

Course Code	USA20101J	Course Name	PROGRAMMING FOR PROBLEM SOLVING	Course Category	C	Professional Core Course			
						L	T	P	C
						4	0	4	6

Pre-requisite Courses	Nil	Co-requisite Courses	Nil	Progressive Courses	Nil
Course Offering Department	Computer Applications	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 logically	1	2	3
CLR-2 :	Write application code for specific purpose	Level of Thinking (Bloom)	Expected Proficiency (%)	Expected Attainment (%)
CLR-3 :	Understand the effectiveness of programming			
CLR-4 :	Customizing functions and procedures to encourage reusability			
CLR-5 :	Establish interaction between stored files and the application code			
CLR-6 :	Solve mathematical, scientific and engineering problems with reduced complexity			
Course Learning Outcomes (CLO):		At the end of this course, learners will be able to:		
CLO-1 :	Apply the features of programming language	2	85	80
CLO-2 :	Choose operators, control structures to solve the problem optimally	3	85	80
CLO-3 :	Analyse the problem thoroughly and choose the prebuilt functions/ customize functions to solve the problem	3	85	80
CLO-4 :	Able to use dynamic memory allocation concepts for problems that demand	3	85	80
CLO-5 :	Defend the need for files storage and the access previledge modes	3	85	80
CLO-6 :	Talk on the data flow	3	85	80

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Fundamental Knowledge	Application of Concepts	Link with Related Disciplines	Procedural Knowledge	Skills in Specialization	Ability to Utilize Knowledge	Skills in Modeling	Analyze, Interpret Data	Investigative Skills	Problem Solving Skills	Communication Skills	Analytical Skills	ICT Skills	Professional Behavior	Life Long Learning
H	H	H	H	H	H	-	M	M	L	-	H	-	M	H
L	H	H	H	H	H	-	M	M	L	-	H	-	M	H
L	H	H	H	H	H	-	M	M	L	-	H	-	M	H
L	H	H	H	H	H	-	M	M	L	-	H	-	M	H
L	H	H	H	H	H	-	M	M	L	-	H	-	M	H

Duration (hour)	24	24	24	24	24
S-1	SLO-1 Evolution of Programming Languages	Relational and logical Operators	Understanding contiguous memory allocation	Formal and Actual Parameters	File Types: text and binary
	SLO-2 Problem solving through programming	Character and Numbers: Manipulation	Array : Advantages and Limitations	Functions: Returning values	File operations:basics
S-2	SLO-1 Writing algorithms/pseudo codes	Expressions with pre / post increment operator	String Basics	Advantages of using Functions	File permissions and access privileges
	SLO-2 Drawing flowcharts	Expression with conditional and assignment operators	String Declaration and Initialization	Passing Array to Function	Changing permissions
S-3	SLO-1 Evolution of C language	Ternary operator	Understanding String Functions: gets(), puts(), getchar(), putchar(), printf()	Call by Value	Writing contents to file
	SLO-2 Program structure	L value and Rvalue in expression	String Functions: atoi, strlen, strcat, strcmp	Call by Reference (An introduction on pointers shall be effective)	Reading file contents
S-4	SLO-1 Need for file header files	Operator precedence	String Functions: sprintf, sscanf, strtok, strcpy, strstr, strtok	Nested functions	Appending an existing file
	SLO-2 Need for linkers and loaders	Type conversion	Need for tokenization	Functions: advantages and	Difference: Append and write



					limitations	
S 5-8	SLO-1 SLO-2	Lab 1: Algorithm, Flow Chart, Pseudo code	Lab 4: Operators and Expressions	Lab 7: Arrays : Multi dimensional	Lab 10: Functions	Lab 13: File: reading and writing
S-9	SLO-1 SLO-2	Input and output statements: scanf,printf	Control Statements : sequential, branching, looping and jump	Need for user-defined data types	Pointers and address operator	fscanf(),fprintf()
		Variables and identifiers	If, if ..else, else if ladder	Stuctures	sizeof Pointer Variable and Pointer Operator	fseek(),ftell()
S-10	SLO-1 SLO-2	Expressions	nested if, switch case	Unions	Pointer Declaration and dereferencing pointers	fputc(),fgetc()
		Single line and multiline comments	for loop	Accessing members of the structure	void Pointers and sizeof void Pointers	fputs(),fgets()
S-11	SLO-1 SLO-2	Constants, Keywords	while loop	Accessing members of the structure	Function and call by reference	fputw(),fgetw()
		Literals	do while	Structure and arrays	Functions and Returning array(use of pointers)	End_of_file in file handling
S-12	SLO-1 SLO-2	Scope and lifetime of variables	goto, break, continue, exit: Jump statements	Structure and arrays	Structures and pointers :dynamic creation of data structures(list)	feof(), remove()
		Storage clauses	Understanding jump statements with branch and iterative statements	Nested structures	Incrementing Pointers	ferror()
S 13-16	SLO-1 SLO-2	Lab 2: Input and Output Statements	Lab 5: Control Statements	Lab 8: Strings, structures and union	Lab 11: Pointers	Lab 14: File Handling fputw(),fgetw(), remove();
S-17	SLO-1 SLO-2	Data types classification:Basic,derived,user-defined	Array Basic	Functions declaration and definition	Constant Pointers	Processor Directives
		Numeric Data types: int, float, long, double	Array Declaration, Initialization	Prebuilt and user defined functions	Pointers and strings	Include
S-18	SLO-1 SLO-2	Non-Numeric Data types: char and string	Types	Function prototypes	Function Pointers	Predefined macros and macros
		Arithmetic operators	Manipulating one dimensional arrays with indices	Defining and calling functions	Array of Function Pointers	
S-19	SLO-1 SLO-2	Increment and decrement operator	Methods: sort, append, reverse, traverse	Multiple functions	Null Pointers	conditional compilation
		Bitwise and sizeof operator	Manipulating two dimensional arrays with indices	Recursion , recursive Functions	Using sizeof(),malloc,calloc()	#pragma
S-20	SLO-1 SLO-2	Using Boolean	Problems: matrix manipulations	Scope of variables across functions	File Handling	Creating include and macros
		Comma, Arrow and Assignment operator	Manipulating more than two dimensions in arrays	Sharing Global variables	Open(),close()	
S 21-24	SLO-1 SLO-2	Lab 3: Data Types	Lab 6: Arrays – One Dimensional	Lab 9: Functions	Lab 12: Pointers	Lab 15: Creating Macros

Learning Resources	1. Zed A Shaw, (2015), "Learn C the Hard Way: Practical Exercises on the Computational Subjects You Keep Avoiding (Like C)", Addison Wesley 2. W. Kernighan, Dennis M. Ritchie, (1996), "The C Programming Language", 2 <sup>nd</sup> Edition. PrenticeHall of India	3. ebook: Bharat Kinariwala, TepDobry, Programming in C, 4. <a href="http://www.c4learn.com/learn-c-programming-language/">http://www.c4learn.com/learn-c-programming-language/</a>
--------------------	---	--



Learning Assessment											
	Bloom's Level of Thinking	Continuous Learning Assessment (50% weightage)								Final Examination (50% weightage)	
		CLA – 1 (10%)		CLA – 2 (10%)		CLA – 3 (20%)		CLA – 4 (10%)#			
		Theory	Practice	Theory	Practice	Theory	Practice	Theory	Practice	Theory	Practice
Level 1	Remember	20%	20%	15%	15%	15%	15%	15%	15%	15%	15%
	Understand										
Level 2	Apply	20%	20%	20%	20%	20%	20%	20%	20%	20%	20%
	Analyze										
Level 3	Evaluate	10%	10%	15%	15%	15%	15%	15%	15%	15%	15%
	Create										
	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
Mr.G.Muruganandam, Group Project Manager, HCL Technologies, Chennai	Dr.S.Gopinathan, Professor, University of Madras, Chennai	Mrs. S. Usha, SRM IST
Mr.M. Hemachandar, Tech Lead, Wipro Limited, Chennai		Dr. P.J.Arul Leena Rose, SRM IST