MAULANA ABUL KALAM AZAD UNIVERSITY OF TECHNOLOGY, WEST BENGAL

(Formerly West Bengal University of Technology)

Syllabus of B.Sc. In IT (Cyber Security)

(Effective from 2023-24 Academic Sessions)

SEMESTER 1

Sl.	Course Type	Code	Course Name	Credits L	Credits T	Credits P	Total Credits
1	Major	BSCITCSM101 BSCITCSM191	Fundamental of Safe Programming practices Fundamental of Safe Programming practices using C lab	3	0	2	5
2	Major	BSCITCSM102 BSCITCSM192	Introduction to Digital Electronics Digital Electronics Lab	3	0	2	5
3	Minor	MIM101	Principles of Management	3	0	0	3
4	GE		Anyone from GE Basket A or D	3	0	0	3
5	AECC	AECC101	English & Professional Communication	2	0	0	2
6	SEC	SEC101	Life Skills & Personality Development	2	0	0	2
7	VAC	VAC181A VAC181B VAC181C	Yoga Health & Wellness Sports	2	0	0	2
Total Credits							22

Course: Fundamental of Safe Programming practices /Fundamental of Safe Programming practices using C lab
Credits: 3L + 2 P
Course Code – BSCITCSM101/BSCITCSM191

COURSE OBJECTIVE:

The objective of the course Fundamental of Safe Programming practices 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 safe solutions.

COURSE OUTCOME	
CO1	Apply programming constructs of C language to solve the real world problem
CO2	Understand the implementation of 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 secured 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

Module 1: Introduction to Principles of Programming

Introduction to Programming, Programming Domain: Scientific Application, Business Applications, Artificial Intelligence, Systems Programming, and Web application Categories of Programming Languages: Machine Level Languages, Assembly Level Languages and High-Level Languages.

Module 2:

Introduction to C Programming:

Features of C and its Basic Structure, Simple C programs, Constants, Integer Constants, Real Constants, Character Constants, String Constants, Backslash Character Constants, Concept of an Integer and Variable, Rules for naming Variables and assigning values to variables

Module 3:

Operators and Expressions:

Arithmetic Operators, Unary Operators, Relational and Logical Operators, The Conditional Operator, Library Functions, Bitwise Operators, The Increment and Decrement Operators, The Size of Operator, Precedence of operators.

Module 4:

Data Types and Input/output Operators:

Floating-point Numbers, Converting Integers to Floating-point and vice-versa, Mixed-mode Expressions, The type cast Operator, The type char, Keywords, Character Input and Output, Formatted input and output, The gets() and puts() functions, Interactive Programming.

Module 5 :

Control Statements and Decision-Making:

The if statement, The if-else statement, Nesting of if statements, The conditional expression, The switch statement, The while loop, The do…while loop, The for loop, The nesting of for loops, The break statement and continue statement, The goto statement: usability and security concerns.

Module 6 :

Arrays and Strings:

One-Dimensional Arrays, Passing Arrays to Functions, Multidimensional Arrays Strings - Concepts, C Strings, String Input/output Functions, Arrays of Strings, String Manipulation Functions.

Module 7:

Pointers:

Pointers for Inter-Function Communication, Pointers to Pointers, Arrays and Pointers, Pointer Arithmetic and Arrays, Passing an Array to a Function, Array of Pointers, Programming Applications, Pointers to void, Pointers to Functions. Safety concerns of pointer handling.

Module 9:

Structures and Unions:

Basics of Structures, Arrays of Structures, Pointers to Structures, Self-referential Structures, Unions

Module 10:

Functions:

Function Basics, Function Prototypes, and Passing Parameters: Passing Parameter by value and Passing Parameter by reference, passing string to function, Passing array to function, Structures and Functions Recursion

Module 11:

Storage Classes:

Storage Classes and Visibility, Automatic or local variables, Global variables, Static variables, External variables

Module 13:

Dynamic Memory Allocation:

Dynamic Memory Allocation, Allocating Memory with malloc, Allocating Memory with calloc, Freeing Memory, Reallocating Memory Blocks

Module 14:

File Management:

Defining and Opening a file, Closing Files, Input/output Operations on Files, Predefined Streams, Error Handling during I/O Operations, Random Access to Files

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 salary according 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 in 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 using recursion.
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 using function.

Text Books:

1. Herbert Schildt, “C: The Complete Reference”, Fourth Edition, McGraw Hill.
2. B. Gottfried, “Programming in C”, Second Edition, Schaum Outline Series.
3. R.S. Salaria, “Problem Solving and Programming in C”, Khanna Publishing House
4. E. Balagurusamy, “Programming in ANSI C”, Eighth Edition, McGraw Hill.

Reference Books:

1. B. W. Kernighan and D. M. Ritchi, The ‘C Programming Language”, Second Edition, PHI.
2. Yashavant Kanetkar, “Let Us C”, BPB Publication

Course: **Introduction to Digital Electronics /Digital Electronics Lab**
Credits: 3L + 2 P
Course Code – BSCITCSM102/BSCITCSM192

COURSE OBJECTIVE:

The objective of the course Introduction to 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

Module 1:
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

Module 2:
Logic Gates:
OR, AND, NOT, NAND, NOR, Exclusive – OR, Exclusive –NOR, Mixed logic.

Module 3:
Minimization Techniques
Sum of Products, Product of Sums, Karnaugh Map [up to 4 variables].

Module 4:
Multilevel Gate Network
Implementation of Multilevel Gate Network, Conversion to NAND-NAND and NOR-NOR Gate Networks.

Module 5:
Arithmetic Circuits
Half Adder, Full Adder, Half Subtractor, Full Subtractor, Carry Look Ahead Adder, 4-Bit Parallel Adder

Module 6:
Combinational Circuits
Basic 2-input and 4-input multiplexer, De-multiplexer, Basic binary decoder, BCD to binary converters, Binary to Gray code converters, Gray code to binary converters, Encoder.

Module 7:
Sequential Circuits
Introduction to sequential circuit, Latch, SR Flip Flop, D Flip Flop, T Flip Flop, JK Flip Flop, Master Slave Flip Flop

Module 8:
Basics of Counters
Asynchronous [Ripple or serial] counter, Synchronous [parallel] counter

Module 9:
Basics of Registers
SISO, SIPO, PISO, PIPO, Universal Register

Assignments:
Based on the curriculum as covered by subject teacher

Practical

Course Code: BSCITCSM192
Credit: 2

List of practical:-

1. Realization of basic gates using Universal logic gates.
2. Code conversion circuits- BCD to Excess-3 and viceversa.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.

Text Books:

1. DIGITAL DESIGN – Third Edition , M.Morris Mano, Pearson Education/P
2. Digital Circuits, Vol - I & II, D. Ray Chaudhuri, Platinum Publishers.
3. Digital Systems - Principle & Applications, Tocci & Widmer, EEE

Reference Books:

1. Fundamentals of Digital Circuits - Fourth edition, Kumar A. Anand, PHI.