VIDYASAGAR UNIVERSITY

Paschim Midnapore, West Bengal

PROPOSED CURRICULUM & SYLLABUS (DRAFT) OF

BACHELOR OF SCIENCE (HONOURS)
MAJOR IN COMPUTER SCIENCE

4-YEAR UNDERGRADUATE PROGRAMME

(w.e.f. Academic Year 2023-2024)

Based on

Curriculum & Credit Framework for Undergraduate Programmes (CCFUP), 2023 & NEP, 2020

VIDYASAGAR UNIVERSITY
BACHELOR OF SCIENCE (HONOURS) MAJOR IN COMPUTER SCIENCE
(under CCFUP, 2023)

Level	YR.	SEM	Course Type	Course Code	Course Title	Credit	L-T-P	Marks		
								CA	ESE	TOTAL
B.Sc. (Hons.)	1st	I	**SEMESTER-I**							
			Major-1	**COSHMJ101**	**T: Programming in C** **P: Programming in C Lab**	**4**	**3-0-1**	**15**	**60**	**75**
			SEC	**COSSEC01**	**P: Office Automation**	**3**	**0-0-3**	**10**	**40**	**50**
			AEC	**AEC01**	Communicative English -1 (***common for all programmes***)	**2**	**2-0-0**	**10**	**40**	**50**
			MDC	**MDC01**	Multidisciplinary Course -1 (***to be chosen from the list***)	**3**	**3-0-0**	**10**	**40**	**50**
			VAC	**VAC01**	ENVS (***common for all programmes***)	**4**	**2-0-2**	**50**	**50**	**100**
			Minor (Disc.-I)	**COSMI01**	**T: Computer Fundamental** ***(To be taken by students of other Disciplines)***	**4**	**3-1-0**	**15**	**60**	**75**
					Semester-I Total	**20**				**400**
		II	**SEMESTER-II**							
			Major-2	**COSHMJ102**	**T: Digital Logic Design** **P: Digital Logic Design Lab**	**4**	**3-0-1**	**15**	**60**	**75**
			SEC	**COSSEC02**	**P: Web Application**	**3**	**0-0-3**	**10**	**40**	**50**
			AEC	**AEC02**	MIL-1 (***common for all programmes***)	**2**	**2-0-0**	**10**	**40**	**50**
			MDC	**MDC02**	Multi Disciplinary Course-02 (***to be chosen from the list***)	**3**	**3-0-0**	**10**	**40**	**50**
			VAC	**VAC02**	Value Added Course-02 (***to be chosen from the list***)	**4**	**4-0-0**	**10**	**40**	**50**
			Minor (Disc.-II)	**COSMI02**	**T: Introduction to Programming ; P: Programming in C Lab** ***(To be taken by students of other Disciplines)***	**4**	**4-0-0**	**15**	**60**	**75**
			Summer Intern.	**CS**	Community Service	**4**	**0-0-4**	**-**	**-**	**50**
					Semester-II Total	**24**				**400**
					TOTAL of YEAR-1	**44**				**800**

MJ = Major, MI = Minor Course, SEC = Skill Enhancement Course, AEC = Ability Enhancement Course, MDC = Multidisciplinary Course, VAC = Value Added Course; CA= Continuous Assessment, ESE= End Semester Examination, T = Theory, P= Practical, L-T-P = Lecture-Tutorial-Practical, MIL = Modern Indian Language, ENVS = Environmental Studies

MAJOR (MJ)

MJ-1: Programming in C — Credits 04 (Full Marks: 75)

OBJECTIVE OF THE COURSE

The objectives of this course are to make the student understand programming language, programming, concepts of Loops, reading a set of Data, stepwise refinement, Functions, Control structure, Arrays. After completion of this course the student is expected to analyze the real-life problem and write a program in 'C' language to solve the problem. The main emphasis of the course will be on problem solving aspect i.e., developing proper algorithms.

- After completion of the course the student will be able to
- Develop efficient algorithms for solving a problem.
- Use the various constructs of a programming language viz. conditional, iteration and recursion.
- Implement the algorithms in "C" language.
- Use simple data structures like arrays, stacks and linked list in solving problems.
- Handling File in "C".

MJ-1T: Programming in C — Credits 03

Course contents:

Module- I Introduction to Programming 04 Hrs.
The Basic Model of Computation, Algorithms, Flow-charts, Programming Languages, Compilation, Linking and Loading, Testing and Debugging, Documentation

Module- II Algorithms for Problem Solving 10 Hrs.
xchanging values of two variables, summation of a set of numbers, Decimal Base to Binary Base conversion, Reversing digits of an integer, GCD (Greatest Common Division) of two numbers, Test whether a number is prime, Organize numbers in ascending order, Find square root of a number, factorial computation, Fibonacci sequence, Evaluate 'sin x' as sum of a series, Reverse order of elements of an array, Findlargest number in an array, Print elements of upper triangular matrix, multiplication of two matrices, Evaluate a Polynomial

Module- III Introduction to 'C' Language 04Hrs.
Character set, Variables and Identifiers, Built-in Data Types, Variable Definition, Arithmetic operators and Expressions, Constants and Literals, Simple assignment statement, Basic input/output statement, Simple 'C' programs.

Module- IV Conditional Statements and Loops 07 Hrs.
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.

Module- V Arrays 06 Hrs.
One dimensional array: 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

Module- VI Functions 06 Hrs.
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.

Module- VII Storage Classes 03 Hrs.
Scope and extent, Storage Classes in a single source file: auto, extern and static, register, Storage Classes in multiple source files: extern and static

Module- VIII Structures and Unions 06 Hrs.
Structure variables, initialization, structure assignment, nested structure, structures and functions, structures and arrays: arrays of structures, structures containing arrays, unions

Module- IX Pointers 06 Hrs.
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.

Module- X Self-Referential Structures and Linked Lists 04 Hrs.
Creation of a singly connected linked list, traversing a linked list, Insertion into a linked list, Deletion from a linked list

Module- XI File Processing 04 Hrs.
Concept of Files, File opening in various modes and closing of a file, reading from a file, writing onto a file

Suggested Readings:

1. Byron S Gottfried “Programming with C” Second edition, Tata McGrawhill, 2007 (Paperback)
2. R.G. Dromey, “How to solve it by Computer”, Pearson Education, 2008.
3. Kanetkar Y, “Let us C”, BPB Publications, 2007.
4. Hanly J R &Koffman E.B, “Problem Solving and Programm design in C”, Pearson Education, 2009.

Supplementary Reading

1. E. Balagurusamy, “Programming with ANSI-C”, Fourth Edition,2008, Tata McGraw Hill.
2. Venugopal K. R and Prasad S. R, “Mastering ‘C’”, Third Edition, 2008, Tata McGraw Hill.
3. B.W. Kernighan & D. M. Ritchie, “The C Programming Language”, Second Edition, 2001, Pearson Education
4. ISRD Group, “Programming and Problem Solving Using C”, Tata McGraw Hill, 2008.
5. Pradip Dey, ManasGhosh, “Programming in C”, Oxford University Press, 2007.

MJ-1P: Programming in C Lab Credits 01

1. Write a program to check a year is Leap year or not.
2. Write a program to solve the following Quadratic equation
   $Ax^2 + Bx + C = 0$
3. Write a program to print the sum and product of digits of an integer.
4. Write a program to find the reverse a number and then check the number is palindrome or not.
5. Write a program to compute the sum of the first n terms of the following series
   $S = 1+1/2+1/3+1/4+$……
6. Write a program to compute the sum of the first n terms of the following series
   $S =1-2+3-4+5$…………….
7. Write a program to find the value of cosxfrom the following Cos series:
   $Cosx = 1-x^2/2!+x^4/4!-$............∞
8. Write a program to find the GCD and LCM of two numbers.
9. Write a program to display Armstrong numbers between the range a to b.
10. Write a program to display Strong numbers between the range a to b.
11. Write a program to convert a Decimal number into its equivalent Binary number.
12. Write a program to convert a Binary number into its equivalent Decimal number.
13. Write a program to convert a Binary number into its equivalent Octal number.
14. Write a function to find whether a given no. is prime or not. Use the same to generate the prime numbers less than 100.
15. Write a program to compute the factors of a given number.
16. Write a function that checks whether a given string is Palindrome or not. Use this function to find whether the string entered by user is Palindrome or not.
17. Write a program to count number of vowels, consonants, digits and blank spaces in a line of text.
18. Write a macro that swaps two numbers. WAP to use it.
19. Write a program in which a function is passed address of two variables and then alter its contents.
20. Write a program to print a triangle of stars as follows (take number of lines from user):

```
    *
   ***
  *****
 *******
*********
```

21. Write a program to print the pyramid of numbers as follows(take number of lines from user):

```
    1
   121
  12321
 1234321
123454321
```

22. Write a program to display Fibonacci series (i) using recursion, (ii) using iteration
23. Write a program to calculate Factorial of a number (i) using recursion, (ii) using iteration
24. Write a program to calculate GCD of two numbers (i) with recursion (ii) without recursion.
25. Write a program to perform following actions on an array entered by the user:

i) Print the even-valued elements
ii) Print the odd-valued elements
iii) Calculate and print the sum and average of the elements of array
iv) Print the maximum and minimum element of array
v) Remove the duplicates from the array vi) Print the array in reverse order

26. Write a program to arrange the list of n numbers in ascending order.
27. Write a program that prints a table indicating the number of occurrences of each alphabet in the text entered as command line arguments.
28. Write a program which takes the radius of a circle as input from the user, passes it to another function that computes the area and the circumference of the circle and displays the value of area and circumference from the main() function.
29. Write a menu driven program to perform following operations on strings:
    a) Show address of each character in string
    b) Concatenate two strings without using strcat() function.
    c) Concatenate two strings using strcat() function.
    d) Compare two strings.
    e) Calculate length of the string (use pointers).
    f) Convert all lowercase characters to uppercase.
    g) Convert all uppercase characters to lowercase.
    h) Calculate number of vowels.
    i) Reverse the string
30. Create a structure Student containing fields for Roll No., Name, Class, Year and Total Marks. Create 10 students and store them in a file.
31. Write a program to retrieve the student information from file created in previous question and print it in following format: Roll No. Name Marks
32. Copy the contents of one text file to another file, after removing all whitespaces.
33. Write a program that will read 10 integers from user and store them in an array. Implement array using pointers. The program will print the array elements in ascending and descending order.

MJ-2: Digital Logic Design Credits 04

OBJECTIVE OF THE COURSE

The objective of digital logic design is to design and create digital circuits that perform specific functions and operations using digital logic gates. It involves the design and analysis of combinational and sequential circuits, which form the building blocks of digital systems such as processors, memory units, and control units.

MJ-1T: Digital Logic Design Credits 03

Module-I Number systems: 10 Hrs.

Positional number systems; Binary, Octal, Hexadecimal and Decimal number systems; conversion of a number in one system to the other; Representation of signed numbers-signed magnitude, one's complement, 2's complement representation techniques, Merits of 2's complement representation scheme; Various binary codes- BCD, excess -3, Gray code, ASCII, EBCDIC, Parity bits; Binary arithmetic-addition, subtraction, multiplication and division of unsigned binary numbers.

Module-II Boolean algebra: 10 Hrs.

Fundamental of Boolean Expression: Definition of Switching Algebra, Basic properties of Switching Algebra, Huntington’s Postulates, Basic Logic gates: (OR, AND, NOT); Universal Logic Gates: (NAND & NOR); Basic logic operations: logical sum (OR), logical product (AND), complementation (NOT), Anti coincidence (EX-OR) and coincidence (EX-NOR) operations: Truth tables of Basic gates; Boolean Variables and Expressions; De-Morgan’s theorem; Boolean expressions Simplification- Algebraic technique, Karnaugh map technique, 3 variable and 4 variable Karnaugh map.

Module-III Combinational Circuits: 13 Hrs.

Half Adder, Full Adder (3-bit), Half Subtractor, Full Subtractor (3-bit) and construction using Basic Logic Gates (OR, AND, NOT) and Universal Logic Gates (NAND & NOR), Multiplexer, Encoders, Demultiplexer and Decoder circuits, Seven Segment Display. BCD adder/ subtract or comparator; parity generators, code converters, priority encoders.

Module-IV Sequential Circuits: 12 Hrs.

Latch, RS, D, JK, T Flip Flops; Race condition, Master Slave JK Flip Flop; Registers: Serial Input Serial Output (SISO), Serial Input Parallel Output (SIPO), Parallel input Serial Output (PISO), Parallel Input parallel Output (PIPO), Universal Shift Registers; Counters: Asynchronous Counter, Synchronous Counter.

Suggested Readings:

1. Digital Principles and Applications: Malvino and Leach
2. Modern Digital Electronics : R.P. Jain.
3. Digital Circuits & Design – S.Salivahanan, S.Arivazhagan– Vikas Publishing House Pvt Ltd.
4. Digital logic & Computer Design- M.Mano- Prentice Hall of India.
5. Fundamental of Digital Circuits by AAnnand Kumar PHI

Supplementary Reading

1. "Digital Design and Computer Architecture" by David Harris and Sarah Harris
2. "Digital Logic Design: A Rigorous Approach" by Guy Even and Moti Medina
3. "Digital Systems: Principles and Applications" by Ronald J. Tocci, Neal S. Widmer, and Greg Moss

MJ-2P: Digital Logic Design Credits 01

Combinational Circuits & Sequential Circuits:

1. Implementation of different functions using Basic and Universal Logic gates, SOP, POS

2. Study and prove De-Morgan’s Theorem.

3. Implementation of Basic gates using NAND and NOR gates

4. Implementation of half and Full Adder (3-bit) using basic logic gates and Universal logic gates (NAND & NOR).

5. Implementation of half and Full Subtractor (3-bit) using basic logic gates and Universal logic gates (NAND & NOR).

6. Design 2 to 4 decoder using basic / universal logic gates.
7. Design and implement a 8:1 multiplexer.
8. Design and implement a 3×8 decoder.
9. Design and implement a 8 bit parity generator.
10. Design and implement a D flip-flop.
11. Design and implement a J. K. flip-flop.
12. Design and implement a 4-bit synchronous counter.

MINOR (MI)

MI – 1: Computer Fundamental **Credits 04 (Full Marks: 75)**

OBJECTIVE OF THE COURSE

The objective of the course "Computer Fundamentals" is to provide students with a comprehensive understanding of the basic principles, components, and operations of computers. The course aims to equip students with the necessary knowledge and skills to effectively use and work with computers in their personal and professional lives.

One of the primary objectives is to familiarize students with the fundamental concepts and terminology related to computers. This includes understanding the history and evolution of computers, different generations of computer technology, and the basic structure and functioning of a computer system. Students will learn about the essential hardware components such as the CPU, memory, storage devices, input devices, and output devices, as well as the purpose and functionality of each component.

MI – 1T: Computer Fundamental **Credits 04**

Course contents:

MODULE- I: Introduction **(10 Hrs.)**

Definition of computer.Characteristics of computer.Generation of computer. Classificationof computer (Micro, Mini, Mainframe, Super), Application of computer, Basic concept aboutSoftware & Hardware, Bit, Byte, Word Nibble, Computer Languages (Low, High & assembly LevelLanguage)

MODULE-II: Basic Components of Computer **(10 Hrs.)**

Basic organization of digital computer (CPU, CU, ALU, Register set, Communication Path way, Input / Output Devices, Memory Module).CPU: Basic explanation about CU, ALU &Register set as well as all over CPU. Communication Pathway: Definition of Bus, Internal &External Bus,Control, Address & Data Bus. Input devices: Keyboard, Pointing device, handheld device, Opticaldevice, Audio visual device. Output device: Soft copy devices & hard copy devices. MemoryHierarchy (Definition, function, classification, Advantages & Disadvantages): Primary Memory, Secondary Memory, Cache Memory, Virtual Memory.

MODULE-III: Number System ` **(10 Hrs.)**

Definition, Positional & non positional number system, Binary,Decimal, octal & hexadecimal number system, Conversion between them, Binary-Decimal-Octal Hexadecimal arithmetic, Signed & Unsigned number, Complement notation (r’s & (r-1)’s complement), Addition &Subtraction operation using complement notation, Floating point representation of number, Computer codes(Weighted binary codes (BCD 8421/2421,Reflective, sequential), Non-weighted binary codes(Excess-3, Gray), Error detecting & correcting codes, Alphanumeric codes(ASCII, EBCDIC, Hollerith),BCDaddition, Gray to Binary & Binary to Gray conversion.

MODULE-IV: Data communication and Computer network **(10 Hrs.)**

Definition of data communication, Characteristics of data communication, Component of data communication, mode of data Communication, Media of data communication (guided & unguided), Channel capacity. Computer Network: Definition, Network topology (Bus, Ring, Star, Mesh, Tree, Hybrid), Types of network (LAN, MAN, WAN, CCAN, PAN), Network devices (Hub, Repeater, Switch, Bridge, Router, Gateway), Basic idea about e-mail, Search engines, Chatting, Internet conferencing, Intranet.

MODULE-V: Operating System (10 Hrs.)

Definition of OS, Function of OS, Need of OS, Classification of OS (CUI & GUI, Single user, Multi User), Concept of Multi Programming, Multi Tasking& Multi Processing. Booting Process), Basic Concept of Assembler, Loader, Linker, Interpreter.

Suggested Readings:

1. Sinha, P. K., &Sinha, P. (2017). Computer Fundamentals: Concepts, Systems & Applications. BPB Publications.
2. Rajaraman, V. (2017). Fundamentals of Computers. PHI Learning.
3. Prakash, S. (2019). Computer Fundamentals and Programming in C. Laxmi Publications.
4. Pradhan, S. (2017). Computer Fundamentals: Architecture and Organization. Oxford University Press.
5. Bharadwaj, A. S. (2017). Computer Fundamentals and Applications. Wiley India.
6. Deo, N. (2017). Fundamentals of Computers. Dreamtech Press.Acharya, S., &Kamath, M. V. (2017). Computer Fundamentals. Prentice

MI-2: Introduction to Programming Credits 04 (Full Marks: 75)

OBJECTIVE OF THE COURSE

The objectives of this course are to make the student understand programming language, programming, concepts of Loops, reading a set of Data, stepwise refinement, Functions, Control structure, Arrays. After completion of this course the student is expected to analyze the real-life problem and write a program in 'C' language to solve the problem. The main emphasis of the course will be on problem solving aspect i.e., developing proper algorithms.

- After completion of the course the student will be able to
- Develop efficient algorithms for solving a problem.
- Use the various constructs of a programming language viz. conditional, iteration and recursion.
- Implement the algorithms in "C" language.
- Use simple data structures like arrays, stacks and linked list in solving problems.
- Handling File in "C".

MI-2T: Introduction to Programming Credits 03

Course contents:

Module- I Introduction to Programming 04 Hrs.

The Basic Model of Computation, Algorithms, Flow-charts, Programming Languages, Compilation, Linking and Loading, Testing and Debugging, Documentation

Module- II Algorithms for Problem Solving 10 Hrs.

Exchanging values of two variables, summation of a set of numbers, Decimal Base to Binary Base conversion, Reversing digits of an integer, GCD (Greatest Common Division) of two numbers, Test whether a number is prime, Organize numbers in ascending order, Find square root of a number, factorial computation, Fibonacci sequence, Evaluate 'sin x' as sum of a series, Reverse order of elements of an array, Find largest number in an array, Print elements of upper triangular matrix, multiplication of two matrices, Evaluate a Polynomial

Module- III Introduction to 'C' Language 04Hrs.

Character set, Variables and Identifiers, Built-in Data Types, Variable Definition, Arithmetic operators and Expressions, Constants and Literals, Simple assignment statement, Basic input/output statement, Simple 'C' programs.

Module- IV Conditional Statements and Loops 07 Hrs.

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.

Module- V Arrays 06 Hrs.

One dimensional array: 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

Module- VI Functions 06 Hrs.

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.

Module- VII Storage Classes 03 Hrs.

Scope and extent, Storage Classes in a single source file: auto, extern and static, register, Storage Classes in multiple source files: extern and static

Module- VIII Structures and Unions 06 Hrs.

Structure variables, initialization, structure assignment, nested structure, structures and functions, structures and arrays: arrays of structures, structures containing arrays, unions

Module- IX Pointers 06 Hrs.

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.

Module- X Self-Referential Structures and Linked Lists 04 Hrs.

Creation of a singly connected linked list, traversing a linked list, Insertion into a linked list, Deletion from a linked list

Module- XI File Processing 04 Hrs.

Concept of Files, File opening in various modes and closing of a file, reading from a file, writing onto a file

Suggested Readings:

1. Byron S Gottfried "Programming with C" Second edition, Tata McGrawhill, 2007 (Paperback)
2. R.G. Dromey, "How to solve it by Computer", Pearson Education, 2008.
3. Kanetkar Y, "Let us C", BPB Publications, 2007.
4. Hanly J R &Koffman E.B, "Problem Solving and Programm design in C", Pearson Education, 2009.

Supplementary Reading

5. E. Balagurusamy, "Programming with ANSI-C", Fourth Edition,2008, Tata McGraw Hill.
6. Venugopal K. R and Prasad S. R, "Mastering 'C'", Third Edition, 2008, Tata McGraw Hill.
7. B.W. Kernighan & D. M. Ritchie, "The C Programming Language", Second Edition, 2001, Pearson Education
8. ISRD Group, "Programming and Problem Solving Using C", Tata McGraw Hill, 2008.
9. Pradip Dey , Manas Ghosh, "Programming in C", Oxford University Press, 2007.

MI-2P: Programming in C Lab Credits 01

Course Outline:

1. WAP to check a year is Leap year or not.
2. WAP to solve the following Quadratic equation

   $Ax^2 + Bx + C = 0$
3. WAP to print the sum and product of digits of an integer.
4. WAP to find the reverse a number and then check the number is palindrome or not.
5. WAP to compute the sum of the first n terms of the following series

   S = 1+1/2+1/3+1/4+……
6. WAP to compute the sum of the first n terms of the following series

   S =1-2+3-4+5…………….
7. WAP to find the value of cosxfrom the following Cos series:

   $\text{Cos}x = 1-x^2/2!+x^4/4!-.............\infty$
8. WAP to find the GCD and LCM of two numbers.
9. WAP to display Armstrong numbers between the range a to b.
10. WAP to display Strong numbers between the range a to b.
11. WAP to convert a Decimal number into its equivalent Binary number.
12. WAP to convert a Binary number into its equivalent Decimal number.
13. WAP to convert a Binary number into its equivalent Octal number.
14. Write a function to find whether a given no. is prime or not. Use the same to generate the prime numbers less than 100.

SKILL ENHANCEMENT COURSE (SEC)

SEC 1: Office Automation Credits 03 (Full Marks: 50)

OBJECTIVE OF THE COURSE -

The objective of the Office Automaton course is to equip participants with the knowledge and skills necessary to automate various tasks and streamline processes in an office environment using automation tools and techniques. The course aims to enhance productivity, efficiency, and effectiveness by leveraging automation technologies.

SEC1P: Office Automation Credits 03

Course Outline:

1. Introduction to Office Automation
   - Definition and benefits of office automation
   - Role of automation in modern workplaces
   - Overview of automation tools and technologies
2. Business Process Analysis and Automation
   - Identifying and analyzing office processes for automation
   - Process mapping and documentation
   - Determining automation feasibility and potential ROI
3. Spreadsheet Automation
   - Introduction to advanced spreadsheet features and functions
   - Automating data entry, calculations, and reporting
   - Macros and scripting for spreadsheet automation
4. Email Automation
   - Managing and organizing emails efficiently
   - Automating email filtering and sorting
   - Creating email templates and automatic responses
5. Document Automation
   - Automating document creation and formatting
   - Using document templates and mail merge
   - Document versioning and collaboration automation

SEC 2: Web Application Credits 03 (Full Marks: 50)

OBJECTIVE OF THE COURSE -

When it comes to web application development, the objectives and syllabus may vary depending on the specific course or program you're undertaking. However, here's a general outline of the objectives and topics commonly covered in a web application development Objective of the Course.web application development course can vary depending on the specific objectives and level of the course. However, here is a general outline of topics that are commonly covered in a web application development.

SEC 2P: Web Application **Credits 03**

Course Outline:

1. **Introduction to Web Development:**
   - Overview of web technologies
   - Client-server architecture
   - Introduction to HTML, CSS, and JavaScript
2. **Front-End Development:**
   - HTML fundamentals and document structure
   - CSS styling and layout techniques
   - JavaScript basics and DOM manipulation
   - Responsive web design and mobile optimization
   - Introduction to front-end frameworks (e.g., Bootstrap, Foundation)
3. **Back-End Development:**
   - Introduction to server-side programming languages (e.g., Python, Ruby, PHP)
   - Introduction to databases (e.g., SQL, MySQL, MongoDB)
   - Server-side frameworks and libraries (e.g., Express.js, Django, Ruby on Rails)
   - Web APIs and RESTful services
4. **Database Integration:**
   - Relational database management systems (RDBMS)
   - Database modeling and design
   - SQL queries and data manipulation
   - Database connectivity and integration with web applications
5. **Authentication and Security:**
   - User authentication and authorization
   - Session management and cookies
   - Cross-site scripting (XSS) and cross-site request forgery (CSRF) prevention
   - Secure coding practices and common security vulnerabilities
6. **Web Application Frameworks:**
   - Introduction to popular web application frameworks (e.g., Angular, React, Vue.js)
   - Single-page application (SPA) development
   - Component-based architecture
   - State management and routing
7. **Web Services and APIs:**
   - Introduction to web services and APIs
   - Representational State Transfer (REST) principles
   - API design and documentation
   - Consuming third-party APIs