

Microsoft C 5.0

优化编译

Quick C 使用说明

(第四册)

中国科学院 科海培训中心
软件所

Microsoft C 5.0 优化编译

Quick C 使用说明

(第四册)

译 校 者 序 言

翻 译：宗丽萃 吴 倩 邦继明 鲁 倩 王 越 徐砥祥 陈 林 彭怀宇
王淑平 何 骏 郭瑞阳 萧燕林

Microsoft C 5.0优化编译语言系统是美国87年国家标准，具有效率高、功能强的特点。为了迎合广大热心者的要求，配合国内用户高水平开发工作，由中国科学院386微机研究，开发组直接组织编译了此书共九册。虽然译校者大多是软件专业的研究生及工作多年的高资历工作人员，但由于印排仓促，审校时间紧迫，难免会有这样或那样的错误。望读者能给以谅解与指正。

参与本书印排工作的除中国科学院软件所、计算所、自动化所之外，还有祥云电脑公司、海声软件开发公司，故上述单位共有此中文资料版权。任何其它单位不得随意翻印此书。

目 录

导言	(1)
系统所需配置	(3)
使用本手册	(3)
本手册是如何组织的	(3)
阅读本手册的哪些内容	(3)
和快速 C 一起提供的其它手册	(4)
记号约定	(4)
学习资料	(5)
获取 Microsoft 的帮助	(6)
第一部分 启动	(7)
1 装配并启动快速 C	(9)
1.1 制作研盘备份	(9)
1.2 核对研盘的内容	(9)
1.3 装配快速 C	(9)
1.3.1 在硬盘系统中装配	(10)
1.3.1.1 在硬盘系统中 SETUP 做些什么	(10)
1.3.1.2 在硬盘系统中运行 SETUP	(10)
1.3.1.3 在硬盘系统中建库	(12)
1.3.1.4 删除库的某些部分	(12)
1.3.1.5 完善装配	(12)
1.3.1.6 在硬盘系统中建立 DOS 环境	(12)
1.3.2 在软盘系统中装配	(13)
1.3.2.1 在软盘系统中 SETUP 做些什么	(13)
1.3.2.2 在软盘系统中运行 SETUP	(13)
1.3.2.3 在软盘系统中建库	(14)
1.3.2.4 在软盘系统中建立 DOS 环境	(15)
1.4 使用非组合库	(15)
1.5 使用 Microsoft C 图形库	(16)
1.6 如果你有鼠标器	(16)
1.7 运行快速 C 编译器	(17)
1.7.1 启动快速 C 编译器	(17)
1.7.2 QC 命令	(17)

1.7.3 退出快速 C	(18)
1.7.4 样例编译	(18)
1.8 保存任选设置:	(19)
2 快速 C 程序设计环境简介	(20)
2.1 快速 C 屏幕	(20)
2.2 使用快速 C 菜单	(21)
2.2.1 快速 C 菜单的内容	(21)
2.2.2 在菜单中选择命令	(22)
2.2.3 命令的简便键	(23)
2.3 使用对话框	(24)
2.4 选择正文	(26)
2.5 步进滚动	(26)
2.6 获取帮助: Help 菜单	(27)
3. C 快速启动	(29)
3.1 C 程序的结构	(29)
3.2 说明	(31)
3.3 预处理程序伪指令和内含文件	(32)
3.4 变量说明	(33)
3.5 语句、表达式和运算符	(35)
3.6 函数说明和定义	(36)
3.7 循环语句	(37)
3.8 条件语句和分支语句	(39)
3.9 数组和字符串	(41)
3.10 指针简介	(42)
3.11 函数指针	(44)
3.12 结构	(45)
3.13 使用 C 输入/输出函数	(47)
3.14 使用文件输入/输出函数	(48)
3.15 存取 C 命令行自变量	(50)
4 图形快速启动	(53)
4.1 绘图程序的结构	(53)
4.2 设置视频方式	(55)
4.3 获取视频配量	(57)
4.4 使用彩色正文方式	(59)
4.5 使用 CGA 彩色图形方式	(60)
4.6 使用 EGA 彩色图形方式	(62)

4.7	使用 VGA 彩色图形方式	(64)
4.8	坐标系统理解	(65)
4.9	图点绘制	(66)
4.10	画直线	(68)
4.11	描述图形对象	(70)
4.12	画基本形状	(71)
4.13	用模式填充图形	(73)
4.14	画图 and 存图	(75)
4.15	动画的使用	(76)
第二部分 快速 C 程序设计环境		(79)
5	显示控制: View 菜单	(81)
5.1	显示程序模块表: Source 命令	(81)
5.2	显示内含文件 Include 命令	(82)
5.3	定制显示格式 Options 命令	(82)
5.4	显示程序输出: Output Screen 命令	(84)
5.5	打开或关闭错误窗口 Errors 命令	(84)
6	建立并保存程序	(85)
6.1	程序和模块: 程序模块表	(85)
6.1.1	单模块程序	(85)
6.1.2	多模块程序	(87)
6.1.3	用程序模块表工作	(87)
6.1.4	快速 C 中程序模块表的使用	(87)
6.1.5	关于内存程序和程序模块表的常见问题	(88)
6.2	处理源文件: File 菜单	(89)
6.2.1	建立新程序: New 命令	(90)
6.2.2	装入文件编辑: Open... 命令	(91)
6.2.2.1	用 Open 装入文件	(91)
6.2.2.2	用 Open 列表文件	(92)
6.2.3	打开最新编辑的文件: Open Last File 命令	(92)
6.2.4	合并文件: Merge... 命令	(92)
6.2.5	保存文件: Save 和 Save As... 命令	(93)
6.2.6	建立并装入程序模块表: Set Program List... 命令	(94)
6.2.7	删除当前程序模块表: Clear Program List 命令	(94)
6.2.8	编辑程序模块表: Edit Program List... 命令	(94)
6.2.9	打印文件: Print... 命令	(96)

6.2.10	暂时返回 DOS, DOS Shell 命令	(97)
6.2.11	退出快速 C, Exit 命令	(97)
编辑源文件		(98)
7.1	使用键盘和鼠标器编辑	(98)
7.1.1	使用快速 C 编辑键	(98)
7.1.2	使用插入和覆盖键入方式	(100)
7.1.3	在正文中使用位置标记	(101)
7.1.4	配对花括号	(101)
7.1.5	使用鼠标器编辑	(101)
7.2	编辑命令: Edit 菜单	(102)
7.2.1	复原编辑: Undo 命令	(102)
7.2.2	删除和插入正文: Cut, Copy, Paste, 和 Clear 命令	(103)
7.2.3	设置只读方式: Read Only 命令	(103)
7.3	搜索和替换正文: Search 菜单	(104)
7.3.1	查找正文: Find .., Selected Text, 和 Repeat Last Find 命令	(105)
7.3.1.1	Find... 命令	(105)
7.3.1.2	查找所选正文: Selected Text 命令	(106)
7.3.1.3	Repeat Last Find 命令	(106)
7.3.1.4	正规表达式中的特殊字符	(106)
7.3.2	替换正文: Change... 命令	(107)
7.3.3	查找函数: Function 命令	(108)
7.3.4	查找程序错误: Next Error 和 Previous Error 命令	(109)
8. 程序的编译、运行和调试		(110)
8.1	编译并运行程序: Run 菜单	(110)
8.1.1	运行程序: Start 命令	(111)
8.1.2	使程序作好重新运行的准备: Restart 命令	(111)
8.1.3	继续程序执行: Continue 命令	(111)
8.1.4	控制编译时任选项: Compile... 命令	(111)
8.1.4.1	隐匿编译程序警告: Warning Level Options	(112)
8.1.4.2	选择输出文件格式: Output Options	(112)
8.1.4.3	调试准备: Debug Option	(113)
8.1.4.4	使用“灵巧指针” Pointer Check Option	(113)
8.1.4.5	检查栈溢出: Stack Check Option	(114)
8.1.4.6	使用 C 的 Microsoft 扩展: Language Extensions Option	(114)
8.1.4.7	建立快速程序 Optimizations Option	(115)
8.1.4.8	查找内含文件: Include Text Box	(115)
8.1.4.9	定义常数和宏: Define Text Box	(115)

8.1.4.10	编译命令按钮	(116)
8.1.5	控制运行任选项: Set Runtime Options 命令	(117)
8.2	程序调试	(118)
8.2.1	一般调试过程	(118)
8.2.1.1	增加观察表达式	(118)
8.2.1.2	设置断点	(118)
8.2.2	调试键盘命令	(119)
8.2.3	调试命令: Debug 菜单	(119)
8.2.3.1	增加观察表达式: Add Watch 命令	(119)
8.2.3.2	删除观察变量: Delete Last Watch/ All Watch 命令	(122)
8.2.3.3	控制踪迹 Trace On 命令	(122)
8.2.3.4	控制屏幕交换: Screen Swapping On 命令	(122)
8.2.3.5	控制断点: Toggle/clear All Breakpoints 命令	(122)
8.2.4	在函数间追踪 Calls 菜单	(122)

第三部分 Quick C 工具集 (125)

9	程序的编译和连接	(127)
9.1	编译和连接过程	(127)
9.1.1	使用单个QCL命令行	(128)
9.1.2	使用QCL和LINK命令	(128)
9.2	单步编译和连接: QCL 命令	(129)
9.2.1	指定文件名	(129)
9.2.1.1	大写字母和小写字母	(129)
9.2.1.2	文件名扩展	(130)
9.2.1.3	路径名	(130)
9.3	用QCL任选项控制编译	(130)
9.3.1	环境任选项	(132)
9.3.2	连接控制任选项	(133)
9.3.3	列出编译程序任选项(/HELP)	(133)
9.3.4	命名目标文件(/Fo)	(133)
9.3.5	内存模型(/A)和浮点(/FP)任选项	(134)
9.3.6	使用8086或80286协同处理器(/GO,/G2)	(135)
9.3.7	控制预处理器	(135)
9.3.7.1	删除已预先定义的标识符的定义(/U,/u)	(136)
9.3.7.2	产生预处理表(/P,/E,/EP)	(136)
9.3.7.3	保留注解(/C)	(137)
9.3.7.4	查找内含文件(/X)	(137)

(811)	9.3.8	调试准备(/zi,/zd)	(138)
(515)	9.3.9	压缩结构成员(/zp)	(138)
(811)	9.3.10	隐匿缺省库选择(/zl)	(139)
(811)	9.3.11	控制调用约定(/Gc)	(139)
(811)	9.3.12	设置数据阈值	(140)
(811)	9.3.13	优化循环(101)	(140)
(81)	9.4	独立的编译和连接: QCL 和 LINK	(140)
(911)	9.4.1	用 QCL 命令连接	(140)
(911)	9.4.2	用 LINK 命令连接	(141)
(851)	9.4.2.1	LINK 的缺省信息	(142)
(851)	9.4.2.2	指定 LINK 文件	(143)
(851)	9.4.2.3	指定 INK 库	(143)
(851)	9.4.2.4	LINK 内存需求	(144)
(85)	9.5	使用连接程序任选项	(145)
	9.5.1	显示任选项表(/HE)	(145)
	9.5.2	连接过程中的暂停(/PAV)	(145)
(751)	9.5.3	显示连接程序进程信息(/I)	(146)
	9.5.4	抑制连接程序提示(/B)	(146)
(851)	9.5.5	建立快速库(/Q)	(146)
(751)	9.5.6	压缩可执行文件(/E)	(147)
(85)	9.5.7	忽略缺省库(/NOD)	(147)
(851)	9.5.8	设置段最大数(/SE)	(147)
(85)	9.5.9	设置最大分配空间(/CP)	(147)
(851)	9.5.10	建立映象(Map)文件(/M,/LI)	(148)
(851)	9.5.11	控制栈的大小(/ST)	(149)
(851)	9.5.12	翻译段外调用(/F,/NOF)	(150)
(851)	9.6.13	压缩连续段(/PAC,/NOP)	(150)
(851)	9.5.14	其它 LINK 任选项	(150)
(85)	9.6	控制栈及堆分配	(151)
(85)	9.7	使用覆盖技术	(152)
(851)	9.7.1	覆盖的若干限制	(152)
(851)	9.7.2	覆盖管理程序提示	(153)
(851)	10	建立Quick库和独立库	(154)
(85)	10.1	Quick库	(154)
(851)	10.1.1	建立Quick库	(154)
(851)	10.1.2	装入Quick库	(155)
(851)	10.1.3	Quick库中的标准库例行程序	(155)
(85)	10.2	独立库管理: LIB 实用程序	(156)

10.2.1	运行 LIB	(157)
10.2.2	LIB 的缺省值	(157)
10.2.3	命令符号	(158)
10.2.4	交叉引用列表文件	(159)
10.2.5	设置库页面大小	(160)
	用 MAKE 进行自动化程序开发	(161)
11.1	MAKE 的核心: 描述文件	(161)
11.1.1	建立 MAKE 描述文件	(161)
11.1.2	描述块	(162)
11.2	运行 MAKE	(164)
11.3	在 MAKE 中使用宏定义	(164)
11.3.1	定义和说明宏	(165)
11.3.2	在宏定义内使用宏	(166)
11.3.3	使用特殊的宏	(166)
11.4	定义推理规则	(167)
11.5	使用 .MAK 文件	(168)
11.5.1	MAKE 使用 .MAK 文件	(168)
11.5.2	内含文件从属关系	(169)
11.5.3	指定连接程序任选项	(169)

附 录

A	ASCII 字符码	(170)
B	用 Quick C 内存模型工作	(170)
B.1	段内寻址和段外寻址	(170)
B.2	使用标准内存模型	(171)
B.2.1	建立小模型程序	(171)
B.2.2	建立中模型程序	(172)
B.2.3	建立紧致大模型程序	(173)
B.2.4	建立大模型程序	(174)
B.3	使用段内关键字和段外关键字	(175)
B.3.1	段内和段外的军支持	(175)
B.3.2	用 near 和 far 说明数据	(176)
B.3.3	用段内和段外关键字说明函数	(177)
B.3.4	指针转换	(178)
B.4	设置数据阈值	(180)
B.5	命名正文段	(180)

C	C 与汇编语言相接口	(180)
C.1	编写汇编过程	(181)
C.1.1	建立过程	(182)
C.1.2	进入过程	(182)
C.1.3	分配局部数据(任选)	(182)
C.1.4	保留寄存器	(183)
C.1.5	访问参数	(183)
C.1.6	返回值(任选)	(184)
C.1.7	退出过程	(185)
C.2	从C调用汇编语言例程序	(185)
C.3	从汇编语言调用C	(187)
C.4	Microsoft 段模型	(188)
D	错误信息参考	(190)
D.1	编译程序错误信息	(190)
D.1.1	严重错误信息	(191)
D.1.2	编译错误信息	(195)
D.1.3	警告信息	(206)
D.1.4	编译程序限制	(211)
D.2	命令行错误信息	(212)
D.2.1	命令行严重错误信息	(212)
D.2.2	命令行错误信息	(212)
D.2.3	命令行警告信息	(214)
D.3	运行时错误信息	(214)
D.3.1	浮点异常	(215)
D.3.2	运行时库错误信息	(216)
D.3.3	运行时限制	(217)
D.4	LINK 错误信息	(218)
D.5	LIB 错误信息	(224)
D.6	MAKE 错误信息	(227)

导 言

欢迎使用 Microsoft 快速 C 编译器! 你已经购买了能够用 C 语言进行工作的最有效的软件包之一, 它是程序设计初学者以及有经验的软件开发者理想的一组工具。

Microsoft 快速 C 编译器是集效率、可移植性、灵活性于一体并提供完善的开发环境的 C 程序设计语言的完全实现。它包括如下一些特性:

在一个集成化程序设计环境中获得所有你需要的工具

Microsoft 快速 C 编译器在单个软件包中提供你所需要的程序开发工具: 集成程序编辑器编译器和调试程序 (第 8 章) 用 Microsoft 快速 C 编译器, 你可以不用退出程序设计环境而对程序进行编译。运行、调试、编辑和重编译快速 C 编译器可以建立在内存中执行的程序, 也可以建立在库加上可执行文件中使用的独立的目标文件, 以便今后在 DOS 下运行。

因为大多数标准 C 库函数建立在程序设计环境中, 因此通常没有必要和外部库连接。任何在环境中没有建立的标准 C 库例行程序可以装入称为快速库的用户可配置库中。

使用 Wordstar 兼容编辑器快速学习

学习快速 C 编辑器是一件简单的事情, 如果你熟悉 MicroPro WordStar 程序, 那么, 你已经知道了大部分快速 C 编辑命令, 因为它们和相应的 WordStar 命令完全一样。(7.1.1 描述快速 C 编辑键)。

使用集成化调试程序快速找错

快速 C 包含许多专用调试程序的排错能力, 它们使得以程序逻辑跟踪错误变得很容易。使用快速 C 调试程序, 你可以:

- 一行一行地走过程序执行
- 设置多个断点
- 在程序执行过程中, 在独立的窗口显示变量的值
- 在程序显示和程序输出显示之间转换
- 在源程序中查找特定的函数

§ 8.2—§ 8.24 给出使用这些调试实用程序的方法, 即使这些特性还不够, 你还可以使用和 Microsoft C 优化编译器以及其它 Microsoft 语言产品一起提供的 Microsoft Code View 窗口调试程序。

在你的手头很方便地获取帮助

你有关于 C 语句语法方面的问题吗? 你有关于库例行程序参数类型方面的问题吗? 你有关于快速 C 编辑器键方面的问题吗? 用快速 C 的上下文相关的连联帮助实用程序可以很快

地得到答案，浏览一下题目的菜单，或者简单地照亮(highligf)某一函数名并按键，你甚至不需要离开编辑程序，信息便出现在屏幕顶端的一个单独的窗口中。

图形库给程序添加直观的紧缩

和快速 C 库一起提供的图形库使你能够写出复杂的图形应用程序，该库支持所有的 IBM 及其兼容机显示适配器，包括 IBM 视频图形陈列 (VGA)，单色显示适配器 (MDA)，彩色/图形适配器 (CGA)，以及增强型图形适配器 (EGA)，有关图形库的信息参见第四章。

快速编写程序，然后用 Microsoft C 优化编译器进行优化

由于 Microsoft 快速 C 编译器和 Microsoft C 优化编译器在源文件和目标文件一级是完全兼容的，当你需要增加 Microsoft C 优化编译器提供的灵活性和效率时，保证你可以顺利通过，如果你是在用 Microsoft C 优化编译器开发程序，你可以使用 Microsoft 快速 C 编译器加快程序开发的早期进程，然后再用 Microsoft C 优化编译器对程序性能进行精细的调整。

用和 OS/2 兼容的用户接口快速启动

Microsoft 快速 C 编译器的用户接口很容易学会。你只需用键盘，任选鼠标器或两者简单地从菜单中选择命令，或者不用打开菜单就按一些特殊键执行命令。看来快速 C 用户接口和工作起来都很象新 Microsoft 显示管理程序与 OS/2 操作系统的接口。

自动维护大程序

具有大量模块的大型程序是用 Microsoft 快速 C 编译器自动进行维护的，在你编辑构成某一程序的模块时，将它们加到“程序模块表”中，在你重新建立该程序时，快速 C 只是重新编译那些自上次程序建立以来修改过的模块，以此节省你的时间。

程序模块表保存在一个与 MAKE 程序维护实用程序兼容的文件中；因而，程序可以在快速 C 程序设计环境外自动更新。

利用下面这些附加特性取得所想要的结果

Microsoft 快速 C 提供下面这些附加特性：

- 一个完善的实用程序工具包，你几乎能满足任何程序设计的要求，该工具包包括 QCL 编译程序/连接程序驱动器；
Microsoft 覆盖连接程序 LINK；Microsoft 库管理程序 LIB；以及 Microsoft 程序维护实用程序 MAKE。
- 使用四种内存模型中的任何一种简化内存管理。QCL 编译程序/连接程序驱动器支持小内存、中内存、紧致内存和大内存模型。
- 出类拔萃的文档参考资料。文档资料包括完整的 C 程序设计语言和标准 C 库例程序的参考资料，其中包括数百个完整的程序例子。

系统所需配置

Microsoft 快速 C 编译器至少需要下述一些设备配置：

- 一台运行 MS-DOS 或 PC-DOS 2.1 版本或其后版本的 IBM 个人计算机或者严格与其兼容的机器。
- 两个软盘驱动器或者一个软盘驱动器和一张硬盘。
- 448k 可用 RAM 内存。

使用本手册

Microsoft 快速 C 编译器是为广泛范围的用户设计的，从对程序设计感到陌生的初学者到能够熟练使用 C 语言的有经验的程序开发者。手册的不同部分满足不同需要的用户。

下面的各段解释本手册是如何组织的，给不同类型的用户建议不同的途径，并描述在该软件包中提供的其它手册。

本手册是如何组织的

手册的第一部分“启动”，解释如何装配 Microsoft 快速 C 编译器，向你介绍快速 C 程序设计环境，并为熟悉诸如 BASIC 或 Pascal 语言的用户描述 C 语言的重要特性，该部分还告诉你如何使用由 Microsoft 快速 C 编译器提供的图形库。

第二部分，“快速 C 程序设计环境”，解释如何在快速 C 程序设计环境内对程序进行编辑，编译和调试。

本手册的第三部分，“快速 C 工具包”，解释如何用 QCL 及 LINK 程序编译和连接程序；如何用 LINK 建立快速库，如何用 LIB 实用程序建立独立的库；以及如何用 MAKE 实用程序对程序进行维护。

手册的附录部分包含你可能发现会有用的附加信息，诸如：

- ASCII 字符码表
- 快速 C 内存模型描述
- Microsoft C 和 Microsoft 宏汇编程序 (MASM) 之间的接口描述
- 错误信息表

本手册中简便快速的参考卡片列出快速 C 编辑和调试键及其键序列。

阅读手册的哪些内容

所有用户都应该阅读第一章，“装配并启动快速 C”，得到装配编译器软件的指示。如果你想用编译器做一些介绍性的手头练习，你会想浏览 § 1.7.4 中描述的简短样例对话。

下面要干什么取决于你的特殊需求，如下所述：

对程序设计和 C 语言陌生的用户

如果你正在同时学习程序设计和 C 语言，参考本引言部分“学习资料”中所列的教材。

对 C 语言不熟悉的程序员

如果你有一些程序设计背景，但是以前从来没有用 C 语言写过程序，参考第 3 章，“C 快速启动”，获取一些关于 C 的特性及 C 和其它如 BASIC 和 Pascal 语言之间的差别的信息。

想要用快速 C 图形编制程序的用户

阅读第四章，“图形快速启动”，获取用 Microsoft 快速 C 编译器提供的图形函数编写程序的方法。

对集成化程序设计环境陌生的用户

阅读第五章，“快速 C 程序设计环境简介，”获取快速 C 程序设计环境组成部分的概况，然后阅读第二部分，“快速 C 程序设计环境”，学习如何使用建立在快速 C 环境中的编辑器、编译器以及调试程序。

对快速 C 提供的其它工具感兴趣的用户

阅读第三部分，“快速 C 工具包”，了解 QCL 编译程序/连接程序驱动器，LINK 连接程序，LIB 库管理程序以及 MAKE 程序维护实用程序。

你可以在快速 C 中得到的其它手册

Microsoft 快速 C 编译器软件包中还提供了两本附加手册：

- 《Microsoft C 语言参考手册》，它描述了 C 语言的 Microsoft ANSI-兼容的实现
- 《Microsoft C 运行库参考手册》，它描述了 Microsoft 快速 C 编译器提供的 350 多个库例行程序和 34 个内含文件。

记号约定

本手册使用如下这些记号：

约定的例子	约定的描述
Example (例子)	左列所示的字样用来模拟将要打印到屏幕上或通过打印机印出的信息的外形
placeholders (定位)	斜体字是命令行中的定位式你必须提供的信息的类型任选项说明。斜体字有时还在正文中用来表示强调。
DOS NAMES (DOS 名), FILE NAMES (文件名), 和 MACROS (宏)	黑体大写字母用来表示可执行文件的名字以及由和产品一起提供的文件名，表示环境变量，表示替换常数，表示宏，表示寄存器、这些命令包括 DOS 内部命令，如 SET，也包括和快速 C 软件包一起提供的文件名。
Reserved words (保留字)	黑体表示正文必须按所示的字体打出。通常用黑体表示的正文包括运算符，关键字，库函数，命令，任选项和预处理

理程序伪指令, 例子包括 C 关键字 `int` 和函数名 `fopen`。
中括号括起来的为命令行中的任选域及任选选项说明。
{选择1|选择2} 大括号和一竖线表示你可以在两项式或多项之中选择。选择用大括号括起来, 用竖线将它们分开。
“术语定义” 引号使正文中定义的术语更明显, 例如, 术语“`far` (段外)”第一次定义时出现在引号中, 引号还使正文中的命令行提示符更明显清晰。
有一些 C 结构需要引号。语言所需的引号为“ ”而不是“ ”。
重复元素…… 跟在某一项目后面三点 (也称为“省略号”) 表示可以键入更多相同形式的项目。

键名 语法行和程序例子中的一列是表示程序的一部分被省略。小的大写字母用来表示你必须按的键的名字。键序列用通过加号 (+) 隔开来的小的大写字母表示。例如包括 `ENTER` 和 `CTRL+C`。

本手册中键的名字对应于 IBM 个人计算机键板上打印出的名字。如果你使用的是其它机器, 这些键的名字可能稍微有些不一样。主键盘右边的数字键盘上的光标移动键 (有时也称为“箭头”) 称做方向键。每个单独的方向键或者指键板上箭头的方向, 或者指键板上的名字 (前者如 `LEFT`, `RIGHT`, `UP`, `DOWN`, 后者如 `PGTP` 和 `PGDN`)。回车键表示为 `ENTER`。

学习资料

你或许想用下面所列的书籍进一步开发 C。它按困难度递增的顺序列出, 从开始到高级。

Hancock, Les, and Morris Krieger. *The C Primer, 2d ed.* New York: McGraw-Hill, 1985. (A beginner's guide to programming in the C language.)

Schilat, Herbert. *C Made Easy.* Berkeley, CA: Osborne/McGraw-Hill, 1985. (A good introduction to C for the reader who already knows BASIC.)

Waite, Mitchell, Stephen Prata, and Donald Martin. *C Primer plus.* Indianapolis, IN: Howard W. Sams, Inc., 1984. (The best-selling introduction to the C language.)

Plum, Thomas. *Learning to Program in C.* Cardiff, New Jersey: Plum Hall, Inc., 1983. (A widely used introductory college text on computer programming using the C language.)

Kochan, Stephen. *Programming in C.* Hasbrouck Heights, NJ: Hayden Book Company, Inc, 1983. (A comprehensive introduction to C with some emphasis on the UNIX environment.)

Harbison, Samuel P. and Guy L. Steele, Jr. *C: A Reference Manual*, 2d ed. Englewood Cliffs, NJ: Prentice-Hall, Inc., 1987. (An outstanding reference to the C language. The second edition incorporates the ANSI standard.)

Kernighan, Brian W., and Dennis M. Ritchie. *The C Programming Language*. Englewood Cliffs, NJ: Prentice-Hall, Inc., 1978. (The original, classic C book, known to insiders as "K & R" or as "the white book." Useful after you have learned C.)

Tondo, Clovis L., and Scott E. Gimple. *The C Answer Book*. Englewood Cliffs, NJ: Prentice-Hall, Inc., 1985. (A collection of answers to the exercises in K & R. A companion volume to K & R.)

Jaesche, Rex. *Solutions in C*. Reading, Massachusetts: Addison Wesley, 1986. (A useful collection of C programming tips.)

Ward, Robert. *Debugging C*. Indianapolis, Indiana: Que Corporation, 1983. (A guide to the techniques of debugging C programs.)

Schustack, Steve. *Variations in C*. Redmond, Washington: Microsoft Press, 1985. (A guide to programming business applications in C.)*

Hansen, Angie. *Proficient C*. Redmond, Washington: Microsoft Press, 1987. (A guide to advanced C programming in the MS-DOS environment.)*

从 Microsoft 获取帮助

如果你感到在软件中发现了问题, 请用本手册后面的 Product Assistance Request (产品辅助要求) 报告该问题。

如果你对本产品的手册有什么注释或建议, 请使用 Documentation Feedback card (文档反馈卡片)。

Howard W. Sams, Inc., 1984. (The best-selling introduction to the C language.)

Pitman, Thomas. *Learning to Program in C*. Englewood Cliffs, NJ: Prentice-Hall, Inc., 1983. (A widely used introductory college text on computer programming using the C language.)

Kochan, Stephen. *Programming in C*. Englewood Cliffs, NJ: Prentice-Hall, Inc., 1988. (A comprehensive introduction to C with some emphasis on the UNIX environment.)

Walters, D. J. *C Programming Language*. Englewood Cliffs, NJ: Prentice-Hall, Inc., 1984. (A widely used introductory college text on computer programming using the C language.)

Schilf, Herbert. *C Made Easy*. Berkeley, CA: Osborne/McGraw-Hill, 1985. (A good introduction to C for the reader who already knows BASIC.)

Harcovitz, Les and Morris Krueger. *The C Primer*. Englewood Cliffs, NJ: Prentice-Hall, Inc., 1985. (A beginner's guide to programming in the C language.)