有一个网络通讯线连到大型机上。

实例3: 质量控制(QC)

某重型设备制造厂的质量管理小组收集了5年来有关驱动轴误差的数据。由164个可测点

对7万个轴测试了与标准值的误差。QC小组得到了过去5年的驱动轴的使用记录。他们雇用
一个统计人员进行这项工作。

数据用顺序文件存贮在Cyber 180机上。公司还有一台Cyber 205备用。Cyber 180支持
汇编语言、BASIC、C、COBOL、FORTRAN和PASCAL。Cyber 205支持汇编和
FORTRAN。统计结果要求立刻做出。你必须注意，QC小组在本财政年度内超出了计算机
时的预算，而现在距年底还有5个月。

实例4：订单管理

某公司要开展电话订购任务。需要一个程序帮助操作员接到用户电话的同时输入订单。
订单包括用户名、地址、电话号码、信用卡信息和1到10个条目。程序接受条目号码查找数
据库，并显示该商品条目的描述和价格，要求在输入商品号码5秒钟内给出回答。操作员输
入数量之后，如果有多种尺寸和颜色，程序要询问进一步的信息。

所有信息齐备后，要把它存入当天订单文件。在订单中还要加入操作员编号，制做订单
的日期和时间。

程序运行在micro VAX II上，支持12个操作员。可用语言有：汇编BASIC、C、
COBOL、FORTRAN和PASCAL。该公司使用的计帐系统和清单管理程序用COBOL写
成。清单数据库是用该公司的帐目管理软件包维护。公司准备将来把系统扩充到支持36个操
作员，那时要用更大的VAX系统或多个VAX机。公司没有打算使用其它机器。

公司经理和市场部经理都指出，该程序是公司电话订购业务成败的关键。

实例5：销售支持系统

一个用户需要一个简单的销货支持数据库。程序接收有希望的顾客姓名，地址和电话。
其它信息包括：

1. 回话日期。
2. 最后一次电话的日期。
3. 希望程度(0—10)
4. 推销员
5. 注释

程序要完成下面功能：

1. 输入新记录。
2. 删除老记录。
3. 允许加入和修改通话日期。
4. 扫描数据库，查找应该回话的记录。

系统要求在IBM PC/AT兼容机上运行，有10兆主频，640K内存，40兆硬盘和单色显
示器。

有5个推销员共同使用这台机器，但分别用不同的数据库。由于每个数据库属于特定推
销员，所以IBM PC/AT足够装下这个系统。

公司在IBM PC机上有一些开发BASIC，C和PASCAL程序的工具。最近的开发一直使
用C和PASCAL。

2. 自顶向下设计

对下面的程序描述，编写一个完整的程序说明书。假定程序是在IBM PC或其兼容机上