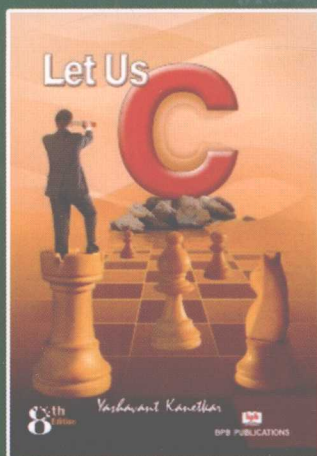


C程序设计基础教程

(第八版)

Let Us C, Eighth Edition

英文版



[印度] Yashavan P. Kanetkar 著



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>

国外计算机科学教材系列

C 程序设计基础教程

(第八版)

(英文版)

Let Us C

Eighth Edition

[印度] Yashavan P. Kanetkar 著

電子工業出版社

Publishing House of Electronics Industry

北京 · BEIJING

内 容 简 介

本书是全球畅销的一本 C 语言编程经典教材,这一版本除了讲解 C 语言中基本的函数、指针、循环、控制、数据类型、数组、字符串、输入/输出、位操作等内容外,书中还用大量的篇幅和实例重点突出了 C 语言在 Internet 编程、Windows 编程以及 Linux 编程中的用法,使读者能尽快熟悉 C 语言的实际应用。

本书结构合理,内容深入浅出,适合作为高等学校本科和专科学习 C 语言编程的教材,也适合初学编程的自学者。

Yashavant P. Kanetkar

Let Us C

Copyright © 2007, 2006, 2005, 2002, 1999, 1995, 1991 by BPB Publications, India

Authorized bilingual from English language edition, entitled LET US C, 8th Edition, 978-81-8333-163-0, originally published by BPB Publications, India copyright © 2007

Publishing House of Electronics Industry is authorized by Cengage Learning to publish and distribute exclusively this bilingual edition. This edition is authorized for sale in the Mainland of China. Unauthorized export of this edition is a violation of the Copyright Act. No part of this publication may be reproduced or distributed by any means, or stored in a database or retrieval system, without the prior written permission of the publisher.

本书中英双语版专有版权由圣智学习出版公司授予电子工业出版社。此版本仅限在中国大陆出版发行。未经授权的本书出口将被视为违反版权法的行为。未经出版者预先书面许可,不得以任何方式复制或发行本书的任何部分。

Cengage Learning Asia Pte. Ltd.

5 Shenton Way, #01-01 UIC Building, Singapore 068808

本书封底贴有 Cengage Learning 防伪标签,无标签者不得销售。

版权贸易合同登记号 图字: 01-2009-3832

图书在版编目(CIP)数据

C 程序设计基础教程: 第 8 版: 英文/(印)卡内特卡(Kanetkar, Y. P.)著. —北京: 电子工业出版社, 2009.8
(国外计算机科学教材系列)

ISBN 978-7-121-09267-1

I. C… II. 卡… III. C 语言—程序设计—教材—英文 IV. TP312

中国版本图书馆 CIP 数据核字(2009)第 119282 号

策划编辑: 许菊芳

责任编辑: 许菊芳

印 刷: 北京市天竺颖华印刷厂

装 订: 三河市鑫金马印装有限公司

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本: 787×980 1/16 印张: 29.25 字数: 655 千字

印 次: 2009 年 8 月第 1 次印刷

定 价: 49.00 元

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系及邮购电话: (010) 88254888。

质量投诉请发邮件至 zltz@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线: (010) 88258888。

出版说明

21 世纪初的 5 至 10 年是我国国民经济和社会发展的关键时期,也是信息产业快速发展的关键时期。在我国加入 WTO 后的今天,培养一支适应国际化竞争的一流 IT 人才队伍是我国高等教育的重要任务之一。信息科学和技术方面人才的优劣与多寡,是我国面对国际竞争时成败的关键因素。

当前,正值我国高等教育特别是信息科学领域的教育调整、变革的重大时期,为使我国教育体制与国际化接轨,有条件的高等院校正在为某些信息学科和技术课程使用国外优秀教材和优秀原版教材,以使我国在计算机教学上尽快赶上国际先进水平。

电子工业出版社秉承多年来引进国外优秀图书的经验,翻译出版了“国外计算机科学教材系列”丛书,这套教材覆盖学科范围广、领域宽、层次多,既有本科专业课程教材,也有研究生课程教材,以适应不同院系、不同专业、不同层次的师生对教材的需求,广大师生可自由选择和自由组合使用。这些教材涉及的学科方向包括网络与通信、操作系统、计算机组织与结构、算法与数据结构、数据库与信息处理、编程语言、图形图像与多媒体、软件工程等。同时,我们也适当引进了一些优秀英文原版教材,本着翻译版本和英文原版并重的原则,对重点图书既提供英文原版又提供相应的翻译版本。

在图书选题上,我们大都选择国外著名出版公司出版的高校教材,如 Pearson Education 培生教育出版集团、麦格劳-希尔教育出版集团、麻省理工学院出版社、剑桥大学出版社等。撰写教材的许多作者都是蜚声世界的教授、学者,如道格拉斯·科默(Douglas E. Comer)、威廉·斯托林斯(William Stallings)、哈维·戴特尔(Harvey M. Deitel)、尤利斯·布莱克(Uyless Black)等。

为确保教材的选题质量和翻译质量,我们约请了清华大学、北京大学、北京航空航天大学、复旦大学、上海交通大学、南京大学、浙江大学、哈尔滨工业大学、华中科技大学、西安交通大学、国防科学技术大学、解放军理工大学等著名高校的教授和骨干教师参与了本系列教材的选题、翻译和审校工作。他们中既有讲授同类教材的骨干教师、博士,也有积累了几十年教学经验的老教授和博士生导师。

在该系列教材的选题、翻译和编辑加工过程中,为提高教材质量,我们做了大量细致的工作,包括对所选教材进行全面论证;选择编辑时力求达到专业对口;对排版、印制质量进行严格把关。对于英文教材中出现的错误,我们通过与作者联络和网上下载勘误表等方式,逐一进行了修订。

此外,我们还将与国外著名出版公司合作,提供一些教材的教学支持资料,希望能为授课老师提供帮助。今后,我们将继续加强与各高校教师的密切联系,为广大师生引进更多的国外优秀教材和参考书,为我国计算机科学教学体系与国际教学体系的接轨做出努力。

电子工业出版社

教材出版委员会

主 任	杨芙清	北京大学教授 中国科学院院士 北京大学信息与工程学部主任 北京大学软件工程研究所所长
委 员	王 珊	中国人民大学信息学院院长、教授
	胡道元	清华大学计算机科学与技术系教授 国际信息处理联合会通信系统中国代表
	钟玉琢	清华大学计算机科学与技术系教授、博士生导师 清华大学深圳研究生院信息学部主任
	谢希仁	中国人民解放军理工大学教授 全军网络技术研究中心主任、博士生导师
	尤晋元	上海交通大学计算机科学与工程系教授 上海分布计算技术中心主任
	施伯乐	上海国际数据库研究中心主任、复旦大学教授 中国计算机学会常务理事、上海市计算机学会理事长
	邹 鹏	国防科学技术大学计算机学院教授、博士生导师 教育部计算机基础课程教学指导委员会副主任委员
	张昆藏	青岛大学信息工程学院教授

About the Author

关于作者

Through his books and Quest! Video Courses on C, C++, Data Structures, VC++, .NET, Embedded Systems, etc. Yashavant Kanetkar has created, moulded and groomed lacs of IT careers in the last decade and half. In recognition of his immense contribution to IT education in India, he has been awarded the “**Best .NET Technical Contributor**” and “**Most Valuable Professional**” awards by Microsoft. His current passion is Internet Technologies, Device Drivers and Embedded Systems. He is currently scripting a book titled “Go Embedded!”. Yashavant holds a BE from VJTI Mumbai and M.Tech. from IIT Kanpur. Yashavant’s current affiliations include being a Director of KICIT and KSET. He can be reached at kanetkar@kicit.com or through www.kicit.com.

Acknowledgments

致谢

Let Us C has become an important part of my life. I have created and nurtured it for last decade or so. While doing so, I have received, in addition to the compliments, a lot of suggestions from students, developers, professors, publishers and authors. So much have their inputs helped me in taking this book up to this eighth edition, that ideally I should put their names too on the cover page.

In particular I am indebted to Manish Jain who had a faith in this book idea, believed in my writing ability, whispered the words of encouragement and made helpful suggestions from time to time. I hope every author gets a publisher who is as cooperative, knowledgeable and supportive as Manish.

The seven editions of this book saw several changes and facelifts. During this course many people helped in executing programs, spotting bugs, drawing figures and preparing index. I trust that with their collective acumen, all the programs would run correctly in all situations. I value the work that they did a lot. Any errors, omissions or inconsistencies that remain are, alas, my responsibility.

I thank Seema, my wife, for enduring the late nights, the clicking keyboard and the noisy printer and mostly for putting up with yet another marathon book effort.

And finally my heartfelt gratitude to the countless students who made me look into every nook and cranny of C. I want to remain in their debt.

Preface to the Eighth Edition

第八版前言

The aim of “Let Us C” for its first few editions was to teach C programming in its entirety in as simple a manner as possible. From fifth edition onwards it took a slightly different turn. It started talking about use of C language in different programming environments like Windows and Linux. From this edition onwards I have

decided to also address the issues and topics where I find many students and professionals face difficulties while doing real-life programming. One topic that came to my mind immediately is Internet programming. So you will find in this edition a complete chapter devoted to it. It discusses the TCP/IP model and usage of different protocols like HTTP, Whois, Time, SMTP, POP3, etc. I sincerely believe that if you try programs in this chapter you will grow in confidence to do larger projects in Internet programming.

Preface to the Seventh Edition

第七版前言

“It all depends on how you C it”.

Having spent a decade and a half in computing, working with tools and technologies that have evolved over the years, I am convinced that a programmer’s life should begin with C. Unless he is comfortable with C, he would find it extremely difficult to master newer tools and technologies. With that motive, as promised last year, I am coming out with another edition of your favorite “Let Us C” with an updated view of how C is being used in practice today. I hope you would find this edition useful.

Countless Indians have relentlessly worked for close to two decades to successfully establish “India” as a software brand. At times, I take secret pleasure in seeing that a book that I have been part of, has contributed in its own little way in shaping so many careers that have made the “India” brand acceptable.

Computing and the way people use C for doing it keeps changing as years go by. So overwhelming has been the response to all the previous editions of “Let Us C” that to keep the readers abreast with the way C is being used today a new edition of “Let Us C” is now published every year.

There are two phases in every C programmer’s life. In the first phase, he is a learner trying to understand the language elements and their nuances. At this stage, he wants a simple learning environment that helps him to master the language. In my opinion, even today there isn’t any learning environment that can beat Turbo C/C++ for simplicity. Hence the first fifteen chapters are written keeping this environment in mind, though a majority of the programs in these chapters would work with any C compiler.

Armed with the knowledge of language elements, the C programmer enters the second phase. Here he wishes to use all that he has learnt to create programs that match the ability of programs that he comes across in today’s world. I am pointing towards programs in Windows and Linux world. Chapters 16 to 20 are devoted to this. I would like to bring to your attention the fact that if you want to program Windows or Linux you need to have a very good grasp over the programming model used by each of these OS. Windows messaging architecture and Linux signaling mechanism are the cases in point. Once you understand these thoroughly, rest is just a matter of time. Chapters 16 to 20 have been written with this motive.

In Linux programming, the basic hurdle is in choosing the Linux distribution, compiler, editor, shell, libraries, etc. To get a headstart, you can follow the choices that I found most reasonable and simple. They have been mentioned in Chapter 20 and Appendix F. Once you are comfortable with Linux, you can explore the other choices on your own.

In fourth edition of Let Us C there were chapters on ‘Disk Basics’, ‘VDU Basics’, ‘Graphics’, ‘Mouse Programming’, ‘C and Assembly’. Though I used to like these chapters a lot I had to take a decision to drop them since most of them were DOS-centric and would not be so useful in modern-day programming. Modern

counterparts of all of these have been covered in Chapters 16 to 20. However, if you still need the chapters from previous edition they are available at www.kicit.com/books/letusc/fourthedition.

Also, all the programs present in the book are available in source code form at www.kicit.com/books/letusc/sourcecode. You are free to download them, improve them, change them, do whatever with them. If you wish to get solutions for the Exercises in the book they are available in another book titled 'Let Us C Solutions'.

If you like 'Let Us C' and want to hear the complete video-recorded lectures created by me on C language (and other subjects like C++, VC++, .NET, Embedded Systems, etc.), then you can visit quest.ksetindia.com for more details.

'Let Us C' is as much your book as it is mine. So if you feel that I could have done certain job better than what I have, or you have any suggestions about what you would like to see in the next edition, please drop a line to letuscsuggestions@kicit.com.

All the best and happy programming!

目 录

Chapter 1: Getting Started	1
1.1 What is C	2
1.2 Getting Started with C	3
1.2.1 The C Character Set	3
1.2.2 Constants, Variables and Keywords	4
1.2.3 Types of C Constants	4
1.2.4 Rules for Constructing Integer Constants	5
1.2.5 Rules for Constructing Real Constants	5
1.2.6 Rules for Constructing Character Constants	6
1.2.7 Types of C Variables	6
1.2.8 Rules for Constructing Variable Names	7
1.2.9 C Keywords	7
1.3 The First C Program	8
1.4 Compilation and Execution	11
1.5 Receiving Input	12
1.6 C Instructions	14
1.6.1 Type Declaration Instruction	14
1.6.2 Arithmetic Instruction	15
1.6.3 Integer and Float Conversions	17
1.6.4 Type Conversion in Assignments	17
1.6.5 Hierarchy of Operations	19
1.6.6 Associativity of Operators	20
1.7 Control Instructions in C	22
1.8 Summary	22
1.9 Exercise	23
Chapter 2: The Decision Control Structure	30
2.1 Decisions! Decisions!	30
2.2 The <i>if</i> Statement	31
2.2.1 The Real Thing	33
2.2.2 Multiple Statements within <i>if</i>	34
2.3 The <i>if-else</i> Statement	35
2.3.1 Nested <i>if-elses</i>	37
2.3.2 Forms of <i>if</i>	38

2.4	Use of Logical Operators	39
2.4.1	The <i>else if</i> Clause	41
2.4.2	The <i>!</i> Operator	44
2.4.3	Hierarchy of Operators Revisited	45
2.5	A Word of Caution	45
2.6	The Conditional Operators	47
2.7	Summary	48
2.8	Exercise	48
Chapter 3: The Loop Control Structure		60
3.1	Loops	60
3.2	The <i>while</i> Loop	61
3.2.1	Tips and Traps	62
3.2.2	More Operators	65
3.3	The <i>for</i> Loop	67
3.3.1	Nesting of Loops	70
3.3.2	Multiple Initialisations in the <i>for</i> Loop	71
3.4	The Odd Loop	72
3.5	The <i>break</i> Statement	73
3.6	The <i>continue</i> Statement	74
3.7	The <i>do-while</i> Loop	75
3.8	Summary	76
3.9	Exercise	77
Chapter 4: The Case Control Structure		84
4.1	Decisions Using <i>switch</i>	84
4.1.1	The Tips and Traps	87
4.2	<i>switch</i> Versus if-else Ladder	90
4.3	The <i>goto</i> Keyword	90
4.4	Summary	92
4.5	Exercise	92
Chapter 5: Functions & Pointers		97
5.1	What is a Function	97
5.1.1	Why Use Functions	102
5.2	Passing Values between Functions	103
5.3	Scope Rule of Functions	106
5.4	Calling Convention	106
5.5	One Dicey Issue	107
5.6	Advanced Features of Functions	108

5.6.1	Return Type of Function	108
5.6.2	Call by Value and Call by Reference	109
5.6.3	An Introduction to Pointers	109
5.6.4	Pointer Notation	109
5.6.5	Back to Function Calls	113
5.6.6	Conclusions	115
5.6.7	Recursion	116
5.6.8	Recursion and Stack	119
5.7	Adding Functions to the Library	120
5.8	Summary	122
5.9	Exercise	123
Chapter 6: Data Types Revisited		132
6.1	Integers, <i>long</i> and <i>short</i>	132
6.2	Integers, <i>signed</i> and <i>unsigned</i>	134
6.3	Chars, <i>signed</i> and <i>unsigned</i>	134
6.4	Floats and Doubles	135
6.5	A Few More Issues... ..	137
6.6	Storage Classes in C	138
6.6.1	Automatic Storage Class	138
6.6.2	Register Storage Class	140
6.6.3	Static Storage Class	141
6.6.4	External Storage Class	142
6.6.5	A Few Subtle Issues	144
6.6.6	Which to Use When	145
6.7	Summary	146
6.8	Exercise	146
Chapter 7: The C Preprocessor		152
7.1	Features of C Preprocessor	152
7.2	Macro Expansion	153
7.2.1	Macros with Arguments	155
7.2.2	Macros versus Functions	158
7.3	File Inclusion	158
7.4	Conditional Compilation	159
7.5	#if and #elif Directives	162
7.6	Miscellaneous Directives	163
7.6.1	#undef Directive	163
7.6.2	#pragma Directive	163
7.7	The Build Process	165

7.7.1	Preprocessing	166
7.7.2	Compilation	166
7.7.3	Assembling	166
7.7.4	Linking	167
7.7.5	Loading	168
7.8	Summary	168
7.9	Exercise	169
Chapter 8:	Arrays	172
8.1	What are Arrays	172
8.1.1	A Simple Program Using Array	173
8.2	More on Arrays	175
8.2.1	Array Initialisation	175
8.2.2	Bounds Checking	176
8.2.3	Passing Array Elements to a Function	177
8.3	Pointers and Arrays	178
8.3.1	Passing an Entire Array to a Function	183
8.3.2	The Real Thing	183
8.4	Two Dimensional Arrays	184
8.4.1	Initialising a 2-Dimensional Array	185
8.4.2	Memory Map of a 2-Dimensional Array	186
8.4.3	Pointers and 2-Dimensional Arrays	186
8.4.4	Pointer to an Array	188
8.4.5	Passing 2-D Array to a Function	189
8.5	Array of Pointers	192
8.6	Three-Dimensional Array	193
8.7	Summary	194
8.8	Exercise	195
Chapter 9:	Puppetting On Strings	210
9.1	What are Strings	210
9.2	More about Strings	211
9.3	Pointers and Strings	214
9.4	Standard Library String Functions	215
9.4.1	strlen()	215
9.4.2	strcpy()	217
9.4.3	strcat()	219
9.4.4	strcmp()	219
9.5	Two-Dimensional Array of Characters	220
9.6	Array of Pointers to Strings	222

9.7	Limitation of Array of Pointers to Strings	224
9.7.1	Solution	225
9.8	Summary	226
9.9	Exercise	226
Chapter 10:	Structures	231
10.1	Why Use Structures	231
10.1.1	Declaring a Structure	233
10.1.2	Accessing Structure Elements	235
10.1.3	How Structure Elements are Stored	235
10.2	Array of Structures	236
10.3	Additional Features of Structures	237
10.4	Uses of Structures	243
10.5	Summary	244
10.6	Exercise	244
Chapter 11:	Console Input/Output	249
11.1	Types of I/O	249
11.2	Console I/O Functions	250
11.2.1	Formatted Console I/O Functions	250
11.2.2	<i>sprintf()</i> and <i>sscanf()</i> Functions	256
11.2.3	Unformatted Console I/O Functions	256
11.3	Summary	259
11.4	Exercise	259
Chapter 12:	File Input/Output	263
12.1	Data Organization	264
12.2	File Operations	264
12.2.1	Opening a File	265
12.2.2	Reading from a File	266
12.2.3	Trouble in Opening a File	266
12.2.4	Closing the File	267
12.3	Counting Characters, Tabs, Spaces,	268
12.4	A File-copy Program	269
12.4.1	Writing to a File	269
12.5	File Opening Modes	270
12.6	String (line) I/O in Files	270
12.6.1	The Awkward Newline	272
12.7	Record I/O in Files	272
12.8	Text Files and Binary Files	275

12.9	Record I/O Revisited	277
12.10	Database Management	279
12.11	Low Level Disk I/O	283
12.11.1	A Low Level File-copy Program	283
12.12	I/O Under Windows	286
12.13	Summary	286
12.14	Exercise	287
Chapter 13: More Issues In Input/Output		294
13.1	Using <i>argc</i> and <i>argv</i>	294
13.2	Detecting Errors in Reading/Writing	297
13.3	Standard I/O Devices	298
13.4	I/O Redirection	299
13.4.1	Redirecting the Output	299
13.4.2	Redirecting the Input	300
13.4.3	Both Ways at Once	301
13.5	Summary	301
13.6	Exercise	302
Chapter 14: Operations On Bits		303
14.1	Bitwise Operators	303
14.1.1	One's Complement Operator	305
14.1.2	Right Shift Operator	306
14.1.3	Left Shift Operator	307
14.1.4	Bitwise AND Operator	310
14.1.5	Bitwise OR Operator	313
14.1.6	Bitwise XOR Operator	313
14.2	The <i>showbits</i> () Function	314
14.3	Hexadecimal Numbering System	315
14.4	Relation between Binary and Hex	316
14.5	Summary	317
14.6	Exercise	317
Chapter 15: Miscellaneous Features		320
15.1	Enumerated Data Type	320
15.1.1	Uses of Enumerated Data Type	321
15.1.2	Are Enums Necessary	323
15.2	Renaming Data types with typedef	323
15.3	Typecasting	325
15.4	Bit Fields	326

15.5	Pointers to Functions	327
15.6	Functions Returning Pointers	329
15.7	Functions with Variable Number of Arguments	330
15.8	Unions	332
15.8.1	Union of Structures	336
15.8.2	Utility of Unions	337
15.9	The volatile Qualifier	338
15.10	Summary	339
15.11	Exercise	339
Chapter 16:	C Under Windows	342
16.1	Which Windows... ..	343
16.2	Salient Features of Windows Programming	343
16.2.1	Powerful API Functions	344
16.2.2	Sharing of Functions	344
16.2.3	Consistent Look and Feel	344
16.2.4	Hardware Independent Programming	345
16.2.5	Event Driven Programming Model	345
16.3	Obvious Programming Differences	347
16.3.1	Integers	347
16.3.2	Heavy Use of typedef	347
16.3.3	Size of Pointers	348
16.4	The First Windows Program	349
16.5	Hungarian Notation	351
16.6	Role of the Message Box	351
16.7	Here Comes the window... ..	352
16.8	More Windows	354
16.9	A Real-World Window	355
16.9.1	Creation and Displaying of Window	356
16.9.2	Interaction with Window	356
16.9.3	Reacting to Messages	357
16.10	Program Instances	359
16.11	Summary	359
16.12	Exercise	360
Chapter 17:	Graphics Under Windows	361
17.1	Graphics as of Now	361
17.2	Device Independent Drawing	362
17.3	Hello Windows	362
17.4	Drawing Shapes	365

17.5	Types of Pens	367
17.6	Types of Brushes	369
17.6.1	Code and Resources	371
17.7	Freehand Drawing, the Paintbrush Style	371
17.7.1	Capturing the Mouse	373
17.8	Device Context, a Closer Look	374
17.9	Displaying a Bitmap	375
17.10	Animation at Work	377
17.10.1	WM_CREATE and OnCreate ()	379
17.10.2	WM_TIMER and OnTimer ()	380
17.10.3	A Few More Points... ..	380
17.11	Windows, the Endless World... ..	381
17.12	Summary	381
17.13	Exercise	382
Chapter 18: Internet Programming		383
18.1	Network Communication	383
18.2	Packets and Sockets.....	385
18.3	Before We Start... ..	385
18.3.1	Protocols	385
18.3.2	IP Addresses	386
18.3.3	Port Numbers.....	386
18.3.4	Byte Ordering	387
18.4	Getting Started... ..	387
18.5	What's The Time Now	389
18.5.1	Creation of Socket	392
18.5.2	Sending Data to a Time Server	392
18.5.3	Receiving Date and Time	393
18.6	Communicating with Whois Server	393
18.7	Give Me the Home Page	396
18.8	Sending and Receiving Emails.....	397
18.9	Two-Way Communication	405
18.10	Summary	410
18.11	Exercise	410
Chapter 19: C Under Linux		411
19.1	What is Linux	411
19.2	C Programming Under Linux	412
19.3	The 'Hello Linux' Program	412
19.4	Processes	413

19.5	Parent and Child Processes	414
19.6	More Processes	417
19.7	Zombies and Orphans	418
19.8	One Interesting Fact	419
19.9	Summary	420
19.10	Exercise	420
Chapter 20: More Linux Programming		422
20.1	Communication using Signals	422
20.2	Handling Multiple Signals	424
20.3	Registering a Common Handler	425
20.4	Blocking Signals	426
20.5	Event Driven Programming	428
20.6	Where Do You Go From Here	431
20.7	Summary	431
20.8	Exercise	432
Appendix A: Precedence Table		433
Appendix B: Library Functions		434
Appendix C: Chasing The Bugs		440
Appendix D: ASCII Chart		444
Appendix E: Helper.h File		447
Appendix F: Linux Installation		448