Paper – 104 Computer Programming & Programming Methodology

(CPPM)

Course Code:	**104**
Course Title:	**Computer Programming & Programming Methodology (CPPM)**
Total Credits :	4 Credits
Nature of Subject :	Theory and Practical application
Teaching per Week:	4 Hours per week per Semester
Minimum weeks per Semester:	15 weeks (Including class work, examination, preparation etc.)
Review/Revision Year:	June, 2020
Purpose of Course :	- Computer programming (often shortened to programming) is a process that leads from an original formulation of a computing problem to executable computer programs. - Programming involves activities such as analysis, developing, understanding, generating algorithms, verification of requirements of algorithms including their correctness, and implementation (commonly referred to as coding) of algorithms in a target programming language. - Students pursuing their Graduation program will encounter their first programming language which is one of the pioneer computer programming languages. - Purpose of the course is to emphasis on concepts of Compiler based programming language, structure of code, algorithms, flow-charts, problem solving attitude, concepts of variables and declaration mechanism of different datatypes, simple I/O statements, conditional statements, loops, compound iterations, strings and certain inbuilt functions, header files, concepts of arrays and one dimensional numeric array operations, numeric inbuilt functions and concepts of pointers.
Objective :	Object of this course is to introduce students the essentials of computer Programming and programming methodology using C language.
Pre-requisite:	None
Course Outcome :	- Students will be able to formulate a computing problem to executable computer program using C language. - Understand about compiler based programming languages. - Concepts of variables, literals, data types, conversions of data types, input and output data and processing of data, inbuilt functions, arrays, header files, conditional and iterative statements.
Course Content:	**UNIT-1:** Introduction 1.1 Concepts of Programming Language 1.1.1 Introduction of Source Code, Object Code and executable code 1.1.2 Algorithm and Flowchart 1.1.3 Concepts of Structured Programming Language 1.2 Concepts of Editor, Interpreter and Compiler 1.2.1 Introduction of C program body structure 1.2.2 Character Set, concepts of variables and constants 1.2.3 Identifiers, literals, Key words 1.2.4 Data types (signed and unsigned) (Numeric : int, short int, long, float, double) , (Character type: char, string) and void. 1.2.5 Concepts of source code, object code and executable code.

UNIT-2: Input/Output Statements and Operators:

- 2.1 Input/Output statements:
  - 2.1.1 Concepts of Header files (STDIO,CONIO)
    - 2.1.1.1 Concepts of pre-compiler directives.
    - 2.1.1.2 Use of #inlcude and #define
- 2.2 Input/Output Statements:
  - 2.2.1 Input statements : scanf(), getc(), getch(), gets(), getchar()
  - 2.2.2 Output Statements: printf(), putc(),puts(), putchar()
  - 2.2.3 Type specifiers (formatting strings) : %d, %ld, %f, %c, %s, %lf
- 2.3 Operators :
  - 2.3.1 Arithmetic operators (+, -, *, /, %, ++, --,)
  - 2.3.2 Logical Operators (&&, ||, !)
  - 2.3.3 Relational Operators (>, <, ==, >=, <=, !=)
  - 2.3.4 Bit-wise operators (&, |, ^ , <<, >>)
  - 2.3.5 Assignment operators (=, +=, -=, *=, /=, %=)
  - 2.3.6 Ternary Operator and use of sizeof() function.
- 2.4 Important Built-in functions:
  - 2.4.1 Use of <string.h> : (strlen, strcmp, strcpy, strcat, strrev)
  - 2.4.2 Use of <math.h> : (abs(), floor(), round(), ceil(), sqrt(), exp(), log(), sin(), cos(), tan(), pow() and trunc())

UNIT-3: Decision Making statements :

- 3.1 if statements :
  - 3.1.1 simple if statements
  - 3.1.2 if…else statements
  - 3.1.3 if…else if….else statements
  - 3.1.4 Nested if statements.
- 3.2 Switch..case statements
  - 3.2.1 Use of break and default
  - 3.2.2 Difference between switch and if statements.

UNIT-4: Iterative statements :

- 4.1 Use of goto statement for iteration
- 4.2 while loop
- 4.3 do..while loop
- 4.4 for loop
- 4.5 Nested while, do..while and for loops
- 4.6 Jumping statement: (break and continue)

UNIT-5: Concepts of Arrays and pointer

- 5.1 Concepts of Single-dimensional Array
  - 5.1.1 Numeric single dimensional Array
  - 5.1.2 Numeric single dimensional array operations:
  - 5.1.2.1 Sorting array in ascending or descending. (Bubble and selection)
  - 5.1.2.2 Searching element from array (Linear Search)
  - 5.1.3 Character Single dimensional Array
  - 5.1.3.1 Character Single dimensional array operations:
  - 5.1.3.2 Use of \0, \n and \t
- 5.2 Pointers:
  - 5.2.1 Concepts of Pointers
  - 5.2.2 Declaring and initializing int, float, char and void pointers
  - 5.2.3 Pointer to single dimensional numeric array.

Reference Books:	1. Programming in C, Balaguruswami – TMH 2. C: How to Program, Deitel & Deitel - PHI 3. C Programming Language, Kernigham & Ritchie - TMH 4. Programming in C, Stephan Kochan - CBS 5. Mastering Turbo C, Kelly & Bootle - BPB 6. C Language Programming – Byron Gottfried - TMH 7. Let us C, Yashwant Kanetkar - BPB Publication 8. Magnifying C, Arpita Gopal - PHI 9. Problem Solving with C, Somashekara - PHI 10. Programming in C, Pradip Dey & Manas Ghosh – Oxford
Teaching Methodology:	Class Work, Discussion, Self-Study, Seminars and/or Assignments
Evaluation Method:	30% internal assessment. 70% External assessment