COURSE CODE	COURSE NAME	L	T	P	CREDIT	YEAR OF INTRODUCTION
101009/IT100C	FUNDAMENTALS OF COMPUTER SCIENCE	3	0	0	3	2021

1. Preamble

The syllabus is prepared with the view of preparing the Engineering Graduates capable of writing readable C programs to solve computational problems that they may have to solve in their professional life. The course content is decided to cover the essential programming fundamentals which can be taught within the given slots in the curriculum.

2. Prerequisite

NIL

3. Syllabus

Module 1

General problem Solving concepts: Algorithm, and Flowchart for problem solving with Sequential Logic Structure, Decisions and Loops.
Imperative languages: Introduction to imperative language; syntax and constructs of a specific language (ANSI C).
Input and Output: Standard I/O, Formatted Output – printf, Formated Input – scanf, Variable length argument list.

Module 2

Types Operator and Expressions with discussion of variable naming and Hungarian Notation: Variable Names, Data Type and Sizes (Little Endian Big Endian), Constants, Declarations, Arithmetic Operators, Relational Operators, Logical Operators, Type Conversion, Increment Decrement Operators, Bitwise Operators, Assignment Operators and Expressions, Precedence and Order of Evaluation, proper variable naming and Hungarian Notation

Control Flow with discussion on structured and unstructured programming: Statements and Blocks, IfElse-If, Switch, Loops – while, do, for, break and continue, goto labels, structured and un- structured programming.

Module 3

Functions and Program Structure with discussion on standard library: Basics of functions, parameter passing and returning type, C main return as integer, External, Auto, Local, Static, Register Variables, Scope Rules, Block structure, Initialisation, Recursion, Pre-processor, Standard Library Functions and return types.

Module 4

Pointers and Arrays: Pointers and address, Pointers and Function Arguments, Pointers and Arrays, Address Arithmetic, character Pointers and Functions, Pointer Arrays, Pointer to Pointer, Multidimensional array and Row/column major formats, Initialisation of Pointer Arrays, Command line arguments, Pointer to functions, complicated declarations and how they are evaluated.

Module 5

Structures: Basic Structures, Structures and Functions, Array of structures, Pointer of structures, Self referral structures, Table look up, typedef, unions, Bit-fields
File access including FILE structure, fopen, stdin, sdtout and stderr, Error Handling including exit, perror and error.h, Line I/O, related miscellaneous functions.

Unix system Interface: File Descriptor, Low level I/O – read and write, open, create, close and unlink, Random access – lseek, Discussions on Listing Directory, Storage allocator.

Programming Method: Debugging, Macro, User Defined Header, User Defined Library Function, makefile utility.

4. Text Books

1. B. W. Kernighan and D. M. Ritchi, *The C Programming Language*, Second Edition PHI.

2. B. Gottfried, *Programming in C*, Second Edition, Schaum Outline Series.

5. Reference Books

1. Herbert Schildt, *C: The Complete Reference*,Fourth Edition, McGraw Hill.

2. Yashavant Kanetkar, *Let Us C*, BPB Publications.

6.Course Outcomes:

CO 1: Analyze a computational problem and develop an algorithm/flowchart to find its solution

CO 2: Develop readable C programs with branching and looping statements, which uses Arithmetic, Logical, Relational or Bitwise operators.

CO 3: Divide a given computational problem into a number of modules and develop a readable multi-function C program by using recursion if required, to find the solution to the computational problem

CO 4: Write readable C programs which use pointers for array processing and parameter passing

CO 5: Develop readable C programs with files for reading input and storing output and to Write readable C programs with structure for storing the data to be processed.

7.Mapping of course outcomes with program outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	3	2		1	-	-	-	1	1	2
CO2	2	2	2	1	1	-	-	-	-	1	-	2
CO3	2	2	2	1	2	-	-	-	-	1	-	2
CO4	3	3	3	2	3	-	-	-	-	1	1	2
CO5	3	2	-	-	2	-	-	-	-	1	-	2

8. Continuous Assessment

Learning Objectives	Continuous Internal Evaluation (CIE)		End Semester Examination (ESE out of 100)
	Internal Examination 1 (50)	Internal Examination 2 (50)	
Remember	15	10	25
Understand	10	15	25
Apply	20	20	40
Analyse	5	5	10
Evaluate			
Create			

9.Mark distribution

Total Marks	CIE Marks	ESE Marks	ESE Duration
150	50	100	3 Hours

9. Continuous Internal Evaluation Pattern:

Attendance: 10 marks

Continuous Assessment Test 1 (for theory, for 2 hrs) : 20 marks

Continuous Assessment Test 2 (for lab, internal examination, for 2 hrs) : 20 marks

10. Internal Examination Pattern:

There will be two parts; Part A and Part B. Part A contains 5 questions with 2 questions from each module (2.5 modules x 2 = 5), having 3 marks for each question. Students should answer all questions. Part B also contains 5 questions with 2 questions from each module (2.5 modules x 2 = 5), of which a student should answer any one. The questions should not have subdivisions and each one carries 7 marks.

11. End Semester Examination Pattern:

There will be two parts; Part A and Part B. Part A contains 10 questions with 2 questions from each module, having 3 marks for each question. Students should answer all questions. Part B contains 2 questions from each module of which a student should answer any one. Each question can have a maximum 2 subdivisions and carry 14 marks.

****