

# Let Us

# C



*Yashwant Kanethkar*

8th  
Edition



BPB PUBLICATIONS

Scanned by TapScanner



# Contents

## 1. Getting Started

	1
What is C	2
Getting Started with C	4
The C Character Set	5
Constants, Variables and Keywords	6
Types of C Constants	7
Rules for Constructing Integer Constants	8
Rules for Constructing Real Constants	9
Rules for Constructing Character Constants	10
Types of C Variables	11
Rules for Constructing Variable Names	11
C Keywords	12
The First C Program	13
Compilation and Execution	19
Receiving Input	21
C Instructions	23
Type Declaration Instruction	24
Arithmetic Instruction	25
Integer and Float Conversions	29
Type Conversion in Assignments	30
Hierarchy of Operations	32
Associativity of Operators	35
Control Instructions in C	37
Summary	38
Exercise	39

## 2. The Decision Control Structure

	49
Decisions! Decisions!	50
The <i>if</i> Statement	51
The Real Thing	55
Multiple Statements within <i>if</i>	56
The <i>if-else</i> Statement	58



Nested <i>if-elses</i>	61
Forms of <i>if</i>	62
Use of Logical Operators	64
The <i>else if</i> Clause	67
The ! Operator	72
Hierarchy of Operators Revisited	73
A Word of Caution	74
The Conditional Operators	76
Summary	78
Exercise	79
<b>3. The Loop Control Structure</b>	<b>97</b>
Loops	98
The <i>while</i> Loop	99
Tips and Traps	101
More Operators	105
The <i>for</i> Loop	108
Nesting of Loops	114
Multiple Initialisations in the <i>for</i> Loop	115
The Odd Loop	116
The <i>break</i> Statement	118
The <i>continue</i> Statement	120
The <i>do-while</i> Loop	121
Summary	123
Exercise	124
<b>4. The Case Control Structure</b>	<b>135</b>
Decisions Using <i>switch</i>	136
The Tips and Traps	140
<i>switch</i> Versus <i>if-else</i> Ladder	144
The <i>goto</i> Keyword	145
Summary	149
Exercise	149



<b>5. Functions &amp; Pointers</b>	<b>155</b>
What is a Function	156
Why Use Functions	164
Passing Values between Functions	165
Scope Rule of Functions	170
Calling Convention	170
One Dicey Issue	172
Advanced Features of Functions	173
Return Type of Function	174
Call by Value and Call by Reference	175
An Introduction to Pointers	175
Pointer Notation	176
Back to Function Calls	182
Conclusions	186
Recursion	186
Recursion and Stack	191
Adding Functions to the Library	194
Summary	198
Exercise	198
<b>6. Data Types Revisited</b>	<b>211</b>
Integers, <i>long</i> and <i>short</i>	212
Integers, <i>signed</i> and <i>unsigned</i>	214
Chars, <i>signed</i> and <i>unsigned</i>	215
Floats and Doubles	217
A Few More Issues...	219
Storage Classes in C	221
Automatic Storage Class	222
Register Storage Class	224
Static Storage Class	226
External Storage Class	229
A Few Subtle Issues	232
Which to Use When	234
Summary	235



Exercise	235
<b>7. The C Preprocessor</b>	<b>243</b>
Features of C Preprocessor	244
Macro Expansion	245
Macros with Arguments	249
Macros versus Functions	253
File Inclusion	254
Conditional Compilation	256
<i>#if</i> and <i>#elif</i> Directives	260
Miscellaneous Directives	261
<i>#undef</i> Directive	262
<i>#pragma</i> Directive	262
The Build Process	265
Preprocessing	267
Compilation	267
Assembling	267
Linking	269
Loading	270
Summary	271
Exercise	272
<b>8. Arrays</b>	<b>277</b>
What are Arrays	278
A Simple Program Using Array	280
More on Arrays	283
Array Initialization	283
Bounds Checking	284
Passing Array Elements to a Function	285
Pointers and Arrays	287
Passing an Entire Array to a Function	294
The Real Thing	295
Two Dimensional Arrays	297
Initializing a 2-Dimensional Array	298



Memory Map of a 2-Dimensional Array	299
Pointers and 2-Dimensional Arrays	300
Pointer to an Array	303
Passing 2-D array to a Function	305
Array of Pointers	308
Three Dimensional Array	310
Summary	312
Exercise	312
<b>9. Puppeting On Strings</b>	<b>335</b>
What are Strings	336
More about Strings	337
Pointers and Strings	342
Standard Library String Functions	344
<i>strlen()</i>	345
<i>strcpy()</i>	347
<i>strcat()</i>	351
<i>strcmp()</i>	351
Two-Dimensional Array of Characters	353
Array of Pointers to Strings	356
Limitation of Array of Pointers to Strings	359
Solution	360
Summary	362
Exercise	362
<b>10. Structures</b>	<b>371</b>
Why Use Structures	372
Declaring a Structure	375
Accessing Structure Elements	378
How Structure Elements are Stored	378
Array of Structures	379
Additional Features of Structures	382
Uses of Structures	391
Summary	392



Exercise	393
<b>11. Console Input/Output</b>	<b>401</b>
Types of I/O	402
Console I/O Functions	403
Formatted Console I/O Functions	404
<i>sprintf( )</i> and <i>scanf( )</i> Functions	413
Unformatted Console I/O Functions	414
Summary	417
Exercise	418
<b>12. File Input/Output</b>	<b>423</b>
Data Organization	424
File Operations	425
Opening a File	426
Reading from a File	427
Trouble in Opening a File	428
Closing the File	430
Counting Characters, Tabs, Spaces, ...	430
A File-copy Program	432
Writing to a File	433
File Opening Modes	433
String (line) I/O in Files	435
The Awkward Newline	437
Record I/O in Files	438
Text Files and Binary Files	441
Record I/O Revisited	445
Database Management	449
Low Level Disk I/O	455
A Low Level File-copy Program	454
I/O Under Windows	461
Summary	461
Exercise	462



**13. More Issues In Input/Output** **473**

Using <i>argc</i> and <i>argv</i>	474
Detecting Errors in Reading/Writing	478
Standard I/O Devices	480
I/O Redirection	481
Redirecting the Output	482
Redirecting the Input	484
Both Ways at Once	485
Summary	486
Exercise	486

**14. Operations On Bits** **489**

Bitwise Operators	490
One's Complement Operator	492
Right Shift Operator	494
Left Shift Operator	496
Bitwise AND Operator	501
Bitwise OR Operator	506
Bitwise XOR Operator	507
The <i>showbits()</i> Function	508
Hexadecimal Numbering System	509
Relationship between Binary and Hex	510
Summary	512
Exercise	513

**15. Miscellaneous Features** **517**

Enumerated Data Type	518
Uses of Enumerated Data Type	519
Are Enums Necessary	522
Renaming Data Types with <i>typedef</i>	522



Typecasting	525
Bit Fields	526
Pointers to Functions	529
Functions Returning Pointers	531
Functions with Variable Number of Arguments	533
Unions	537
Union of Structures	543
Utility of Unions	544
The <i>volatile</i> Qualifier	546
Summary	547
Exercise	547
<b>16. C Under Windows</b>	<b>551</b>
Which Windows...	552
Integers	553
Heavy Use of <i>typedef</i>	553
Size of Pointers	555
DOS Programming Model	550
Windows Programming Model	559
Event Driven Model	563
Windows Programming, a Closer Look	564
The First Windows Program	566
Hungarian Notation	570
Role of the Message Box	570
Here Comes the window...	571
More Windows	574
A Real-World Window	576
Creation and Displaying of Window	578
Interaction with Window	579
Reacting to Messages	580
Program Instances	583
Summary	583
Exercise	584
<b>17. Graphics Under Windows</b>	<b>587</b>



Graphics as of Now	588
Device Independent Drawing	588
Hello Windows	590
Drawing Shapes	594
Types of Pens	598
Types of Brushes	600
Code and Resources	604
Freehand Drawing, the Paintbrush Style	604
Capturing the Mouse	608
Device Context, a Closer Look	609
Displaying a Bitmap	611
Animation at Work	615
WM_CREATE and <i>OnCreate()</i>	619
WM_TIMER and <i>OnTimer()</i>	619
A Few More Points...	620
Windows, the Endless World...	622
Summary	622
Exercise	623
<b>18. Internet Programming</b>	<b>625</b>
Network Communication	624
Packets and Sockets	626
Before We Start	627
Protocols	627
IP Addresses	629
Port Numbers	629
Byte Ordering	631
Getting Started...	631
What's The Time Now	634
Creation of Socket	638
Sending Data to a Time Server	639
Receiving Date and Time	640
Communicating with Whois Server	641
Give Me the Home Page	645



Sending and Receiving Emails	647
Two-Way Communication	659
Summary	666
Exercise	666
<b>19. C Under Linux</b>	<b>669</b>
What is Linux	670
C Programming Under Linux	671
The 'Hello Linux' Program	672
Processes	674
Parent and Child Processes	675
More Processes	679
Zombies and Orphans	681
One Interesting Fact	683
Summary	684
Exercise	685
<b>20. More Linux Programming</b>	<b>687</b>
Communication using Signals	688
Handling Multiple Signals	691
Registering a Common Handler	693
Blocking Signals	695
Event Driven Programming	698
Where Do You Go From Here	703
Summary	704
Exercise	704
<i>Appendix A – Precedence Table</i>	691
<i>Appendix B – Standard Library Functions</i>	695
<i>Appendix C – Chasing the Bugs</i>	703
<i>Appendix D – ASCII Chart</i>	711
<i>Appendix E – Helper.h File</i>	717
<i>Appendix F – Boot Parameters</i>	721
<i>Index</i>	731