

Contents

<i>Preface</i>	<i>vii</i>
<i>List of Programs</i>	<i>xvii</i>

CHAPTER 1: Basics of Programming	1.1–1.30
---	-----------------

1.1	Introduction	1.1
1.2	Problem Solving Phase	1.1
1.3	Software Design Techniques	1.6
1.4	Top-Down Design Approach	1.7
1.5	Programming Methodologies	1.8
1.6	Program Analysis Chart (PAC)	1.10
1.7	Flowchart	1.11
1.8	Pseudocode	1.17
1.9	Algorithm	1.19
1.10	Loop Constructs	1.24
1.11	Summary of Important Concepts	1.29

CHAPTER 2: Fundamentals of C	2.1–2.56
-------------------------------------	-----------------

2.1	Introduction	2.1
2.2	The C character set	2.2
2.3	Data Types	2.2
2.3.1	Primary Data Types	2.2
2.3.2	User-Defined Data Type	2.4
2.3.3	Enumerated Data Type	2.5
2.3.4	Empty Data Type	2.5
2.4	Arrays	2.5
2.5	Variables	2.6
2.5.1	Declaration of Variables	2.7
2.5.2	Initialization of Variables	2.7
2.5.3	Classification of Variables	2.8
2.6	Tokens	2.8
2.6.1	Identifiers	2.9
2.6.2	Keywords	2.10
2.7	Constants	2.10
2.7.1	Integer Constants	2.10
2.7.2	Real Constants	2.11
2.7.3	Single Character Constants	2.12

2.7.4 String Constants	2.12
2.8 Escape sequences	2.13
2.9 Operators	2.13
2.9.1 Arithmetic Operators	2.14
2.9.2 Unary Operators	2.15
2.9.3 The sizeof Operator	2.16
2.9.4 Relational Operators	2.16
2.9.5 Equality Operators	2.16
2.9.6 Logical Operators	2.17
2.9.7 Bitwise Operators	2.18
2.9.8 Conditional Operator	2.20
2.9.9 Assignment Operators	2.21
2.9.10 Type Casting	2.21
2.9.11 Special Operators	2.22
2.9.12 Precedence and Associativity of Operators	2.23
2.10 Statements	2.25
2.10.1 Expression Statements	2.25
2.10.2 Compound Statements	2.25
2.10.3 Control Statements	2.26
2.11 Expressions	2.26
2.12 Library Functions	2.27
2.13 Header Files	2.28
2.14 Structure of a C program	2.28
2.15 The main() Function	2.31
2.16 Input and Output in C	2.32
2.16.1 Formatted I/O Functions	2.32
2.16.2 Unformatted I/O Functions	2.36
2.17 Additional Programs	2.38
2.18 Summary of Important Concepts	2.42
2.19 Short Answer Questions	2.43
2.20 Exercises	2.50

CHAPTER 3: Control Statements

3.1–3.36

3.1 Introduction	3.1
3.2 The if Statement	3.1
3.3 The if-else Statement	3.3
3.4 Multiple if-else Statement	3.5
3.5 Nested if-else Statement	3.7
3.6 The Switch Statement	3.9
3.7 Conditional Operator	3.12
3.8 Loop Statements	3.13

3.8.1 The while Loop	3.14
3.8.2 The do-while Loop	3.15
3.8.3 The for Loop	3.16
3.9 The break Statement	3.18
3.10 The continue Statement	3.18
3.11 The goto Statement	3.19
3.12 Additional Programs	3.21
3.13 Summary of Important Concepts	3.29
3.14 Short Answer Questions	3.30
3.15 Exercises	3.32

CHAPTER 4: Functions**4.1–4.30**

4.1 Introduction	4.1
4.2 Defining a Function	4.2
4.3 The return Statement	4.5
4.4 Types of Functions	4.6
4.4.1 Function with no Arguments and no Return Value	4.6
4.4.2 Function with Return Value and no Arguments	4.7
4.4.3 Function with Arguments and no Return Value	4.8
4.4.4 Function with Arguments and Return Value	4.9
4.5 Nested Functions	4.9
4.6 Recursion	4.10
4.7 Storage Classes	4.12
4.7.1 Automatic Storage Class	4.12
4.7.2 Register Storage Class	4.13
4.7.3 Static Storage Class	4.14
4.7.4 External Storage Class	4.15
4.8 Additional Programs	4.17
4.9 Summary of Important Concepts	4.21
4.10 Short Answer Questions	4.22
4.11 Exercises	4.25

CHAPTER 5: Arrays and Strings**5.1–5.38**

5.1 Introduction	5.1
5.2 Single Dimensional Arrays	5.1
5.2.1 Declaring Single Dimensional Arrays	5.2
5.2.2 Accessing Single Dimensional Array Elements	5.3
5.2.3 Initializing Single Dimensional Arrays	5.4
5.3 Double Dimensional Arrays	5.6
5.3.1 Initializing Double Dimensional Arrays	5.7
5.4 Multidimensional Arrays	5.9

5.5	Strings	5.10
5.5.1	Declaring Strings	5.10
5.5.2	Initializing Strings	5.11
5.5.3	Reading Strings	5.11
5.6	String Library Functions	5.14
5.6.1	The strlen() Function	5.15
5.6.2	The strcpy() Function	5.15
5.6.3	The strcat() Function	5.16
5.6.4	The strcmp() Function	5.17
5.6.5	The strrev() Function	5.19
5.6.6	The strlwr() Function	5.19
5.7	Additional Programs	5.20
5.8	Summary of Important Concepts	5.32
5.9	Short Answer Questions	5.33
5.10	Exercises	5.35

CHAPTER 6: Pointers**6.1–6.24**

6.1	Introduction	6.1
6.2	Pointer Variables	6.1
6.2.1	Initialization of Pointers	6.3
6.2.2	Pointer Arithmetic	6.3
6.3	Pointers and Arrays	6.3
6.4	Array of Pointers	6.5
6.5	Pointer to Functions	6.6
6.6	Pointer to Pointers	6.7
6.7	Passing Values to Functions	6.8
6.7.1	Pass by Value	6.8
6.7.2	Pass by Reference	6.9
6.8	Passing Arrays to Functions	6.10
6.8.1	Passing Array Element to a Function	6.10
6.8.2	Passing an Entire Array to a Function	6.11
6.9	Memory Allocation	6.12
6.9.1	Static Memory Allocation	6.12
6.9.2	Dynamic Memory Allocation	6.12
6.10	Summary of Important Concepts	6.20
6.11	Short Answer Questions	6.20
6.12	Exercises	6.22

CHAPTER 7: Structures and Unions**7.1–7.22**

7.1	Introduction	7.1
7.2	Declaring a Structure	7.1

7.3	Defining a Structure	7.2
7.4	Accessing Structure Members	7.4
7.5	Initializing a Structure	7.4
7.6	Nested Structures	7.6
7.7	Array of Structures	7.8
7.8	Structures and Functions	7.11
7.8.1	Passing a Structure Member to Functions	7.11
7.8.2	Passing the Address of Members to Functions	7.12
7.8.3	Passing Entire Structure to Functions	7.12
7.9	Structures and Pointers	7.13
7.10	Bit Fields	7.14
7.11	Unions	7.15
7.11.1	Initializing an Union	7.17
7.12	Summary of Important Concepts	7.17
7.13	Short Answer Questions	7.18
7.14	Exercises	7.20

CHAPTER 8: File Handling**8.1–8.28**

8.1	Introduction	8.1
8.2	Standard Files	8.1
8.3	Defining and Opening a File	8.2
8.4	Closing a File	8.4
8.5	The fprintf() and fscanf() Functions	8.5
8.6	Character Input / Output from Files	8.6
8.7	String Input / Output from Files	8.7
8.8	End of File	8.8
8.9	The stdin, stdout and stderr Pointers	8.8
8.10	Error Handling During Input / Output Operations	8.9
8.11	Random Access Functions	8.10
8.11.1	The fseek() Function	8.10
8.11.2	The ftell() Function	8.11
8.11.3	The rewind() Function	8.11
8.12	Merging Two Files	8.13
8.13	The Preprocessor	8.20
8.13.1	File Inclusion	8.21
8.13.2	Macro Definition	8.21
8.13.3	Conditional Compilation	8.24
8.14	Summary of Important Concepts	8.25
8.15	Short Answer Questions	8.26
8.16	Exercises	8.27

CHAPTER 9: Sorting and Searching	9.1-9.30
9.1 Introduction	9.1
9.2 Bubble Sort	9.2
9.3 Insertion Sort	9.4
9.4 Selection Sort	9.6
9.5 Shell Sort	9.8
9.6 Quick Sort	9.11
9.7 Merge Sort	9.14
9.8 Linear Search	9.17
9.9 Binary Search	9.18
9.10 Additional Programs	9.20
9.11 Summary of Important Concepts	9.25
9.12 Short Answer Questions	9.25
9.13 Exercises	9.28
<i>Appendices</i>	<i>A.1–A4.2</i>
Appendix 1: ASCII Code	A.1
Appendix 2: Precedence Table	A.5
Appendix 3: Header Files	A.6
Appendix 4: Library Functions	A.9
Appendix 5: Common Programming Errors	A.18
Appendix 6: Keys used in C Compiler	A.25
Appendix 7: Glossary	A.28
Appendix 8: Important Program Coding	A.34
<i>Index</i>	<i>I.1–I.4</i>

List of Programs

Chapter 2

Program 2.1 : Sample program of C language.	2.28
Program 2.2 : Program to demonstrate main() function.	2.32
Program 2.3 : Program to demonstrate usage of scanf() function.	2.33
Program 2.4 : Program 1 to demonstrate usage of printf() function.	2.35
Program 2.5 : Program 2 to demonstrate usage of printf() function.	2.36
Program 2.6 : Program 1 to demonstrate usage of gets() and puts() function.	2.38
Program 2.7 : Program for swapping (interchanging) two numbers (using three variables).	2.38
Program 2.8 : Program for swapping (interchanging) two numbers (using two variables).	2.39
Program 2.9 : Program for swapping two numbers without using arithmetic operators.	2.39
Program 2.10 : Program to convert temperature in centigrade to fahrenheit.	2.40
Program 2.11 : Program to convert polar form to rectangular form.	2.40
Program 2.12 : Program to find simple and compound interest.	2.41

Chapter 3

Program 3.1 : Program to find the biggest of two numbers using an if statement .	3.3
Program 3.2 : Program to check whether the no is odd or even using an if-else statement.	3.4
Program 3.3 : Program to demonstrate usage of multiple if-else statement.	3.6
Program 3.4 : Program to demonstrate usage of nested if-else statement.	3.7
Program 3.5 : Program to demonstrate the usage of switch statement.	3.10
Program 3.6 : Program for switch case execution without break statement.	3.11
Program 3.7 : Program to find the biggest of two numbers using conditional operator.	3.12
Program 3.8 : Program to find the biggest of two numbers using ternary operator.	3.13
Program 3.9 : Program to demonstrate the usage of while loop.	3.14
Program 3.10: Program to demonstrate the usage of do-while loop.	3.16
Program 3.11: Program to find the sum of first N natural numbers using for loop.	3.17
Program 3.12: Program to demonstrate the usage of break statement.	3.18
Program 3.13: Program to demonstrate the usage of continue statement.	3.19
Program 3.14: Program to demonstrate the usage of goto statement.	3.20
Program 3.15 : Program to find the maximum of three numbers using an if statement.	3.21
Program 3.16 : Program to find whether the given number is odd or even.	3.21
Program 3.17 : Program to find whether the given year is leap or not.	3.22
Program 3.18 : Program to count the number of digits in an integer.	3.22
Program 3.19 : Program to find the sum of digits of an integer.	3.23
Program 3.20 : Program for reversing an integer.	3.23

Program 3.21 : Program to print all divisors of a given number.	3.24
Program 3.22 : Program to check whether the given number is palindrome or not.	3.24
Program 3.23 : Program to check whether the given number is prime or not.	3.25
Program 3.24 : Program to print prime numbers between any two given numbers.	3.25
Program 3.25 : Program to calculate $\sin(x)$ series.	3.26
Program 3.26 : Program to calculate the e^x series.	3.27
Program 3.27 : Program to generate fibonacci series.	3.27
Program 3.28 : Program to calculate sum of the series $1 + 2^2 + 4^2 + 6^2 + \dots + N^2$.	3.28
Program 3.29 : Program to calculate sum of the series $1 + 3^2 + 5^2 + 7^2 + \dots + N^2$.	3.29
Program 3.30 : Program to calculate sum of the series $1 - 3 + 5 - 7 + \dots + N^2$.	3.29

Chapter 4

Program 4.1 : Program to print a message using function.	4.1
Program 4.2 : Program for function with no arguments and no return value.	4.6
Program 4.3 : Program for function with return value and no arguments.	4.7
Program 4.4 : Program for function with arguments and no return value.	4.8
Program 4.5 : Program for function with arguments and return value.	4.9
Program 4.6 : Program to demonstrate the usage of nested functions.	4.10
Program 4.7 : Program to find factorial of a number using recursive function.	4.11
Program 4.8 : Program to demonstrate the automatic storage class.	4.13
Program 4.9 : Program to demonstrate the register storage class.	4.13
Program 4.10: Program to demonstrate the static storage class.	4.14
Program 4.11 : Program to demonstrate the usage of global variables.	4.15
Program 4.12 : Program using function to check whether the year is leap or not.	4.17
Program 4.13 : Program using function to calculate X^N .	4.17
Program 4.14 : Program using function to find the square of first N numbers and to calculate its sum.	4.18
Program 4.15 : Program to sort the numbers in ascending order using functions.	4.19
Program 4.16 : Program to read an integer of N digits and reduce the integer to a single digit.	4.20
Program 4.17 : Program to generate fibonacci series using recursion.	4.20
Program 4.18 : Program using recursion to find the sum of all elements of an array.	4.21

Chapter 5

Program 5.1 : Program to find the biggest of 10 numbers using arrays.	5.3
Program 5.2 : Program to calculate the sum of all elements in a matrix.	5.9
Program 5.3 : Program to find length of a string using <code>strlen()</code> function.	5.15
Program 5.4 : Program for copy a string using <code>strcpy()</code> function.	5.16
Program 5.5 : Program for concatenation of two strings using <code>strcat()</code> function.	5.17
Program 5.6 : Program for comparing two strings using <code>strcmp()</code> function.	5.17
Program 5.7 : Program for reversing a string using <code>strrev()</code> function.	5.19
Program 5.8 : Program for converting a string to lowercase using <code>strlwr()</code> function.	5.20
Program 5.9 : Program to read and reverse an array.	5.20

Program 5.10 : Program to find the smallest number & its position in an array.	5.21
Program 5.11 : Program to display only the negative elements of an array.	5.21
Program 5.12 : Program to find the biggest number in a matrix.	5.22
Program 5.13 : Program to calculate the sum of all elements in a matrix.	5.23
Program 5.14 : Program to find the diagonal of the given matrix and to find the sum of the diagonal elements.	5.23
Program 5.15 : Program to print the transpose of the given matrix.	5.24
Program 5.16 : Program to multiply two matrices.	5.25
Program 5.17 : Program to evaluate the determinant of a matrix.	5.26
Program 5.18: Program to find the length of a string (without using strlen() function).	5.27
Program 5.19 : Program to copy a string from one variable to another variable (without using strcpy() function).	5.28
Program 5.20 : Program to compare two strings for equality (without using strcmp() function).	5.28
Program 5.21 : Program to concatenate two strings (without using strcat() function).	5.29
Program 5.22 : Program to reverse a string (without using strrev() function).	5.30
Program 5.23 : Program to replace a character in a string.	5.30
Program 5.24 : Program to count the number of spaces in a given text.	5.31

Chapter 6

Program 6.1 : Program for accessing a variable through a pointer.	6.2
Program 6.2 : Program to find the biggest of N numbers in an array using pointers.	6.4
Program 6.3 : Program to demonstrate the usage of pointer to function.	6.6
Program 6.4 : Program to demonstrate pass by value.	6.8
Program 6.5 : Program to demonstrate pass by reference.	6.9
Program 6.6 : Program to pass array elements to a function.	6.10
Program 6.7 : Program to pass an entire array to a function.	6.11
Program 6.8 : Program to demonstrate the usage of malloc() function.	6.13
Program 6.9 : Program to demonstrate the usage of calloc() function.	6.15
Program 6.10 : Program to demonstrate the usage of realloc() function.	6.16

Chapter 7

Program 7.1 : Program to initialize the structure using the structure name.	7.5
Program 7.2 : Program to illustrate the nested structures.	7.7
Program 7.3 : Program to illustrate array of structures.	7.9
Program 7.4 : Program for initializing array of structures.	7.10
Program 7.5 : Program for passing structure to a function.	7.12
Program 7.6 : Program for demonstrating structure pointer.	7.14
Program 7.7 : Program to find the size of structure and union.	7.16
Program 7.8 : Program to demonstrate initialization of an union.	7.17