Course Code	18CSC208L	Course Name	COMPETITIVE PROFESSIONAL SKILLS – I	Course Category	C	Professional Core	L	T	P	C
							0	0	2	1

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:*

CLR-1 :	*Understand importance of mathematics and problem solving approaches for programming.*
CLR-2 :	*Understand importance of optimized solutions for problems solving and its relevance to industry.*
CLR-3 :	*Implement mathematical and logical understanding approaches to implement test driven development practices.*
CLR-4 :	*Start participating in global coding competitions relevant to the syllabus.*

Course Learning Outcomes (CLO): *At the end of this course, learners will be able to:*

		Learning			Program Learning Outcomes (PLO)														
		1	2	3	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
		Level of Thinking (Bloom)	Expected Proficiency (%)	Expected Attainment (%)	Engineering Knowledge	Problem Analysis	Design & Development	Analysis, Design, Research	Modern Tool Usage	Society & Culture	Environment & Sustainability	Ethics	Individual & Team Work	Communication	Project Mgt. & Finance	Life Long Learning	PSO - 1	PSO - 2	PSO – 3
CLO-1 :	*Able to understand test and development aspects of programming by solving problems at Industry standards.*	*2*	*85*	*80*	*L*	*H*	*H*	*H*	*H*	-	-	*M*	*M*	*L*	-	*H*	-	-	-
CLO-2 :	*Able to interpret any given problem using required domain skills, mathematics.*	*3*	*85*	*80*	*L*	*H*	*H*	*H*	*H*	-	-	*M*	*M*	*L*	-	*H*	-	-	-
CLO-3 :	*Able to learn applicable methods to optimize solutions for any given problem.*	*3*	*85*	*80*	*L*	*H*	*H*	*H*	*H*	-	-	*M*	*M*	*L*	-	*H*	-	-	-
CLO-4 :	*Able to develop programs using C language until elementary data structures with test driven development.*	*3*	*85*	*80*	*L*	*H*	*H*	*H*	*H*	-	-	*M*	*M*	*L*	-	*H*	-	-	-

Duration (hour)		6	6	6	6	6
S-1	**SLO-1**	*Introduction to coding, datatypes and I/O Statements, Expression Evaluation, Arithmetic Operations, Assignment Operations, Relational Operations, Logical Operations,*	*Introduction to Linear data, Subscript of an array, Representing the array data, Insert values into an array, Print the values of an array, print the values of an array in reverse, find an element in an array,*	*Introduction to Modular Programming, Function Terminology, Inter Function communication, call-by-value and call-by-reference,*	*Introduction to user defined data, structures, array within structure, array of structures,*	*Introduction to tuple, accessing tuples, tuple operations,*
	SLO-2	*Bitwise Operations, Ternary Operations, Increment Operations, Decrement Operations, Special Operators usage, Example Problems*	*Find the Max element in an array, Find the min element in an array, Print the sum of the elements of an array, Print the sum of positive elements of an array*	*passing an array, returning a pointer, Dangling pointing & Memory leak, Global Vs. Local data space, Storage classes*	*nested structures, structure padding, bit-fields, union, enumeration*	*introduction to dictionaries, accessing values in dictionaries, properties and functions*
S-2	**SLO-1** **SLO-2**	***Lab 1:****Coding on expression evaluations, understanding precedence and associativity*	***Lab 4:****Basic list data problems, time efficient and classical problems on arrays.*	***Lab 7:****Coding programs using functions*	***Lab 10:****Coding problems including problems on implementation of user-defined data types*	***Lab 13:****Coding problems implementing tuples*
S-3	**SLO-1**	*Control Structures, Branching, If statement, If-Else statement, Else-If Ladder, Nested If, Loops, While Statement, Nested while statement, do while statement,*	*Matrix Representation* *Introduction to 2D Array, 2D Array Subscript,*	*Introduction to Recursion, Recursive nature, Recursion evaluation methods,*	*Introduction to Python, Basic syntax, variables and data types, operators, Input and Output, conditional statements and loops,*	*Introduction to modules, importing modules, math module, random module,*
	SLO-2	*For statement, nested for statement, Switch-case statement, Branching Un-Conditional, goto statement, break statement, continue statement, return statement.*	*RMO & CMO Representation, Matrix Problems.*	*Head and Tail recursion, Iteration Vs Recursion*	*accessing strings, string operations, string slices, functions and methods,*	*packages and composition*
S-4	**SLO-1** **SLO-2**	***Lab 2:****Programs include coding for Control structure evaluations*	***Lab 5:****Classical problems on matrix data, Matrix rotations, and display patterns*	***Lab 8****: Coding programs using functions and recursions, finding factorial/Fibonacci series etc.*	***Lab 11****: Problem solving on display patterns, series, strings and matrix using python*	***Lab 14:****Problem solving implementing math and random modules and packages using python*
S-5	**SLO-1**	*Time Complexity Analysis* *Introduction to Time Complexities, Analyzing the code, Consecutive Statements, Conditional Statements.*	*Introduction to Pointers, Pointer Variable, Pointer Arithmetic, Pointer to an array, Pointer to a String, Memory Layout, Runtime memory allocation, Stack memory Vs Heap memory,*	*Recursion Analysis, forming a recurrence relation, Evaluating a recurrence relation,*	*Introduction to lists, accessing list,*	*Introduction to exceptions, exception handling, except clause,*
	SLO-2	*Loop Statements, Square root Complexities, Logarithmic Complexities, Exponential Complexities, Examples*	*Array Vs Pointer Array, Array Vs Pointer, Introduction to String Data, User defined string handling methods, String handling functions.*	*Time Analysis, Pseudocodes, Example exercises.*	*Working on Lists*	*try? finally clause, user defined exceptions*
S-6	**SLO-1**	***Lab 3:****Coding for Generating Patterns,*	***Lab 6:****Coding problems on strings and*	***Lab 9****: Coding problems on matrix data,*	***Lab 12:****Problems using Lists*	***Lab 15:****Implementation of exception*

	SLO-2	*Number series*	*pointer to strings*	*strings using functions*		*handling using python*

Learning Resources	*1. Problem solving with C++ -9e- Walter Savitch – Pearson, 2018* *2. Programming in Python 3, A complete introduction to Python language - 2e - Mark Summerfield – Addison-Wiley, 2009* *3. Guide to Competitive Programming: Learning and Improving Algorithms Through Contests by Antti Laaksonen - Springer; 1st ed. 2017 edition , 2018*	

Learning Assessment

	Bloom's Level of Thinking	Continuous Learning Assessment (100% weightage)								Final Examination	
		CLA – 1 (15%)		CLA – 2 (15%)		CLA – 3 (50%)		CLA – 4 (20%)#			
		Theory	Practice	Theory	Practice	Theory	Practice	Theory	Practice	Theory	Practice
Level 1	Remember	-	*40%*	-	*30%*	-	*30%*	-	*30%*	-	-
	Understand										
Level 2	Apply	-	*40%*	-	*40%*	-	*40%*	-	*40%*	-	-
	Analyze										
Level 3	Evaluate	-	*20%*	-	*30%*	-	*30%*	-	*30%*	-	-
	Create										
	Total	100 %		100 %		100 %		100 %		-	

CLA – 4 will be weekly Assignments

Course Designers

Experts from Industry	Experts from Higher Technical Institutions	Internal Experts
Experts from Campus Corporate Connect		