Weekly Teaching Schedule for IT-323 Python Programming

Week 1

- Lecture #1: Introduction Of Computers
- Lecture #2: Hardware, I/O Devices

Week 2

- Lecture #1: Memory, Storage Devices, Software's
- Lecture #2: Information Technology, People and Communication, Data Processing, Bits ,Bytes and Words Windows Operating System and Managing Windows 2000/XP

Week #1

- **Lecture #1:** Introduction: Understating computer programming, how a computer program works
- **Lecture #2:** Installing and using a Python development environment (Anaconda + Spyder, Jupyter, Google Colab, etc.)
- **Lecture #3:** Modes of using Python (Interactive and Scripting).

Week #2

- Lecture #1: Input, Processing, Output: Understanding the importance of Python built-in functions and library functions,
- Lecture #2: Output statements for phrases,
- Lecture #3: Declaring and initializing variables of different data types,

Week #3

Lecture #7 Concept of mutability, Introduction to errors and exceptions, Input statements, Lecture #8 Output statements for printing values, Dealing with variables of different data types, Lecture #9 Statements for arithmetic expressions and assignments, Basic memory concepts in Python,

Week #4 Lecture #10 Importing libraries/modules and importing from libraries/modules, using library functions, Lecture #11 The math module. Lecture #12 Python's built-in data structures, Identity and Membership operators. (3 weeks)

Week #5 Lecture #13 Decision (Selection Structures) and Boolean Logic: Understanding of the flow control structures in computer programs, Relational operators and Boolean expressions, Lecture #14 Single-Selection structure (using if statement), Double-Selection structure (using if-else statement), Multiple-Selection structure (using ifelif-else statement), Lecture #15 Multiple-Selection structure: (using multiple nested if-else statements), Week #6 Lecture #16 Logical or Boolean operators. Lecture #17 Planning a computer program, Lecture #18 Representations of algorithms: (pseudo-code, program source-code and flowchart). (2 weeks) Week #7 Lecture #19 Count-Controlled Repetition Structures: Count-Controlled repetition using for Lecture #20 and while loops, Lecture #21 Generating sequences and sums. (1 week) Week #8 Lecture #22 Sentinel-

Controlled Repetition Structures: Sentinel-Controlled repetition using while Loop, Lecture #23 break and continue statement, Lecture #24 Sentinel-Controlled repetition using for loop. (1 week)

Week #9 Lecture #25 Introduction to Strings: String Manipulations, Lecture #26 Formatting strings: Adjusting case, Lecture #27 Adding and removing spaces, Week #10 Finding and replacing substrings, Splitting and partitioning strings. (2 weeks) User-Defined Functions: Defining and calling user-defined functions, Value-returning functions, Local variables, Global variables and constants, Insights into passing arguments to functions, Default arguments, Flexible arguments, Anonymous (lambda) function, Storing functions in modules. (2 weeks) Lists and Tuples: Creating 1D and 2D lists/tuples, Indexing, Slicing, Operations, Deep copy versus shallow copy, Lists/tuples as function arguments, Explicit traversing in lists, List comprehension. (2 weeks) Introduction to Miscellaneous Topics: NumPy Arrays, Catching Exceptions (try and except), Creating Dictionaries and Sets. (1 week)