Government of Karnataka
DEPARTMENT OF TECHNICAL EDUCATION

Program	ELECTRONICS AND COMMUNICATION	**Semester**	**III**
Course Name	**C-Programming**	**Type of Course**	Integrated
Course Code	25EC34I	**Contact Hours**	7 Hours/Week
Teaching Scheme	L: T:P :: 3:0:4	**Credits**	6 Credits
CIE Marks	**50**	**SEE Marks**	**50 (Practice)**

1. Rationale:

The C programming language serves as a fundamental tool for enhancing logical reasoning and problem-solving abilities. This course aims to build students confidence in programming skills while fostering a strong understanding of programming concepts and methodologies for addressing engineering challenges. The syllabus offers a solid foundation in programming principles, practical problem-solving and hands-on experience with both C and Arduino, effectively preparing diploma students for successful careers in electronics and communication.

2. Course Outcomes: At the end of the Course, the student will be able to:

CO-01	Implement C programming and algorithms to solve problems logically using loops, decision-making statements and functions.
CO-02	Apply arrays, structures, and pointers in C for data storage and management.
CO-03	Develop basic Arduino programs for sensor interfacing and control using the Arduino IDE.

3. Course Content

WEEK	CO	PO	Theory	Practice
1	**1**	**1, 4, 7**	**Overview of Programming Languages** 1. Need for Programming languages. 2. Types of programming languages-Low level, assembly and high-level languages. 3. Features of Procedure Oriented Programming (POP) and Object Oriented Programming (OOP).	1. Installation of C Programming Software (GCC, Code Blocks, Turbo C, Dev-C++). 2. Familiarization of C software.
2	**1**	**1,2, 3,4**	1. Flowcharts and Algorithms. 2. Development tools – Introduction to Assembler, Linker, Compiler, and	1. Write Flowchart and Algorithm for displaying the personal address and official address. 2. Write C program for printing

			Interpreter. 3. Structure of 'C' Program.	your mailing address. (File Management-Create a new folder and save the programs in the folder.)
3	**1**	**1, 2, 4**	1. Character set, Keywords, identifiers, Constants and variables. 2. Data types: Predefined (Integer, signed and unsigned, long, float, double, character). **3.** User defined datatypes (Arrays and structures).	1. Write C program to read and display the variables of different data types. 2. Write C program to input the radius of a circle as an integer, convert it to a float, and calculate the circumference**.**
4	**1**	**1, 2, 4**	**1.** Operators: Arithmetic, Logical, Relational, increment and decrement, bitwise, assignment operator. **2.** Special operators-unary, ternary operators. Precedence and associativity. **3.** Formatted input and output statements.	1. Write C program for addition of three numbers. 2. Write C program to find the biggest of three numbers using ternary operator.
5	**1**	**1,2, 3,4**	**Decision making statements** **1.** if-else, if-else-if statements with simple examples. **2.** nested if statement with simple examples. **3.** Switch statement with simple examples.	1. Write a C program to check whether the alphabet is vowel or not using switch statement. 2. Write a C program that checks if a number is even or odd using if-else.
6	**1**	**1,2, 3,4**	**Looping and Branching statements** 1. for statement with simple examples. 2. while, do-while statements with simple examples. **3.** break, continue statements with simple examples.	1. Write a C program to print first 20 even numbers using for loop. 2. Write a C program to find the sum of numbers from 1 to N using while loop
7	**1**	**1, 2, 3,4, 5,7**	**Functions** 1. Concept and Need of a Function. 2. Declaration, definition and Calling of function. **3.** Passing values between functions: Call by value and Call by reference.	1. Write a C program to implement swapping of two integers by using functions Call by value. 2. Write a C program to implement swapping of two integer by using functions call by reference.
8	**1**	**1, 2, 3,4, 5,7**	1. Library functions: Math function like: mod(), sqrt(), pow(),exp(), sum(), round(). 2. Character functions like islower(), isupper(), isdigit(), tolower(). **3.** Recursive functions-Factorial of a number.	1. Write a C program to find the roots of the quadratic equations using Math function (only for real roots). 2. Write a C program to implement the following character functions islower(), isupper(), isdigit(), tolower().
9		**1, 2,**	**Arrays** 1. Concept and need of arrays.	1. Write a C Program to store 10 numbers in an array and find the

	2	**3,4, 5,7**	2. Declaration, Initialization, Storing of Array Elements in Memory, Displaying array elements 3. Two-Dimensional Arrays: Initialization, Adding elements to 2-D array and display elements of 2-D array.	sum and average of 10 numbers. 2. Write a C Program to perform the following operation on matrix using 2-D array: Addition & Subtraction.
10	**2**	**1, 2, 3,4, 5,7**	**Strings** 1. Declaration, Initialization and display of string. 2. Standard Library String functions Strlen(), strcpy(), strcat(),strcmp(). 3. Simple example programs.	1. Write a C program to check if the string is Palindrome or not. 2. Write a C program to concatenate two strings and find the length of the concatenated string.
11	**2**	**1, 2, 3,4, 5,7**	**Structures & Pointer** 1. Introduction to structures, declaring a structure. 2. Accessing structure elements, displaying structure elements. 3. Concept of pointer: Pointer variables, Declaration of pointer.	1. Write a C program to store and display the student record using structure. 2. Write a simple C program for demonstration of & and * in pointers.
12	**3**	**1, 2, 3,4, 5,7**	**Introduction to Arduino** 1. Overview of Arduino and its applications and introduction to the Arduino IDE. 2. Arduino Programming Basics.Basic syntax and structure of Arduino code. **3.** Understanding functions: setup (), loop ().Variables, data types and operators.	1. Write an Arduino Program for blinking LED with different delays. 2. Write an Arduino Program to turn ON and OFF the buzzer.
13	**3**	**1, 2, 3,4, 5,7**	1. Introduction to sensors and actuators. 2. Working of IR sensor and ultrasonic sensor. 3. Working of a relay and buzzer.	1. Write an Arduino Program to interface IR sensor/Ultrasonic sensor for obstacle detection. Turn ON the buzzer when the obstacle is detected. 2. Write an Arduino program to read temperature using LM35 and display on the serial monitor. Also, turn ON the 5V fan using Relay when the temperature is above certain value.

Note:

1. In practice sessions all video demonstrations should be followed by MCQ/Quiz/ Subjective questions and evaluation has to be documented.

2. Online course completion certification to be done on relevant topics on Swayam/NPTEL/Infosys Springboard platforms or any other platform.

3. Problems statement to be collected from the relevant industries, resolve and submit it to

the course coordinator.

4. References:

1. C Programming Language" by Brian W. Kernighan and Dennis M. Ritchie
2. "Programming in ANSI C" by E. Balaguruswamy
3. "C: A Modern Approach" by K. N. King
4. "C Programming: A Problem-Solving Approach" by Forouzan, Gilberg
5. "C Programming Absolute Beginner's Guide" by Greg Perry and Dean Miller
6. "The C Programming Language (2nd Edition)" by Brian W. Kernighan and Dennis M. Ritchie
7. "C Programming in Easy Steps" by Mike McGrath
8. "C Programming: A Comprehensive Approach" by B. R. D. H. A. Dhananjay
9. "Arduino Projects for Beginners" by M. J. T. C. Anaya
10. "Arduino Project Handbook: 25 Practical Projects to Get You Started" by Mark Geddes
11. "Getting Started with Arduino" by Massimo Banzi

5. CIE Assessment Methodologies

Sl.No	CIE Assessment	Test Week	Duration (minutes)	Max marks	
1.	CIE-1 Theory Test	4	90	50	Average of all CIE=50 Marks
2.	CIE-2 Practice Test	7	180	50	
3	CIE-3 Theory Test	10	90	50	
4.	CIE-4 Practice Test	13	180	50	
5	CIE-5 Portfolio evaluation of all the activities through Rubrics	1-13		50	
				Total	**50 Marks**

Note: Portfolio evaluation includes average of (a) and (b)

a) Any one suggested activity with report /presentation and simulation evaluated for 50 Marks.

b) Each laboratory exercise will be evaluated for a total of 50 marks. The evaluation will include the following components:

(i) Written description of the experiment in the observation book.
(ii) Conducting the experiment and the associated learning outcomes.
(iii) The results obtained from the experiment.
(iv) Corrections and evaluations of the experiment completed in the previous class, documented in the record book.

6. SEE – Practice Assessment Methodologies

Sl. No	SEE – Practice Assessment	Duration (minutes)	Max marks	Min marks to pass
1.	Semester End Examination-Practice	180	50	20

7. CIE Theory Test model question paper

Program	**ELECTRONICS AND COMMUNICATION**			**Semester -III**	
Course Name	**C-Programming**			**Test**	**I/III**
Course Code	**25EC34I**	**Duration**	**90min**	**Marks**	**50**
Name of the Course Coordinator:					
Note: Answer any one full question from each section. Each full question carries equal marks.					

Q.No	**Questions**	**Cognit ive Level**	**Course Outcome**	**Mark s**
	Section - 1			
1	a) Compare low-level, assembly, and high-level languages with examples. b) Write a flowchart and algorithm to calculate the sum of two numbers. c) Discuss the roles of an assembler, linker, compiler, and interpreter in the software development process. Provide examples of each.	L2 L3 L2	CO1 CO1 CO1	5 10 10
2	a) Explain the primary characteristics of Object- Oriented Programming (OOP)? b) Write a flowchart and algorithm to illustrate the process of finding the largest of three numbers. c) Describe the structure of a C program and write a simple program that prints "Hello, World!"	L2 L3 L2	CO1 CO1 CO1	5 10 10
	Section - 2			
3	a) Develop a program to input the radius of a circle as an integer, convert it to a float, and calculate the circumference. b) Discuss the significance of variables and constants in C programming with examples. c) Interpret the concept of operator precedence and associativity with examples in C.	L3 L2 L2	CO1 CO1 CO1	5 10 10
4	a) Write a program to calculate the area and perimeter of a rectangle using arithmetic operators. b) Explain the concept of data types in C. Compare predefined and user-defined data types? c) Describe logical and relational operators in C.	L3 L2 L2	CO1 CO1 CO1	5 10 10

Note for the Course coordinator:

1. Each question may have one, two or three subdivisions. Optional questions in each section carry the same weightage of marks, cognitive level and course outcomes.
2. All questions must be framed under Understand (L2) & Apply (L3) cognitive level using Revised Bloom's Taxonomy.

Signature of the Course Coordinator | **Signature of the HOD** | **Signature of the IQAC Chairman**

8. CIE-2 Practice Test model question paper

Program	**ELECTRONICS AND COMMUNICATION**			**Semester**	**III**
Course Name	**C-PROGRAMMING**			**Test**	**II**
Course Code	**25EC34I**	**Duration**	**180 min**	**Marks**	**50**
Name of the Course Coordinator:					
Questions				**CO**	**Marks**
Give two 'C' programs for students to write and any one program for execution				**CO1**	**50**
Scheme of assessment: **a) Writing two programs** **b) Execution of any one program.** **c)Result** **d)Viva-voce**					 **20** **15** **05** **10**
			Total Marks		**50**

Signature of the Course Coordinator **Signature of the HOD**

9. **Suggestive Activities:**

The List is an Example and not inclusive of all possible activities of the course. Student and Faculty are encouraged to choose activities that are relevant to the topic.

Note: Activity can be undertaken by either an individual or a team comprising up to five students.

Sl.No.	Suggestive Activities for Tutorials
01	Write a C program that prints your mailing address using variables for different components.
02	Measure the distance to an object using an ultrasonic sensor and display the result on the Serial Monitor using Arduino.
03	Implement a number guessing game where the user has to guess a randomly generated number within a range.
04	Relay-Controlled Home Automation using Arduino.
05	Develop a calculator program that performs basic arithmetic operations
06	Implement a program that stores and displays student records (name, roll number, marks) using structures.
07	Create a simple ATM simulation program that allows users to check balance, deposit, and withdraw money.

10. Rubrics for Assessment of Activity (Qualitative Assessment)

Sl. No.	Dimension	Beginner	Intermediate	Good	Advanced	Expert	Students Score
		10	20	30	40	50	
1		Descriptor	Descriptor	Descriptor	Descriptor	Descriptor	40
2		Descriptor	Descriptor	Descriptor	Descriptor	Descriptor	30
3		Descriptor	Descriptor	Descriptor	Descriptor	Descriptor	50
4		Descriptor	Descriptor	Descriptor	Descriptor	Descriptor	20
	Average Marks=(40+30+50+20)/4=35						**35**

Note: Dimension and Descriptor shall be defined by the respective course coordinator as per the activities.

11. **CIE-4/SEE- Model Practice Question Paper**

Program	**ELECTRONICS AND COMMUNICATION**		**Semester**	**III**
Course Name	**C-PROGRAMMING**	**Course Code**	**25EC34I**	**180 min**
Questions			CO1,CO2,CO3	Marks
				50
Scheme of assessment: **PART A** **a) Writing two C programs** **b) Execution of any one program.** **c)Result** **PART B** **a) Writing arduino program** **b) Execution** **c) Result** **Viva-voce**				**10** **10** **05** **05** **05** **05** **10**
			Total Marks	**50**

Signature of the Examiner **Signature of the Examiner**

12. Equipment/software list with Specification for a batch of 30 students

Sl.No.	Particulars	Specification	Quantity
01	Computers	Dual core/i3/i5/i7/i9/i10 processor, 8 GB RAM, 1TB hard disk/SSD	30
02	Software Code::Blocks for C-Programming	-	-
03	Software Code Turbo C++ for C-Programming	-	-
04	Software Arduino IDE	-	-
05	Arduino board (e.g., Arduino UNO)		30 Each
06	Ultrasonic sensor , IR sensor module, relay, buzzer ,LM35 Temperature sensor		30 Each