Syllabus of Programming and Problem Solving Through Python Language (M3-R5)

17.5. Marks Distribution

Module Unit	Written Marks (Max.)
1. Introduction to Programming, Algorithm and Flowcharts to solve problems	20
2. Introduction to Python, Operators, Expressions and Python Statements, Sequence data types	30
3. Functions, File Processing, Modules	40
4. NumPy Basics	10
5. Total	100

17.6. Detailed Syllabus

(i) Introduction to Programming

The basic Model of computation, algorithms, flowcharts, Programming Languages, compilation, testing & debugging and documentation.

(ii) Algorithms and Flowcharts to Solve Problems

Flow Chart Symbols, Basic algorithms/flowcharts for sequential processing, decision based processing and iterative processing. Some examples like: Exchanging values of two variables, summation of a set of numbers, Decimal Base to Binary Base conversion, Reversing digits of an integer, GCD (Greatest Common Divisor) of two numbers, Test whether a number is prime, factorial computation, Fibonacci sequence, Evaluate 'sin x' as sum of a series, Reverse order of elements of an array, Find largest number in an array, Print elements of upper triangular matrix, etc.

(iii) Introduction to Python

Python Introduction, Technical Strength of Python, Introduction to Python Interpreter and program execution, Using Comments, Literals, Constants, Python's Built-in Data types, Numbers (Integers, Floats, Complex Numbers, Real, Sets), Strings (Slicing, Indexing, Concatenation, other operations on

Strings), Accepting input from Console, printing statements, Simple ‘Python’ programs.

(iv) Operators, Expressions and Python Statements

Assignment statement, expressions, Arithmetic, Relational, Logical, Bitwise operators and their precedence, Conditional statements: if, if-else, if-elif-else; simple programs, Notion of iterative computation and control flow –range function, While Statement, For loop, break statement, Continue Statement, Pass statement, else, assert.

(v) Sequence Data Types

Lists, tuples and dictionary, (Slicing, Indexing, Concatenation, other operations on Sequence datatype),concept of mutability, Examples to include finding the maximum, minimum, mean; linear search on list/tuple of numbers, and counting the frequency of elements in a list using a dictionary.

(vi) Functions

Top-down approach of problem solving, Modular programming and functions, Function parameters, Local variables, the Return statement, DocStrings, global statement, Default argument values, keyword arguments, VarArgs parameters.

Library function-input(), eval(),print(), String Functions: count(), find(), rfind(), capitalize(), title(), lower(), upper(), swapcase(), islower(), isupper(), istitle(), replace(), strip(), lstrip(), rstrip(), aplit(), partition(), join(), isspace(), isalpha(), isdigit(), isalnum(), startswith(), endswith(), encode(), decode(), String: Slicing, Membership, Pattern Matching, Numeric Functions: eval(), max(), min(), pow(), round(), int(), random(), ceil(), floor(), sqrt(), Date & Time Functions, Recursion.

(vii) File Processing

Concept of Files, File opening in various modes and closing of a file, Reading from a file, Writing onto a file, File functions-open(), close(), read(), readline(), readlines(),write(), writelines(),tell(),seek(), Command Line arguments.

(viii) Scope and Modules

Scope of objects and Names, LEGB Rule
Module Basics, Module Files as Namespaces, Import Model, Reloading Modules.

(ix) NumPy Basics

Introduction to NumPy, ndarray, datatypes, array attributes, array creation routines, Array From Existing Data, Array From Numerical Ranges, Indexing & Slicing.