

**Course Code: 104****Course Title: Computer Programming & Programming Methodology (CPPM)**

Course Code	104																																																															
Course Title	Computer Programming & Programming Methodology (CPPM)																																																															
Credits	4																																																															
Course Category	Major Course																																																															
Level of Course	200-299 ( Intermediate Level )																																																															
Teaching per Week	4 Hrs.																																																															
Minimum weeks per Semester	15 (Including class work, examination, preparation etc.)																																																															
Review / Revision	2022-2023																																																															
Implementation Year:	A.Y. 2023-2024																																																															
Purpose of Course	<ul style="list-style-type: none"><li>- Computer programming is a process that leads from an original formulation of a computing problem to executable computer programs.</li><li>- 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.</li><li>- 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</li></ul>																																																															
Course Objective	To introduce students the essentials of computer Programming and programming methodology using C Programming language.																																																															
Pre-requisite	Concepts of Algorithms, flowcharts and basic Logical ability.																																																															
Course Outcomes	CO1: Students will be able to learn advanced programming concept of compiler based programming language. CO2: Students will be proficient working on conditional statements, iterative Statements and fundamentals of programming concepts using C and Python. CO3: Students will be able to represent compound data using lists, tuples and dictionaries in Python programs. CO4: Students will be able to develop real world application. CO5: Students will learn important libraries like Numpy, Pandas which are useful in Data analysis, Machine Learning.																																																															
Mapping between Course Outcomes(CO) with Program Specific Outcomes(PSO)	<table><tr><td></td><td>PSO1</td><td>PSO2</td><td>PSO3</td><td>PSO4</td><td>PSO5</td><td>PSO6</td><td>PSO7</td><td>PSO8</td></tr><tr><td>CO1</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>CO2</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>CO3</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>CO4</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>CO5</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table>											PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8	CO1									CO2									CO3									CO4									CO5								
	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8																																																								
CO1																																																																
CO2																																																																
CO3																																																																
CO4																																																																
CO5																																																																
Course Content	<b>UNIT-1: Introduction</b> 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,																																																															

	<p>double) , (Character type: char, string) and void.</p> <p>1.2.5 Concepts of source code, object code and executable code.</p> <p><b>UNIT-2: Input/Output Statements and Operators:</b></p> <p>2.1 Input/Output statements:</p> <p>2.1.1 Concepts of Header files (STDIO,CONIO)</p> <p>2.1.1.1 Concepts of pre-compiler directives.</p> <p>2.1.1.2 Use of #include and #define</p> <p>2.2 Input/Output Statements:</p> <p>2.2.1 Input statements : scanf(),getc(), getch(), gets(), getchar()</p> <p>2.2.2 Output Statements: printf(), putc(),puts(), putchar()</p> <p>2.2.3 Type specifiers (formatting strings) : %d, %ld, %f, %c, %s, %lf</p> <p>2.3 Operators :</p> <p>2.3.1 Arithmetic operators ( +, -, *, /, %, ++, --, )</p> <p>2.3.2 Logical Operators ( &amp;&amp;,   , ! )</p> <p>2.3.3 Relational Operators ( &gt;, =, &lt;=, != )</p> <p>2.3.4 Bit-wise operators ( &amp;,  , ^, &lt;&gt;)</p> <p>2.3.5 Assignment operators ( =, +=, -=, *=, /=, %=)</p> <p>2.3.6 Ternary Operator and use of sizeof() function.</p> <p>2.4 Important Built-in functions:</p> <p>2.4.1 Use of : ( strlen, strcmp, strcpy, strcat, strrev)</p> <p>2.4.2 Use of : (abs(), floor(), round(), ceil(), sqrt(), exp(), log(), sin(), cos(), tan(), pow() and trunc())</p> <p><b>UNIT-3: Decision Making statements :</b></p> <p>3.1 if statements :</p> <p>3.1.1 simple if statements</p> <p>3.1.2 if...else statements</p> <p>3.1.3 if...else if....else statements</p> <p>3.1.4 Nested if statements.</p> <p>3.2 Switch..case statements</p> <p>3.2.1 Use of break and default</p> <p>3.2.2 Difference between switch and if statements.</p> <p><b>UNIT-4: Iterative statements :</b></p> <p>4.1 Use of goto statement for iteration</p> <p>4.2 while loop</p> <p>4.3 do..while loop</p> <p>4.4 for loop</p> <p>4.5 Nested while, do..while and for loops</p> <p>4.6 Jumping statement: (break and continue)</p> <p><b>UNIT-5: Concepts of Arrays and pointer</b></p> <p>5.1 Concepts of Single-dimensional Array</p> <p>5.1.1 Numeric single dimensional Array</p> <p>5.1.2 Numeric single dimensional array operations:</p> <p>5.1.2.1 Sorting array in ascending or descending. (Bubble and selection)</p> <p>5.1.2.2 Searching element from array (Linear Search)</p> <p>5.1.3 Character Single dimensional Array</p> <p>5.1.3.1 Character Single dimensional array operations:</p> <p>5.1.3.2 Use of \0, \n and \t</p> <p>5.2 Pointers:</p> <p>5.2.1 Concepts of Pointers</p> <p>5.2.2 Declaring and initializing int, float, char and void pointers</p> <p>5.2.3 Pointer to single dimensional numeric array.</p>
--	--

<b>Reference Books</b>	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
<b>Teaching Methodology</b>	Class Work, Discussion, Lab work, Self-Study, Seminars and/or Assignments
<b>Evaluation Method</b>	30% Internal assessment. 70% External assessment.