COURSE STRUCTURE

Course Code	**CET1003A**			
Course Category	*Engineering Science*			
Course Title	**Programming and Problem Solving**			
Teaching Scheme and Credits	**L**	**T**	**Laboratory**	**Credits**
Weekly load hrs	**3**	**-**	**2**	**2+0+1=3**

Pre-requisites: Introductory Knowledge of Computers

Course Objectives:

1. **To understand the problem solving framework and approaches.**
2. **To learn the Programming Language constructs.**
3. **To acquire programming skills for problem solving.**

Course Outcomes:

After completion of this course students will be able to

1) Develop efficient logic and algorithm for solving a problem.
2) Analyze the given problem and solve it using suitable programming constructs.
3) Apply programming skills for solving real world problems.

Course Contents:

Basic Concepts of computers: Architecture of computer, Types of Processor, Primary Storage, Secondary Storage, Number Systems, Data Representation-Signed, Unsigned, one's complement, two's complement, Floating point, char, String. Introduction to system software: Operating system, Editor, Compiler, Assembler, Linker, Loader.

Introduction to Problem Solving: Problem solving process/framework, Algorithms, Pseudo-code and flowchart, Case study for Algorithm, flowchart and pseudo code: calculate slope of a line, Factorial, Fibonacci, snake and ladder, tic-tac-toe. Top down and Bottom up design approach, Software Development life cycle, Programming paradigms: Imperative, object oriented, functional and logic programming. Role of programming languages, need to study programming languages, Characteristics of Programming Languages.

Introductions to C: Fundamentals of C-Programming - Character Set, Identifiers and keywords, Data types, Constants, Variables, Operators, Expression, statements, Library Functions, Pre-processor directives. Data Input and Output, Control Structures- Decision making, Control Structures- Iterative, break and continue statements, Structure of C program, Coding conventions. Array-single, multidimensional arrays, String in C –standard string functions in string.h. Functions in C, recursion, Different parameter passing methods, Lifetime of variables, Scope rules: Static and Dynamic scope, user defined string functions, Structure, Pointers, Structure - Array of structure, Union, Pointers, Pointers and arrays, Dynamic allocation and its application, Files: Types of File, File operation, Processing File.

Introductions to Python Programming: History, Features, Setting up path, Working with python, Basic Syntax, Character Set, Python Tokens, Keywords, Identifiers, Literals, Variables, Operators and Basic and Built-in Data Types: Numeric: int, float, complex,Text:String,Sequence::list,tuple, range, Mapping: dictionary, Set: set , frozenset , Boolean : bool , Arrays data types, Input and Output in Python, Structure of Python program, Control
Structures: Conditional statement, Looping and Iteration, Range Functions, Functions & classes: In-built

functions and user-defined functions, recursive Functions, Classes & Objects, Data Handling: Python libraries for data handling. APIs in Python.

Laboratory Work:

List of Assignments *(any 10)*

No.	Title of the Assignment
1.	A) Write algorithm and draw a flowchart to log in to gmail account B) Write algorithm and draw flowchart to Calculate the volume and area of sphere using the formulas $V=4* Pi*r^3 /3$ and $A=4*pi*r^2$ where r is radius of square of the sphere.
2.	A) Write algorithm and draw flowchart to calculate roots of a quadratic equation. B) Write algorithm and draw flowchart to convert given decimal number to binary.
3.	Write a menu driven program in C to implement the basic arithmetic operations.
4.	Write a menu driven program to perform basic operation such as addition and subtraction of two matrices.
5.	Write a menu driven program to perform all string operations (user defined function)
6.	Write a C function to compute factorial of a given number using recursion.
7.	Write a C program to accept student details and display their result using array of structures.
8.	Write a C program to swap two numbers using pointers.
9.	Write a C program to copy contents of one file to another using File handling.
10.	Write a Python program to accept n numbers and sort them in ascending/descending order.
11.	Write a Python function that accepts a string and calculates the number of uppercase letters and lower-case letters.
12.	Develop an application using Python API (Domain specific).

Learning Resources:

Reference Books:

1. Pradeep Sinha, Priti Sinha, “Computer Fundamentals”, Sixth edition, bpb publication.
2. Ramon Mata-Toledo, Pauline K. Cushman, “Introduction to Computer Science”, Schaum’s Outline series.
3. Herbert Schildt, "C: The Complete Reference", Fourth Edition, McGraw Hill Professional.
4. Yashwant Kanetkar, “Let us C”, Fifteenth edition, bpb publication.
5. David Ascher and MARK LUTZ, “Learning Python”, Packt Publishing Limited.
6. Martin C. Brown, “Python: The Complete Reference”, McGraw Hill Education.

Web Resources:

Web Links:

1. http://www.studytonight.com/c/overview-of-c.php
2. https://www.tutorialspoint.com/cprogramming
3. https://www.realpython.com/
4. https://pythoninstitute.org/

MOOCs:

1. http://nptel.ac.in/courses/106105085/2
2. http://nptel.ac.in/courses/106104074/1
3. https://nptel.ac.in/courses/106/105/106105171
4. https://nptel.ac.in/courses/106/106/106106212/
5. https://www.coursera.org/projects/introduction-to-python
6. https://www.coursera.org/learn/python
7. https://www.udemy.com/course/the-python-mega-course/

Pedagogy:

- Power point presentations
- Practical Demos
- Videos
- Online Classroom
- Expert lectures

Syllabus:

Assessment Scheme:

Class Continuous Assessment (CCA): (50 marks)

Assignments	Test	Case study/Group activity Presentations	MCQ	Oral	Attendance
10 Marks 20%	15 Marks 30 %	5 Marks 10%	20 Marks 40%	Nil	Nil

Laboratory Continuous Assessment (LCA): (50 marks)

Each assignment is assessed out of 20 marks and total marks of all assignment is mapped to 50 marks.

Regularity, Punctuality and Ethics	Understanding the objective	Assignment Design	Programming Skills/Logic Design
5 Marks 25%	5 Marks 25%	5 Marks 25%	5 Marks 25%

Term End Examination: (50 marks)

Prepared By	Checked By	Approved By
Prof. Nital Adikane **Course Co-Ordinator**	**Dr. Vrushali Kulkarni** **HOS, School of CET** **MITWPU**	**Dr. Pradnya Deshpande** **HOS, School of FY BTech** **MITWPU**