

Course Code	<b>040110102</b>			
Category	<b>Core Subject</b>			
Course Title	<b>Programming in C Language ( C )</b>			
Scheme and Credits	<b>Theory</b>	<b>Tutorial</b>	<b>Lab</b>	<b>Credits</b>
	<b>3</b>	<b>1</b>	<b>4</b>	<b>6</b>
Pre-requisites (if any)	<b>No</b>			

### 1. Course Objectives:

<b>1</b>	To understand what is programming language
<b>2</b>	To develop ability to solve the given problem using flow chart and algorithm
<b>3</b>	To develop and enhance the problem solving skills using C Programming
<b>4</b>	To understand functional hierarchical code organization
<b>5</b>	To understand effective use of user define data types
	To be able to store & manage data in permanent storage media

### 2. Course contents :

Module	Content	Weightage
<b>Unit I</b>	<p><b>Introduction to programming in C:</b>            What is programming, Introduction to program and programming languages, Simple Program in C, compiler, interpreter, loader, linker, C program execution, Classification of Programming Languages, Flow Chart &amp; Algorithm</p> <p><b>Good programming practices:</b> Comments, Meaningful identifiers name, Escape sequence, Structure of C Program, Tokens, Concept of Identifier, Variable, Constant and Keywords, Data types in C.</p> <p><b>Basics of C:</b>            Program statements, declarations, How the computer stores data in memory, Operators and Expressions, L-values and R-values. Type casting &amp; type conversion.</p> <p><b>Input Output:</b> Basic Screen and Keyboard I/O in C, Unformatted Input and Output, Formatted Input, Output Functions and format specifier.</p>	<b>20%</b>
<b>Unit II</b>	<p><b>Control Statements, Arrays &amp; Strings:</b>  <b>Control Statements:</b>            Specifying test condition for selection and iteration, writing</p>	<b>25%</b>



	<p>test expression, Conditional execution and selection, Iteration and repetitive Execution: goto statement, for, while, do-while loops; variations of for loop, break, continue statement, nested loops.</p> <p><b>Arrays &amp; strings:</b> One-dimensional Array, Strings, String: One dimensional Array, Multi-dimensional array, Array of string. Writing first program with the help of flow chart and user define function.</p>	
<b>Unit III</b>	<p><b>Functions and Pointers:</b></p> <p><b>Function</b>-What is function, need of function, types of function, Passing array to Function, Scope and Extent, Storage class</p> <p><b>Recursion</b>-Concept of recursion, Types of recursion, comparison of iteration and recursion</p> <p><b>Pointer</b> - Concept of memory addresses, use of the (&amp;) operator.</p> <p><b>Pointer Execution</b> – Declaring and Initializing a pointer, Indirection Operator and dereferencing a pointer.</p> <p><b>Pointer Arithmetic</b> – Assignment, Addition-subtraction with integers, pointer comparison.</p> <p><b>Use of Pointers</b> – Returning more than one value from the function (Call by value and call by References concept).</p> <p><b>Pointers and Arrays</b> – One dimensional Arrays and pointers, Difference between array name and pointers, passing one dimensional and two dimensional array to functions, Pointers for character array(strings), Array of pointers 2-D Array and pointer, pointer to function.</p> <p><b>Dynamic memory allocation</b> – Array allocation, Memory freeing, Memory reallocation.</p> <p><b>Informative</b>- void Pointers, Null pointers, Pointers to pointers, Memory leak and Memory corruption, pointer to constant and constant pointer.</p>	<b>25%</b>
<b>Unit IV</b>	<p><b>User Defined Data types and Files in C:</b></p> <p><b>User Defined Data types – Basic structure operations</b>- Declaring structures and structure variables, Accessing the members of the structure, Initialization of structure, Structure comparisons and copying, typedef and its use.</p> <p><b>Advanced</b> - Nested structures, Array within the structure and Array of structures, Structures with functions and structures with pointers.</p> <p>Union and its operations Enumerated data type, bit field</p> <p><b>Files in C: Introduction – Streams in C, Types of Files.</b> <b>Files concept in C</b> – Declaration of File pointer, opening a file, closing a file various modes to open the file, Handling file functions for reading data from files– fscanf (), fgetw (),</p>	<b>20%</b>

	<p>fgets (), fgetc (), fread(), Handling file functions for writing data into files- fprintf, fputw, fputs(), fputc(), fwrite(), Detecting the end of the file – what is EOF and feof() function</p> <p><b>Types of the files</b> – Binary and Text files.</p> <p><b>Types of access to the file</b> – Sequential and Random access, Difference between both, how to read and write data in both the cases, for random access (fseek(), ftell(), rewind(), fgetpos(), fsetpos())</p> <p><b>Error handling during File operations</b></p> <p><b>Files of records</b> – Working with files of records (fread() and fwrite()).</p> <p><b>Other functions</b> – Renaming a file, Removing a file, Command line Arguments</p>	
<b>Unit V</b>	<p><b>Pre-processor directives:</b></p> <p><b>Pre-processor Directives</b>– Types of pre-processor directives, #define, #include, #undef, #line, pragma directives, Conditional directives, Predefined identifiers, Type Qualifiers, variable length arguments. recursion</p>	<b>10%</b>

### 3. Text Books:

- 1) Pradip Dey – Manas Ghosh, Programming in C Second Edition Oxford Publication).
- 2) Reema Thareja, "Programming in C", 2nd Edition, Oxford University Press.
- 3) Balagurusamy, Programming in ANSI C , Tata McGraw Hill.
- 4) Ashok N Kamthane, Programming with ANSI and Turbo C, Pearson Education.
- 5) Brian W. Kernigham, Dennis Ritchie, “The C Programming Language”, Pearson
- 6) Yeshvant Kanetkar, “Let Us C”, BPB Publication

### 4. Webilography:

- 1) ‘C’ Programming Language: <http://www.w3schools.in/cprogramming-language/intro/>
- 2) Learn C Online: <http://www.learnonline.com/>
- 3) ‘C’ Frequently Asked Questions: <http://www.c-faq.com>
- 4) ‘C’ Programming: <http://www.cprogramming.com>
- 5) Sams Teach Yourself C in 24 Hours: <http://aelinik.free.fr/c/>

### 5. Accomplishment of the student after completing the course:

After completion of the course students should become capable of solving problems using computers through C programming language.