A) Course Code **: 2418101(T2418101/P2418101/S2418101)**

B) Course Title **:** Programming with C (CSE, AIML)

C) Pre- requisite Course(s) **:**

D) Rationale **:**

In order to write programs to cater with various IT solutions, software developer needs to build logic, develop algorithms and flow charts and then apply coding in a suitable programming language.
This course is designed keeping in view developing these skills in students with the 'C' programing language. The 'C' has been widely used as a general-purpose language to develop basic and advanced applications. The course is basically designed to create a base to develop foundation skills of programming language.
This course deals with fundamental syntactics of 'C' that will help the students to apply the basic concepts, program structure and principles of 'C' programming paradigm to build given application.

E) Course Outcomes (COs): After the completion of the course, teachers are expected to ensure the accomplishment of following course outcomes by the learners. For this, the learners are expected to perform various activities related to three learning domains (Cognitive, Psychomotor and Affective) in classroom/ laboratory/ workshop/ field/ industry.

After completion of the course, the students will be able to-

CO-1 Develop flowchart and algorithm to solve problems logically.
CO-2 Write simple 'C' programs using arithmetic expressions.
CO-3 Develop 'C' programs using control structure.
CO-4 Develop 'C' programs using arrays and structures.
CO-5 Develop/Use functions in C programs for modular programming approach.
CO-6 Develop 'C' programs using pointers.

F) Suggested Course Articulation Matrix (CAM):

Course Outcomes (COs)	Programme Outcomes (POs)							Programme Specific Outcomes* (PSOs)	
	PO-1 Basic and Discipline Specific Knowledge	**PO-2** Problem Analysis	**PO-3** Design/ Development of Solutions	**PO-4** Engineering Tools	**PO-5** Engineering Practices for Society, Sustainability and Environment	**PO-6** Project Management	**PO-7** Life Long Learning	**PSO-1**	**PSO-2**
CO-1	1	2	1	-	-	-	-		
CO-2	1	1	1	1	-	-	-		
CO-3	1	2	1	1	-	-	-		
CO-4	1	2	1	1	-	-	-		
CO-5	1	2	2	1	-	-	-		
CO-6	1	2	3	1	-	-	-		

Legend: High (3), Medium (2), Low (1) and No mapping (-)

***** PSOs will be developed by the respective program coordinator at the institute level. As per the latest NBA guidelines, formulating PSOs is optional

G) Teaching and Learning Scheme:

Course Code	Course Title	Scheme of Study (Hours/Week)					
		Classroom Instruction (CI)		Lab Instruction (LI)	Notional Hours (TW+ SL)	Total Hours (CI+LI+TW+SL)	Total Credits (C)
		L	T				
2418101	Programming with 'C'	03	-	04	02	09	06

Legend:

CI: Classroom Instruction (Includes different instructional/implementation strategies i.e. Lecture (L), Tutorial (T), Case method, Demonstrations, Video demonstration, Problem based learning etc. to deliver theoretical concepts)

LI: Laboratory Instruction (Includes experiments/practical performances /problem-based experiences in laboratory, workshop, field or other locations using different instructional/Implementation strategies)

Notional Hours: Hours of engagement by learners, other than the contact hours for ensuring learning.

TW: Term Work (includes assignments, seminars, micro projects, industrial visits, any other student activities etc.)

SL: Self Learning, MOOCs, spoken tutorials, online educational resources etc.

C: Credits = (1 x CI hours) + (0.5 x LI hours) + (0.5 x Notional hours)

Note: TW and SL have to be planned by the teacher and performed by the learner under the continuous guidance and feedback of teacher to ensure outcome of learning.

H) Assessment Scheme:

Course Code	Course Title	Assessment Scheme (Marks)						Total Marks (TA+TWA+LA)
		Theory Assessment (TA)		Term Work & Self Learning Assessment (TWA)		Lab Assessment (LA)		
		Progressive Theory Assessment (PTA)	End Theory Assessment (ETA)	Internal	External	Progressive Lab Assessment (PLA)	End Laboratory Assessment (ELA)	
2418101	Programming with 'C'	30	70	20	30	20	30	200

Legend:

PTA: Progressive Theory Assessment in class room (includes class test, mid-term test and quiz using online/offline modes)

PLA: Progressive Laboratory Assessment (includes process and product assessment using rating Scales and rubrics)

TWA: Term work & Self Learning Assessment (Includes assessment related to student performance in assignments, seminars, micro projects, industrial visits, self-learning, any other student activities etc.

Note:

- ETA & ELA are to be carried out at the end of the term/ semester.
- Term Work is to be done by the students under the guidance of internal faculty but its assessment will be done **internally (40%)** as well as **externally (60%)**. Assessment related to planning and execution of Term Work activities like assignment, micro project, seminar and self-learning is to be done by internal faculty (Internal Assessment) whereas assessment of output/product/ presentation related to these activities will be carried out by external faculty/expert (External Assessment). However, criteria of internal as well as external assessment may vary as per the requirement of respective course. For valid and reliable assessment, the internal faculty should prepare checklist & rubrics for these activities.

I) Course Curriculum Detailing: This course curriculum detailing depicts learning outcomes at course level and session level and their attainment by the students through Classroom Instruction (CI), Laboratory Instruction (LI), Term Work (TW) and Self Learning (SL). Students are expected to demonstrate the attainment of Theory Session Outcomes (TSOs) and Lab Session Outcomes (LSOs) leading to attainment of Course Outcomes (COs) upon the completion of the course. While curriculum detailing, NEP 2020 related reforms like Green skills, Sustainability, Multidisciplinary aspects, Society connect, Indian Knowledge System (IKS) and others must be integrated appropriately.

J) Theory Session Outcomes (TSOs) and Units: T2418101

Major Theory Session Outcomes (TSOs)	Units	Relevant COs Number(s)
TSO 1a. Write Pseudo program logic for the given problem. TSO 1b. Identify the given symbols of a flow chart. TSO 1c. Explain guidelines for preparing flowchart with example. TSO 1d. Create flowchart to logically solve the given problem.	**Unit-1.0 Program Logic Development** **1.1 Fundamentals of Algorithms:** Notion of an algorithm. Pseudo-code conventions like assignment statements and basic control structures. **1.2 Algorithmic Problems:** Develop fundamental algorithms to solve simple problems such as: (i) solve simple arithmetic expression (ii) find the greatest of three numbers (iii) determine whether a given number is even or odd (iv) determine whether a given number is prime. **1.3 Flowchart:** Flowchart, Symbols of flowchart, Guidelines for preparing Flowchart	**CO1**
TSO 2a. Identify the given building block of a C program. TSO 2b. Write simple 'C' program using the given arithmetic expressions TSO 2c. Write a simple 'C' Program demonstrating the given data type conversion TSO 2d. Write I/O Statements for the given data.	**Unit-2.0 Basics of C Programming** **2.1 Introduction to C**: History of 'C' General Structure of a 'C' program: Header files, 'main' function. **2.2 Data Concepts:** Character set, tokens, keywords, Identifiers, Variables, Constant, data types, C operators, Arithmetic operators, Arithmetic expression, declaring variables, and data type conversion. **2.3 Basic Input Output:** Input and Output statements, using printf() and scanf(), character input/output statements, Input/output formatting, Use of comments	**CO2**
TSO 3a. Write a 'C' program using decision making structure for two-way branching to solve the given problem. TSO 3b. Write a 'C' program using decision making structure for multi-way branching to solve the given problem. TSO 3c. Write a 'C' program using loop statements to solve the given iterative problem. TSO 3d. Use appropriate statements to change the program flow in the given loop.	**Unit-3.0 Decision Making and Branching** **3.1 Decision Making and Branching**: Relational and logical operators, if statement, if else statement, nested if-else, if-else ladder', The switch statement **3.2 Looping:** while loop, do… while loop for loop, go to statement, Use of break and continue statements	**CO3**
TSO 4a. Write statements to read, write the given array. TSO 4b. Manipulate the given array of characters and numbers. TSO 4c. Use the structure for solving the given problem. TSO 4d. Write a sample program to demonstrate use of the given enumerated data type.	**Unit-4.0 Array and Structure** **4.1** Characteristics of an array, One dimension and two-dimension arrays, Array declaration and Initialization **4.2** Array of characters, Operation on array Character and String input/output **4.3** Introduction and Features of Structures, Declaration and Initialization of Structures Typedef, Enumerated Data Type, using structures in C Program	**CO4**
TSO 5a. Use the given Library function. TSO 5b. Develop user defined functions for the given problem. TSO 5c. Write 'C' codes to pass the given function parameters using "call by value" and "call by	**Unit-5.0 Concept and Need of Functions** **5.1** Library functions: Math functions, String handling functions, other miscellaneous functions.	**CO5**

Major Theory Session Outcomes (TSOs)	Units	Relevant COs Number(s)
reference" approach. TSO 5d. Write recursive function for the given problem.	**5.2** Writing User defined functions, scope of variables, Parameter passing: call by value, call by reference. **5.3** Recursive functions	
TSO 6a. Use pointers to access memory locations using pointer to solve the given problem. TSO 6b. Use pointers for performing the given arithmetic operation. TSO 6c. Develop a program to access elements of the given array using pointers. TSO 6d. Develop a program to access elements of the given structure using pointers.	**Unit-6.0 Pointers** **6.1** Concepts of pointers: declaring, initializing, accessing, Pointer arithmetic. **6.2** Handling arrays using pointers, Handling functions using pointers, Handling structures using pointers	**CO6**

Note: One major TSO may require more than one Theory session/Period.

K) Suggested Laboratory (Practical) Session Outcomes (LSOs) and List of Practical: P2418101

Practical/Lab Session Outcomes (LSOs)	S. No.	Laboratory Experiment/Practical Titles	Relevant COs Number(s)
LSO 1.1. Write, execute and debug simple 'C' program LSO 1.2. Write and execute simple 'C' program using variables, arithmetic expressions	1.	(a) Develop minimum 3 programs using Constants, Variables, arithmeticexpression. (b) Develop minimum 3 programs to exhibiting use of increment/decrement operators, data type conversion	CO1, CO2
LSO 2.1. Use scanf statement to take user input. LSO 2.2. Use printf statement to print the output.	2.	(a) Write simple program to convert temperature in Fahrenheit degrees to Centigrade degrees. (take input from the user) (b) Write simple programs to calculate the area and perimeter of the rectangle, and the area & circumference of the circle (take input from the user)	CO1, CO2 CO1, CO2
LSO 3.1. Write C Program using Decision Making and two-way branching statements.	3.	Write program to: (a) Determine whether a given year is a leap year or not. (b) Determine whether a string is palindrome. (c) Find the greatest of the three numbers using conditional operators (d) Find if a given character is vowel (use if-else ladder).	CO1, CO2, CO3
LSO 4.2. Write C Program using "switch-case" statement for multi-way branching. LSO 4.2. Use the "if" and "Switch" statements appropriately for decision making in C Program.	4.	Using switch statement- Write program to: (a) Print day of week by taking number from 1 to 7. (b) Print a student's grade ("A", "B", "C" etc.) by accepting his/her percent marks. (c) Find if a given character is vowel. (d) Using "if" and "switch" statements-	CO1, CO2, CO3

Practical/Lab Session Outcomes (LSOs)	S. No.	Laboratory Experiment/Practical Titles	Relevant COs Number(s)
		Write programs to check whether the triangle is equilateral, isosceles, or scalene triangle.	
LSO 5.1 Write and execute C programs using various types of loop statements to solve iterative problems.	5.	Write Program to: (a) Find sum of digits of a given number. (b) Generate multiplication table up to 10 for numbers 1 to 5. (c) Find Fibonacci series for given number. (d) Write a program to produce the following output: 1 2 3 4 5 6 7 8 9 10	CO1, CO2, CO3
LSO6.1 Write and execute C programs using one-dimension array. LSO 6.2 Write and execute C program using two-dimensional array.	6	Develop a Program to: (a) Sort list of 10 numbers. (b) Perform addition of 3x3 matrix.	CO1, CO2, CO3, CO4
LSO 7.1 Write and execute C program using Structures to solve given problem.	7	Develop Program to: (a) Create a structure called "library" to hold details of a book viz. accession number, title of the book, author name, price of the book, and flag indicating whether book is issued or not. Fetch some sample data and display the same. (b) Develop and execute C Program to Add Two Distances given in kilometer-meter Using Structures.	CO1, CO2, CO3, CO4
LSO8.1 Write C program using different types of library functions to solve given problem. LSO8.2 Write C program to Create and use user defined functions	8	Develop Program to demonstrate: (a) Use of all String handling functions. (b) Use of few Mathematical functions. (c) Use of few other miscellaneous functions. Develop a Program to: (a) Create a function to find GCD of given number. Call this function in a program. (b) Find Factorial of given number using recursion.	CO1, CO2, CO3, CO4, CO5
LSO 9.1 Write C program using pointers to manipulate the data by accessing the computer's memory	9	(a) Develop a Program to Print values of variables and their addresses. (b) Develop a Program to Find sum of all elements stored in given array using pointers.	CO1, CO2, CO3, CO4, CO6

L) **Suggested Term Work and Self Learning: S2418101** Some sample suggested assignments, micro project and other activities are mentioned here for reference.

a. **Assignments**: Questions/Problems/Numerical/Exercises to be provided by the course teacher in line with the targeted COs.

b. Micro Projects:

Use the features of Arrays and Structure appropriately for following problems:

1. Prepare sample mark sheet for 10 students.
2. Generate salary slips of employees in an organization.
3. Develop book issue system of library.
4. Design a basic calculator program that performs arithmetic operations like addition, subtraction, multiplication, and division based on user input.
5. Any other micro-projects suggested by subject faculty on similar line.
   (Use structure/ Text file and other features of 'C' to develop above listed applications)

c. Other Activities:

1. Seminar Topics: suggested few sample topics are given below. Teachers are suggested to find more topics for seminar presentations.
   - Application of pointers in C
   - C date/time functions

M) Suggested Course Evaluation Matrix: The course teacher has to decide and use appropriate assessment strategy and its weightage in theory, laboratory and Term work for ensuring CO attainment. The response/performance of each student in each of these designed activities is to be used to calculate **CO attainment**.

COs	Course Evaluation Matrix						
	Theory Assessment (TA)**		Term Work Assessment (TWA)			Lab Assessment (LA)#	
	Progressive Theory Assessment (PTA) Class/Mid Sem Test	End Theory Assessment (ETA)	Term Work & Self Learning Assessment			Progressive Lab Assessment (PLA)	End Laboratory Assessment (ELA)
			Assignments	Micro Projects	Other Activities*		
CO-1	15%	10%	15%	17%	17%	10%	17%
CO-2	10%	10%	10%	17%	17%	10%	17%
CO-3	20%	25%	15%	17%	17%	25%	17%
CO-4	20%	25%	20%	17%	17%	25%	17%
CO-5	20%	20%	20%	17%	17%	20%	17%
CO-6	15%	10%	20%	15%	15%	10%	15%
Total Marks	**30**	**70**	**20**	**20**	**10**	**20**	**30**
			50				

Legend:

***:** Other Activities include self- learning, seminar, visits, surveys, product development, software development etc.

******: Mentioned under point- (N)

#: Mentioned under point-(O)

Note:

- The percentage given are approximate
- In case of Micro Projects and End Laboratory Assessment (ELA), the achieved marks will be equally divided in all those COs mapped with total experiments.
- For CO attainment calculation indirect assessment tools like course exit survey need to be used which comprises of questions related to achievement of each COs.

N) **Suggested Specification Table for End Semester Theory Assessment:** Specification table represents the reflection of sample representation of assessment of cognitive domain of full course.

Unit Title and Number	Total Classroom Instruction (Cl) Hours	Relevant COs Number(s)	Total Marks	ETA (Marks)		
				Remember (R)	Understanding (U)	Application & above (A)
Unit-1.0 Program Logic Development	6	CO1	7	2	2	3
Unit-2.0 Basics of C programming	4	CO2	7	2	2	3
Unit-3.0 Control Structures	10	CO3	18	6	4	8
Unit-4.0 Array and Structure	10	CO4	17	4	2	11
Unit-5.0 Functions	10	CO5	14	4	2	8
Unit-6.0 Pointers	8	CO6	7	2	2	3
Total	**48**	**-**	**70**	**20**	**14**	**36**

Note: Similar table can also be used to design class/mid-term/ internal question paper for progressive assessment.

O) Suggested Assessment Table for Laboratory (Practical):

S. No.	Laboratory Practical Titles	Relevant COs Number(s)	PLA/ELA		
			Performance		Viva-Voce (%)
			PRA* (%)	PDA** (%)	
1.	(a) Develop minimum 3 programs using Constants, Variables, arithmeticexpression. (b) Develop minimum 3 programs to exhibiting use of increment/decrement operators, data type conversion	CO1, CO2	30	60	10
2.	(a) Write simple program to convert temperature in Fahrenheit degrees to Centigrade degrees. (take input from the user) (b) Write simple programs to calculate the area and perimeter of the rectangle, and the area & circumference of the circle (take input from the user)	CO1, CO2	40	50	10
3.	Write program to: (a) Determine whether a given year is a leap year or not. (b) Determine whether a string is palindrome. (c) Find the greatest of the three numbers using conditional operators (d) Find if a given character is vowel (use if-else ladder).	CO1, CO2, CO3	30	60	10
4.	Using switch statement- Write program to: (a) Print day of week by taking number from 1 to 7. (b) Print a student's grade ("A", "B", "C" etc.) by accepting his/her percent marks. (c) Find if a given character is vowel. (d) Using "if" and "switch" statements- Write programs to check whether the triangle is equilateral, isosceles, or scalene triangle.	CO1, CO2, CO3	30	60	10
5.	Write program to: (a) Find sum of digits of a given number.	CO1, CO2, CO3	30	60	10

S. No.	Laboratory Practical Titles	Relevant COs Number(s)	PLA/ELA		
			Performance		Viva-Voce (%)
			PRA* (%)	PDA** (%)	
	(b) Ge 1 o 10 for numbers 1 to 5. (c) Fi 2 3 nber. (d) W 4 5 6 llowing output: 7 8 9 10				
6.	Develop a Program to: (a) Sort list of 10 numbers. (b) Perform addition of 3x3 matrix.	CO1, CO2, CO3, CO4	30	60	10
7.	Develop Program to: (a) Create a structure called "library" to hold details of a book viz. accession number, title of the book, author name, price of the book, and flag indicating whether book is issued or not. Fetch some sample data and display the same. (b) Develop and execute C Program to Add Two Distances given in kilometer-meter Using Structures.	CO1, CO2, CO3, CO4	30	60	10
8.	Develop Program to demonstrate: (a) Use of all String handling functions. (b) Use of few Mathematical functions. (c) Use of few other miscellaneous functions. Develop a Program to: (a) Create a function to find GCD of given number. Call this function in a program. (b) Find Factorial of given number using recursion.	CO1, CO2, CO3, CO4, CO5	40	50	10
9.	(a) Develop a Program to Print values of variables and their addresses. (b) Develop a Program to Find sum of all elements stored in given array using pointers.	CO1, CO2, CO3, CO4, CO6	40	50	10

Legend:
PRA*: Process Assessment
PDA**: Product Assessment

Note: This table can be used for both end semester as well as progressive assessment of practical. Rubrics need to be prepared by the course teacher for each experiment/practical to assess the student performance.

P) Suggested Instructional/Implementation Strategies: Different Instructional/ ImplementationStrategies may be appropriately selected, as per the requirement of the content/outcome. Some of them are Improved Lecture, Tutorial, Case Method, Group Discussion, Industrial visits, Industrial Training, Portfolio Based Learning, Role Play, Live Demonstrations in Classrooms, Lab, Field, Information and Communications Technology (ICT) Based Teaching Learning, Blended or flipped mode, Brainstorming, Expert Sessions, Video Clippings, Use of Open Educational Resources (OER), MOOCs etc.

Q) List of Major Laboratory Equipment, Tools and Software:

S. No.	Name of Equipment, Tools and Software	Broad Specifications	Relevant Experiment/Practical Number
1.	Computer system	(Any computer system with basic configuration)	All
2.	'C' Compiler	Any C compiler for ANSI C	All

R) Suggested Learning Resources:

(a) Books:

S. No.	Titles	Author(s)	Publisher and Edition with ISBN
1.	Programming in ANSI C	Balagurusamy, E.	McGraw Hill Education, New Delhi 2019, ISBN-13: 978-9351343202 ISBN: 978-1259004612
2.	The C Programming Language	Brian, W. Kernighan, Ritchie Dennis	PHI Learning Private Limited, New Delhi 1990, ISBN13: 9789332549449, ISBN10: 9332549443
3.	Let us C	Kanetkar, Yashawant	BPB Publications, New Delhi 2020, ISBN-10: 9389845688 ISBN-13: 978-9389845686

(b) Online Educational Resources:

1. http://nptel.ac.in/courses/106105085/4
2. www.w3schools.com
3. www.programiz.com/c-programming
4. https://www.codecademy.com/courses/getting-started-v2/0/1
5. http://spoken-tutorial.org/

Note: Teachers are requested to check the creative commons license status/ financial implications of the suggested, online educational recourses before use by the students.

(c) Others:

-

******