



**Unit I**

**Chapter 1 : Problem Solving, Programming and Python Programming 1-1 to 1-47**

**Syllabus : General Problem Solving Concepts :** Problem solving in everyday life, types of problems, problem solving with computers, difficulties with problem solving, problem solving aspects, top down design. Problem solving strategies.

**Program Design Tools :** Algorithms, Flowcharts and Pseudo-codes, implementation of algorithms.

**Basics of Python Programming :** Features of Python, History and Future of Python, Writing and executing Python program, Literal constants, Variables and identifiers, Data Types, Input operation, Comments, Reserved words, Indentation, Operators and expressions, Expressions in Python.

1.1 Problem Solving in Everyday Life ..... 1-1

1.2 Types of Problems ..... 1-3

1.2.1 Algorithmic ..... 1-3

1.2.2 Heuristic ..... 1-3

1.3 Problem Solving with Computers..... 1-4

1.4 Difficulties with Problem Solving..... 1-4

1.5 Problem Solving Aspect ..... 1-5

1.5.1 Problem Definition Phase ..... 1-5

1.5.2 Getting Started On a Problem ..... 1-5

1.5.3 The Use of Specific Examples..... 1-5

1.5.4 Similarities among Problems ..... 1-6

1.5.5 Working Backwards from the Solution..... 1-6

1.6 General Problem Solving Strategies..... 1-6

1.7 Top Down Design ..... 1-7

1.8 Algorithm ..... 1-10

1.8.1 Advantages of Algorithms..... 1-10

1.9 Generalized Algorithms ..... 1-11

1.9.1 Advantages ..... 1-11

1.9.2 Disadvantages..... 1-11

1.9.3 How to Make an Algorithm Generalized?..... 1-12

1.10 Infinite Loop..... 1-12

1.10.1 How to Avoid Infinite Loops ? ..... 1-12

1.11 Different Ways of Representing Algorithms ..... 1-12

1.11.1 Flowcharts..... 1-13

1.11.2 Pseudo Code ..... 1-13

1.11.3 Program ..... 1-13

1.12 Program Planning..... 1-13

1.12.1 Need of Program Planning or Designing..... 1-13

1.12.2 Program Planning Tools..... 1-13

1.13 Implementation of Algorithms..... 1-16

1.14 Introduction of Python ..... 1-17

1.15 Features of Python ..... 1-18

1.15.1 Easy to Use..... 1-18

1.15.2 High Level Language..... 1-18

1.15.3 Expressive Language ..... 1-18

1.15.4 Interpreted Language ..... 1-19

1.15.5 Platform Independent Language ..... 1-19

1.15.6 Free, Open Source and Redistribution Language ..... 1-19

1.15.7 Object-Oriented and Procedural Oriented Programming Language Support..... 1-19

1.15.8 Extensive Standard Library Support ..... 1-19

1.15.9 GUI Programming and Web Application Support..... 1-19

1.15.10 Integrated Language ..... 1-20

1.15.11 Portable..... 1-20

1.15.12 Python is Extensible Language ..... 1-20



1.15.13	Embeddable Language .....	1-20	1.19.9.2	Explicit Type Conversion .....	1-34
1.15.14	Python is Dynamically Typed.....	1-20	1.20	Input Output Operation .....	1-36
1.16	Python History .....	1-20	1.20.1	Python Input.....	1-36
1.16.1	Python Version .....	1-21	1.20.2	Python Output .....	1-36
1.17	Literals in Python.....	1-21	1.21	Comments.....	1-38
1.17.1	String Literals 8 .....	1-22	1.22	Reserved Words .....	1-38
1.17.2	Numeric Literals .....	1-23	1.23	Indentation .....	1-39
1.17.3	Boolean Literals.....	1-23	1.24	Operators and Expressions .....	1-40
1.17.4	Special Literals .....	1-24	1.24.1	Arithmetic Operator .....	1-40
1.17.5	Literal Collections .....	1-24	1.24.2	Comparison Operator.....	1-41
1.18	Python Variables and Constants .....	1-25	1.24.3	Assignment Operators.....	1-42
1.18.1	Declaring and Assigning a Value to a Variable in Python.....	1-25	1.24.4	Logical Operators .....	1-43
1.18.2	Changing the Value of a Variable .....	1-25	1.24.5	Bitwise Operators .....	1-44
1.18.3	Assigning Multiple Values to Multiple Variables .....	1-25	1.24.6	Membership Operators.....	1-44
1.18.4	Rules and Naming Convention for Variables and Constants .....	1-26	1.24.7	Identity Operators.....	1-45
1.19	Data Types.....	1-27	1.24.8	Operator Precedence .....	1-46
1.19.1	Python Numbers.....	1-27	1.25	Practice Questions and Answers.....	1-46
1.19.2	Python List .....	1-28			
1.19.3	Python Tuple .....	1-28			
1.19.4	Python Strings.....	1-29			
1.19.5	Python Set.....	1-30			
1.19.6	Python Dictionary .....	1-31			
1.19.7	type() Function .....	1-31			
1.19.8	Conversion between Data Types.....	1-32			
1.19.9	Python Type Conversion and Type Casting.....	1-33			
1.19.9.1	Implicit Type Conversion .....	1-33			

## Unit II

### Chapter 2 : Decision Control Statements 2-1 to 2-45

#### Syllabus :

Decision control statements, Selection/conditional branching statements : if, if-else, nested if, if-elif-else statements. Basic loop Structures/Iterative statements: while loop, for loop, selecting appropriate loop, Nested loops, The break, continue, pass, else statement used with loops. Other data types-Tuples, Lists and Dictionary.

2.1	Decision Control Statements .....	2-1
-----	-----------------------------------	-----



2.1.1	Selection or Conditional Branching.....	2-1	2.13.1	Creating or Declaring a Dictionary.....	2-38
2.1.2	Basic Loop Structure or Iterative or Repetitive Execution .....	2-1	2.13.2	Accessing Elements from Dictionary .....	2-39
2.2	If Statement.....	2-1	2.13.3	Updating a Dictionary .....	2-39
2.3	if-else Statement .....	2-2	2.13.4	Deleting Values from a Dictionary .....	2-40
2.4	if-elif-else Statements.....	2-4	2.13.5	View Keys and Values.....	2-41
2.5	Structures / Iterative Statements .....	2-6	2.13.6	Python Dictionary Methods.....	2-42
2.5.1	while Loop .....	2-6	2.13.7	Built-in Functions with Dictionary.....	2-44
2.6	for Loop.....	2-8	2.14	Practice Questions and Answers.....	2-45
2.6.1	The range() Function.....	2-10	<b>Unit III</b>		
2.6.2	Nested for Loop.....	2-10	<hr/>		
2.7	else with for and while Loop .....	2-12	<b>Chapter 3 : Functions and Modules</b> <span style="float: right;"><b>3-1 to 3-17</b></span>		
2.8	pass Loop.....	2-13	<b>Syllabus</b> : Need for functions, Function : definition, call, variable scope and lifetime, the return statement. Defining functions, Lambda or anonymous function, documentation string, good programming practices. Introduction to modules, Introduction to packages in Python, Introduction to standard library modules.		
2.9	Break Statement .....	2-15	3.1	Need of Functions .....	3-1
2.10	Continue Statement.....	2-18	3.2	Function Definition or Defining Function .....	3-1
2.11	Tuples .....	2-20	3.2.1	Call to a Function .....	3-2
2.11.1	Creating or Declaring a Tuple.....	2-21	3.3	Variable Scope and Lifetime.....	3-3
2.11.2	Accessing Tuple Elements .....	2-21	3.3.1	Local Variables / Objects.....	3-3
2.11.3	Changing a Tuple .....	2-24	3.3.2	Global Variables / Objects .....	3-3
2.11.4	Deleting a Tuple .....	2-25	3.4	Arguments to a Function .....	3-4
2.11.5	Tuple Methods.....	2-26	3.4.1	Types of Arguments .....	3-5
2.11.6	Advantages of Tuple over List .....	2-29	3.5	Return Statement .....	3-8
2.12	Lists.....	2-29	3.6	Anonymous Functions / Lambda Functions.....	3-10
2.12.1	Creating or Declaring a List .....	2-29	3.7	Documentation String.....	3-12
2.12.2	Accessing List Elements .....	2-30	3.8	Standard Libraries in Python .....	3-12
2.12.3	Changing a List .....	2-33	3.9	Introduction to Modules .....	3-12
2.12.4	Deleting a List .....	2-34			
2.12.5	List Methods.....	2-35			
2.13	Dictionary .....	2-38			



3.10	Introduction to Packages in Python .....	3-13
3.11	Good Programming Practices.....	3-13
3.12	Practice Questions and Answers.....	3-14

## Unit IV

### Chapter 4 : Strings 4-1 to 4-25

**Syllabus :** Strings and Operations-concatenation, appending, multiplication and slicing. Strings are immutable, strings formatting operator, built in string methods and functions. Slice operation, ord() and chr() functions, in and not in operators, comparing strings, iterating strings, the string module.

4.1	Strings.....	4-1
4.1.1	String Declaration.....	4-1
4.1.2	String Accessing .....	4-2
4.1.3	Escape Sequence or a Back-slash.....	4-3
4.2	String Slicing .....	4-3
4.3	String Concatenation.....	4-4
4.4	Multiplication or Repitition.....	4-4
4.5	String Formatters.....	4-5
4.5.1	f-strings .....	4-5
4.5.2	Format() method.....	4-5
4.5.3	% Operator.....	4-6
4.5.4	Template .....	4-6
4.6	String Methods and Functions .....	4-6
4.6.1	len().....	4-7
4.6.2	str().....	4-7
4.6.3	lower() and upper() .....	4-8
4.6.4	islower().....	4-8
4.6.5	isupper() .....	4-8
4.6.6	strip().....	4-9

4.6.7	isdigit() .....	4-9
4.6.8	isalpha() .....	4-10
4.6.9	isspace() .....	4-10
4.6.10	isalnum() .....	4-10
4.6.11	istitle() .....	4-11
4.6.12	capitalize().....	4-11
4.6.13	title().....	4-12
4.6.14	swapcase().....	4-12
4.6.15	startswith().....	4-13
4.6.16	endswith().....	4-13
4.6.17	find().....	4-14
4.6.18	replace().....	4-14
4.6.19	split() .....	4-15
4.6.20	lstrip() .....	4-15
4.6.21	rstrip().....	4-16
4.6.22	join() .....	4-16
4.6.23	center().....	4-16
4.6.24	count().....	4-17
4.6.25	ljust() .....	4-17
4.6.26	rjust().....	4-18
4.6.27	max().....	4-19
4.6.28	min().....	4-19
4.6.29	splitlines() .....	4-19
4.6.30	zfill() .....	4-20
4.6.31	ord() and chr() .....	4-20
4.7	String Operations .....	4-21
4.7.1	Comparison.....	4-21
4.7.2	Arithmetic .....	4-21
4.7.3	Membership .....	4-22



---

4.7.4	Identity .....	4-22	➤ <b>Mini Project.....</b>	<b>M-1 to M-11</b>
4.7.5	Logical.....	4-22		
4.8	String Module .....	4-24	➤ <b>Programming and Problem Solving Lab</b>	
4.8.1	String Module Constant .....	4-24	<b>Assignments.....</b>	<b>L-1 to L-15</b>
4.8.2	String Module capwords() Function .....	4-24		
4.9	Practice Questions and Answers.....	4-25	➤ <b>Appendix - A .....</b>	<b>A-1 to A-8</b>

---