

Course Code	18CSS101J	Course Name	PROGRAMMING FOR PROBLEM SOLVING	Course Category	S	Engineering Sciences	L	T	P	C
							3	0	4	5

Pre-requisite Courses	Nil	Co-requisite Courses	Nil	Progressive Courses	Nil
Course Offering Department	Computer Science and Engineering	Data Book / Codes/Standards	Nil		

Course Learning Rationale (CLR):	The purpose of learning this course is to:	Learning	Program Learning Outcomes (PLO)
CLR-1 :	Think and evolve a logically to construct an algorithm into a flowchart and a pseudocode that can be programmed	1 2 3	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
CLR-2 :	Utilize the logical operators and expressions to solve problems in engineering and real-time	Level of Thinking (Bloom)	Engineering Knowledge
CLR-3 :	Store and retrieve data in a single and multidimensional array	Expected Proficiency (%)	Problem Analysis
CLR-4 :	Utilize custom designed functions that can be used to perform tasks and can be repeatedly used in any application	Expected Attainment (%)	Design & Development
CLR-5 :	Create storage constructs using structure and unions. Create and Utilize files to store and retrieve information		Analysis, Design, Research
CLR-6 :	Create a logical mindset to solve various engineering applications using programming constructs in C		Modern Tool Usage
			Society & Culture
			Environment & Sustainability
			Ethics
			Individual & Team Work
			Communication
			Project Mgt. & Finance
			Life Long Learning
			PSO - 1
			PSO - 2
			PSO - 3
Course Learning Outcomes (CLO):	At the end of this course, learners will be able to:		
CLO-1 :	Identify methods to solve a problem through computer programming. List the basic data types and variables in C	2 85 80	L H H H H - - M M L - H - - -
CLO-2 :	Apply the logic operators and expressions. Use loop constructs and recursion. Use array to store and retrieve data	3 85 80	L H H H H - - M M L - H - - -
CLO-3 :	Analyze programs that need storage and form single and multi-dimensional arrays. Use preprocessor constructs in C	3 85 80	L H H H H - - M M L - H - - -
CLO-4 :	Create user defined functions for mathematical and other logical operations. Use pointer to address memory and data	3 85 80	L H H H H - - M M L - H - - -
CLO-5 :	Create structures and unions to represent data constructs. Use files to store and retrieve data	3 85 80	L H H H H - - M M L - H - - -
CLO-6 :	Apply programming concepts to solve problems. Learn about how C programming can be effectively used for solutions	3 85 80	L H H H H - - M M L - H - - -

Duration (hour)	21	21	21	21	21
S-1	SLO-1 Evolution of Programming & Languages	Relational and logical Operators	Initializing and Accessing 2D Array	Passing Array Element to Function	Initializing Structure, Declaring structure variable
	SLO-2 Problem solving through programming	Condition Operators, Operator Precedence	Initializing Multidimensional Array	Formal and Actual Parameters	Structure using typedef, Accessing members
S-2	SLO-1 Creating algorithms	Expressions with pre / post increment operator	Array Programs – 2D	Advantages of using Functions	Nested structure
	SLO-2 Drawing flowcharts	Expression with conditional and assignment operators	Array Contiguous Memory	Processor Directives and #define Directives	Accessing elements in a structure array
S-3	SLO-1 Writing pseudocode	If statement in expression	Array Advantages and Limitations	Nested Preprocessor Macro	Array of structure
	SLO-2 Evolution of C language, its usage history	L value and R value in expression	Array construction for real-time application	Advantages of using Functions	Accessing elements in a structure array
S-4	SLO-1 Lab 1: Algorithm, Flow Chart, Pseudocode	Lab 4: Operators and Expressions	Common Programming errors	Pointers and address operator	Passing Array of structure to function
	SLO-2			Size of Pointer Variable and Pointer Operator	Array of pointers to structures
S-8	SLO-1 Input and output functions: Printf and scanf	Control Statements – if and else	String Basics	Pointer Declaration and dereferencing pointers	Lab 13: Structures & Unions
	SLO-2 Variables and identifiers	else if and nested if, switch case	String Declaration and Initialization	Void Pointers and size of Void Pointers	Bit Manipulation to structure and Pointer to structure
S-9	SLO-1 Expressions	Iterations, Conditional and Unconditional branching	String Functions: gets(), puts(), getchar(), putchar(), printf()	Arithmetic Operations	Union Basic and declaration
	SLO-2 Single line and multiline comments	For loop	String Functions: atoi, strlen, strcat, strcmp		Accessing Union Members Pointers to Union
S-10	SLO-1 Constants, Keywords	While loop	String Functions: sprintf, sscanf, strcmp, strcpy, strstr, strtok		Dynamic memory allocation, malloc, realloc, free
					Allocating Dynamic Array

	SLO-2	Values, Names, Scope, Binding, Storage Classes	do while, goto, break, continue	Arithmetic Characters on Strings	Incrementing Pointers	Multidimensional array using dynamic memory allocation.
S 11-14	SLO-1	Lab 2: Input and Output Statements	Lab 5: Control Statements	Lab 8: Strings	Lab 11: Pointers	Lab 14: Structures & Unions
	SLO-2					
S-15	SLO-1	Numeric Data types: integer	Array Basic and Types	Functions declaration and definition	Constant Pointers	file: opening, defining, closing, File Modes, File Types
	SLO-2	Numeric Data types: floating point	Array Initialization and Declaration	Types: Call by Value, Call by Reference	Pointers to array elements and strings	Writing contents into a file
S-16	SLO-1	Non-Numeric Data types: char and string	Initialization: one Dimensional Array	Function with and without Arguments and no Return Values	Function Pointers	Reading file contents
	SLO-2	Increment and decrement operator	Accessing, Indexing one Dimensional Array Operations	Function with and without Arguments and Return Values	Array of Function Pointers	Appending an existing file
S-17	SLO-1	Comma, Arrow and Assignment operator	One Dimensional Array operations	Passing Array to Functions with return type	Accessing Array of Function Pointers	File permissions and rights
	SLO-2	Bitwise and Sizeof operator	Array Programs – 1D	Recursion Functions	Null Pointers	Changing permissions and rights
S 18-21	SLO-1	Lab 3: Data Types	Lab 6: Arrays – One Dimensional	Lab 9: Functions	Lab 12: Pointers	Lab 15: File Handling
	SLO-2					

Learning Resources	1. Zed A Shaw, <i>Learn C the Hard Way: Practical Exercises on the Computational Subjects You Keep Avoiding (Like C)</i> , Addison Wesley, 2015 2. W. Kernighan, Dennis M. Ritchie, <i>The C Programming Language</i> , 2 nd ed. Prentice Hall, 1996	3. Bharat Kinariwala, <i>Tep Dobry, Programming in C</i> , eBook 4. http://www.c4learn.com/learn-c-programming-language/
---------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Learning Assessment											
	Bloom's Level of Thinking	Continuous Learning Assessment (50% weightage)								Final Examination (50% weightage)	
		CLA – 1 (10%)		CLA – 2 (15%)		CLA – 3 (15%)		CLA – 4 (10%)#			
		Theory	Practice	Theory	Practice	Theory	Practice	Theory	Practice	Theory	Practice
Level 1	Remember Understand	20%	20%	15%	15%	15%	15%	15%	15%	15%	15%
Level 2	Apply Analyze	20%	20%	20%	20%	20%	20%	20%	20%	20%	20%
Level 3	Evaluate Create	10%	10%	15%	15%	15%	15%	15%	15%	15%	15%
	Total	100 %		100 %		100 %		100 %		100 %	

CLA – 4 can be from any combination of these: Assignments, Seminars, Tech Talks, Mini-Projects, Case-Studies, Self-Study, MOOCs, Certifications, Conf. Paper etc.,

Course Designers		
Experts from Industry	Experts from Higher Technical Institutions	Internal Experts
1. Dr. Sainarayanan Gopalakrishnan, HCL Technologies, sai.jgk@gmail.com	1. Prof. Janakiram D, IIT Madras, djam@iitm.ac.in	1. Dr. Christhu Raj M R, SRMIST
2. Dr. Sritharan Srinivasan, Wipro Technologies, sritharanms@gmail.com	2. Dr. Rajeev Sukumaran, IIT Madras, rajeev@wmail.iitm.ac.in	2. Dr. B. Amutha, SRMIST