MAULANA ABUL KALAM AZAD UNIVERSITY OF TECHNOLOGY, WEST BENGAL
(Formerly West Bengal University of Technology)
Syllabus of BCA
(Effective from 2023-24 Academic Sessions)

SEMESTER: I

DEFINITION OF CREDIT

1 HR LECTURE PER WEEK	1 CREDIT
1 HR TUTORIAL PER WEEK	1CREDIT
2 HR PRACTICAL PER WEEK	1 CREDIT

SUBJECT NUMBERING SCHEME:

CODE FOR THE DEPT. OFFERING SUBJECT	SUBJECT TYPE	SEM	SUBJECT CODE

C	CORE MAJOR

SUBJECT NAME: Digital Electronics **CREDIT: 3L + 2P**

SUBJECT CODE: BCAC101

COURSE OBJECTIVE:

The objective of the course "Digital Electronics" is to provide students with a comprehensive understanding of the principles, theory, and practical applications of digital circuits and systems. Throughout the course, students will explore the foundational concepts of digital electronics, enabling them to design, analyze, and troubleshoot digital circuits commonly used in various electronic devices and systems.

COURSE OUTCOME	
CO1	To gain basic knowledge of digital electronics circuits and its levels.
CO2	To understand and examine the structure of various number system and its conversation.
CO3	To learn about the basic requirements for a design application
CO4	To enable the students to understand, analyze and design various combinational and sequential circuits

CO5	To understand the logic functions, circuits, truth table and Boolean algebra expression

DETAILED SYLLABUS:

Module No:	**NAME OF THE TOPIC**	**HOURS**	**MARKS**
M1	Number Systems & Codes Decimal Number, Binary Number, Octal Number, Hexadecimal Number, Conversion – Decimal to Binary, Binary to Decimal, Octal to Binary, Binary to Octal, Hexadecimal to Binary, Binary to Hexadecimal, Octal to Binary to Hexadecimal, Hexadecimal to Binary to Octal; Floating Point Number Representation, Conversion of Floating Point Numbers, Binary Arithmetic, 1's and 2's Complement, 9's and 10's Complement, Complement Arithmetic, BCD, BCD addition, BCD subtraction, Weighted Binary codes, Non-weighted codes, Parity checker and generator, Alphanumeric codes.	5	10
M2	Logic Gates : OR, AND, NOT, NAND, NOR, Exclusive – OR, Exclusive – NOR, Mixed logic.	2	10
M3	Boolean Algebra: Boolean Logic Operations, Basic Law of Boolean Algebra, Demorgan's Theorem, Principle of Duality.	6	10
M4	Minimization Techniques Sum of Products, Product of Sums, Karnaugh Map [up to 4 variables].	4	10
M5	Multilevel Gate Network Implementation of Multilevel Gate Network, Conversion to NAND-NAND and NOR-NOR Gate Networks.	2	5
M6	Arithmetic Circuits Half Adder, Full Adder, Half Subtractor, Full Subtractor, Carry Look Ahead Adder, 4-Bit Parallel Adder	5	5
M7	Combinational Circuits Basic 2-input and 4-input multiplexer, Demultiplexur, Basic binary decoder, BCD to binary converters, Binary to Gray code converters, Gray code to binary converters, Encoder.	5	5
M8	Sequential Circuits Introduction to sequential circuit, Latch, SR Flip Flop, D Flip Flop, T Flip Flop, JK Flip Flop, Master Slave Flip Flop	8	5

M9	Basics of Counters Asynchronous [Ripple or serial] counter, Synchronous [parallel] counter	4	5
M10	Basics of Registers SISO, SIPO, PISO, PIPO, Universal Registers	4	5
	sub total	45	70
	Internal examination	3	30
	TOTAL	48	100

PRACTICAL:
SUBJECT NAME: Digital Electronics Lab Credit: 2
SUBJECT CODE: BCAC191

List of Practical's: -

1. Realization of basic gates using Universal logic gates.
2. Code conversion circuits- BCD to Excess-3
and vice- versa.3 Four-bit parity generator and
comparator circuits.
4. Construction of simple Decoder and Multiplexer circuits using logic gates.
5. Design of combinational circuit for BCD to decimal conversion to drive 7-segment display using multiplexer.
6. Construction of simple arithmetic Circuits-Adder, Subtractor.
7. Realization of RS-JK and D flip-flops using Universal logic gates.
8. Realization of Universal Register using JK flip-flops and logic gates.
9. Realization of Universal Register using multiplexer and flip-flops.
10. Realization of Asynchronous Up/Down counter.
11. Realization of Synchronous Up/Down counter.
12. Realization of Ring counter and Johnson's counter.
13. Construction of adder circuit using Shift Register and full Adder.

SUGGESTED READING:

1. "Digital Design" by M. Morris Mano and Michael D. Ciletti Publisher: Pearson India Education Services Pvt. Ltd.
2. "Digital Fundamentals" by Thomas L. Floyd and R. David Maki Publisher: Pearson India Education Services Pvt. Ltd.
3. "Digital Electronics: Principles, Devices and Applications" by Anil K. Maini Publisher: John Wiley & Sons (Asia) Pte. Ltd.
4. "Digital Electronics: A Practical Approach" by William Kleitz Publisher: Pearson India Education Services Pvt. Ltd.
5. "Digital Logic Design" by Brian Holdsworth and Clive Woods Publisher: Pearson India Education Services Pvt. Ltd.

6. "Digital Electronics: Principles and Applications" by Roger L. Tokheim Publisher: McGraw-Hill Education (India) Pvt. Ltd.
7. "Fundamentals of Digital Logic with VHDL Design" by Stephen Brown and Zvonko Vranesic Publisher: McGraw-Hill Education (India) Pvt. Ltd.
8. "Digital Electronics: A Primer" by Michael J. Ciletti Publisher: Pearson India Education Services Pvt. Ltd.

SUBJECT NAME: Programming for Problem Solving through C Credit: 3L + 2P

SUBJECT CODE: BCAC102

COURSE OBJECTIVE:

The objective of the course "Programming for Problem Solving through C" is to equip students with fundamental programming skills using the C programming language and foster a problem-solving mindset. Throughout the course, students will develop a solid foundation in computer programming concepts and techniques, enabling them to tackle real-world problems and develop efficient, structured, and modular solutions.

COURSE OUTCOME	
CO1	Apply programming constructs of C language to solve the real world problem
CO2	To implement conditional branching, iteration and recursion
CO3	Explore user-defined data structures like arrays in implementing solutions to problems like searching and sorting
CO4	Explore user-defined data structures like structures, unions and pointers in implementing solutions
CO5	Create problem-solving solutions utilizing modular programming elements and functions.
CO6	Use files to store information after solving the problem related to the real world

DETAILED SYLLABUS:

Module No:	**NAME OF THE TOPIC**	**HOURS**	**MARKS**
M1	**Introduction to components of a computer system**: Memory, processor, I/O Devices, storage, operating system, Concept of assembler, compiler, interpreter, loader and linker. Representation of Algorithm, Flowchart, Pseudo code with examples, From algorithms to programs, source code. Compilation process of C program; .asm file, .obj file and .exe file. Number Systems: Binary, Octal, Decimal, Hexadecimal format.	4	5

M2	**Introduction to 'C' Language**: C Language - Background, C Programs, Identifiers, Data Types, Variables, Constants, Input / Output Statements Arithmetic Operators and Expressions: Evaluating Expressions, Precedence and Associativity of Operators, Type Conversions.	4	2
M3	**Conditional Statements and loops :** Decision making within a program, Conditions, Relational Operators, Logical Connectives, if statement, if-else statement, Loops: while loop, do while, for loop, Nested loops, Infinite loops, Switch statement, structured Programming	7	15
M4	**Arryays:** One dimensional arrays: Array manipulation; Searching, Insertion, Deletion of an element from an array; Finding the largest/smallest element in an array; Two dimensional arrays, Addition/Multiplication of two matrices, Transpose of a square matrix; Null terminated strings as array of characters, Standard library string functions	8	15
M5	**Pointers :**Address operators, pointer type declaration, pointer assignment, pointer initialization, pointer arithmetic, functions and pointers, Arrays and Pointers, pointer arrays, pointers and structures, dynamic memory allocation.	8	8
M6	**Functions:** Top-down approach of problem solving, Modular programming and functions, Standard Library of C functions, Prototype of a function: Formal parameter list, Return Type, Function call, Block structure, passing arguments to a Function: call by reference, call by value, Recursive Functions, arrays as function arguments. **Storage Classes :** Scope and extent, Storage Classes in a single source file: auto, extern and static, register, Storage Classes in a multiple source files: extern and static	8	15
M7	**File Processing** : Concept of Files, File opening in various modes and closing of a file, Reading from a file, Writing onto a file	6	10
	INTERNAL EXAMINATION	3	30
	TOTAL	48	100

PRACTICAL:

SUBJECT NAME: Programming for Problem Solving Lab   Credit: 2

SUBJECT CODE: BCAC192

List of Practical:

1. Write a c program to display the word "welcome".
2. Write a c program to take a variable int and input the value from the user and display it.
3. Write a c program to add 2 numbers entered by the user and display the result.

4. Write a c program to calculate the area and perimeter of a circle.
5. Write a C program to find maximum between two numbers.
6. Write a C program to check whether a number is divisible by 5 and 11 or not.
7. Write a C program to input angles of a triangle and check whether triangle is valid or not.
8. Write a C program to check whether a year is leap year or not.
9. Write a C program to input basic salary of an employee and calculate its Gross salaryaccording to following:
   Basic Salary <= 10000 : HRA = 20%,
   DA = 80% Basic Salary <= 20000 :
   HRA = 25%, DA = 90% Basic Salary
   > 20000 : HRA = 30%, DA = 95%
10. Write a c program to print “welcome” 10 times.
11. Write a c program to print first n natural numbers using while loop.
12. Write a c program to print all the odd numbers in a given range.
13. Write a c program to add first n numbers using while loop.
14. Write a c program to print all numbers divisible by 3 or 5 in a given range.
15. Write a c program to add even numbers in a given range.
16. Write a c program to find the factorial of a given number.
17. Write a c program to find whether a number is prime or not.
18. Write a c program to print the reverse of a number.
19. Write a c program to add the digits of a number.
20. Write a c program to print the fibonacci series in a given range.
21. Write a c program to check whether a number is an Armstrong number or not.
22. Write a c program to find g.c.d. and l.c.m. of two numbers.
23. Write a C program that writes “hello File Handling” to the File.
24. Write a C program that reads the information from the file.
25. Write a C program that defines a structure for student. Then Create five instance of the student and find who scores the highest marks. Write the information of the student who scores the highest marks.

SUGGESTED READING:

1. "Let Us C" by Yashavant Kanetkar Publisher: BPB Publications
2. "C Programming: A Modern Approach" by K. N. King Publisher: W. W. Norton & Company India Pvt. Ltd.
3. "C Programming for the Absolute Beginner" by Perry, Greg Publisher: Course Technology PTR (Cengage Learning)
4. "C How to Program" by Deitel, Paul, and Deitel, Harvey Publisher: Pearson Education India
5. "Programming in ANSI C" by Kochan, Stephen G. Publisher: Pearson Education India
6. "C Primer Plus" by Prata, Stephen Publisher: Pearson Education India
7. "C Programming Absolute Beginner's Guide" by Perry, Greg Publisher: Pearson Education India
8. "Programming with C" by Gottfried, Byron S. Publisher: Tata McGraw-Hill Education