GURU JAMBHESHWAR UNIVERSITY OF SCIENCE & TECHNOLOGY, HISAR
(Established by State Legislature Act 17 of 1995)
'A+' Grade, NAAC Accredited State Govt. University

Acad./AC-III/BOS&R-6/2025/ 2768
Dated: 22/5/25

To

The Controller of Examinations
GJUS&T, Hisar.

Sub: Approval of the scheme of examinations and syllabi of Integrated BCA-MCA programme w.e.f. academic session 2024-25 (Minor w.e.f. academic session 2025-26) – 1st to 6th semester under scheme 'D' being run in University Teaching Department.

Sir,

I am directed to inform you that the Vice-Chancellor, on the recommendations of Dean, Faculty of Engineering & Technology on dated 12.05.2025, is pleased to approve the scheme of examinations and syllabi of Integrated BCA-MCA programme w.e.f. academic session 2024-25 (Minor w.e.f. academic session 2025-26) – 1st to 6th semester under scheme 'D' being run in University Teaching Department, under Section 11(5) of the University Act, 1995 in anticipatton of approval of the Academic Council.

A copy of the scheme of examinations & syllabi of above said course is enclosed herewith.

You are therefore, requested to take further necessary action accordingly.

DA: As above

Yours faithfully
22/5/25
Assistant Registrar (Academic)
for Registrar

Endst. No. Acad./AC-III/BOS&R-6/2025/ 2769-72 Dated: 22/5/25

A copy of the above is forwarded to the following for information and necessary action:-

1. Dean, Faculty of Engineering & Technology, GJUST, Hisar.
2. Chairperson, Department of Computer Science & Engineering, GJUST, Hisar. He is requested to get upload the scheme of examinations & syllabi of above said course being run in University Teaching Department on the website of the University.
3. OSD to Vice-Chancellor (for kind information of the Vice-Chancellor), GJUST, Hisar.
4. Secretary, Office of Registrar (for kind information of the Registrar), GJUST, Hisar.

22/5/25
Assistant Registrar (Academic)

27/5/25
For Web please.
Dr Sunil Kumar
Mr Ashwani Kumar

Department of Computer Science & Engineering

Integrated BCA-MCA [Master of Computer Applications]

(Integrated)

Semester-1(w.e.f Academic Session 2024-25)

(Minor w.e.f. Session 2025-26)

Scheme-D

Subject: Computer Applications

S. No.	Course Code	Nomenclature	Course Type	Hours / Week			Credits	Exam Hours	Int.	Ext.	Total
				L	T	P					
Discipline Specific Courses (DSC)											
1	24ICA0101T	Essentials of Computer Science	DSC-A1	3	0	0	3	2.5	20	50	70
2.	24ICA0101P	Essentials of Computer Science Lab	DSC-A1	0	0	2	1	3	10	20	30
3	24ICA0102T	Problem Solving using C	DSC-A2	3	0	0	3	2.5	20	50	70
4	24ICA0102P	Problem Solving using C Lab	DSC-A2	0	0	2	1	3	10	20	30
5	24ICA0103T	Fundamentals of PC Software	DSC-A3	3	0	0	3	2.5	20	50	70
6	24ICA0103P	Fundamentals of PC Software Lab	DSC-A3	0	0	2	1	3	10	20	30
Minor Course (MIC)											
7	24MIC0121T	Essentials of Computer	MIC1	2	0	0	2	2	15	35	50
Multidisciplinary Course (MDC)											
8	24MDC0110T	Fundamental of Computer Science	MDC1	3	0	0	3	2.5	25	50	75
Skill Enhancement Course (SEC)											
9	24SEC0110T	PC Hardware and Networking	SEC1	2	0	0	2	2	15	35	50
10	24SEC0110P	PC Hardware and Networking Lab	SEC1	0	0	2	1	3	10	15	25

Department of Computer Science & Engineering

Integrated BCA-MCA [Master of Computer Applications] (Integrated)

Semester-2 (w.e.f Academic Session 2024-25)

(Minor w.e.f. Session 2025-26)

Scheme-D

Subject: Computer Applications

S. No.	Course Code	Nomenclature	Course Type	Hours / Week			Credits	Exam Hours	Int.	Ext.	Total
				L	T	P					
Discipline Specific Courses (DSC)											
1	24ICA0201T	Database Technologies-1	DSC-A4	3	0	0	3	2.5	20	50	70
2	24ICA0201P	Database Technologies-1 Lab	DSC-A4	0	0	2	1	3	10	20	30
3	24ICA0202T	Object Oriented Programming using C++	DSC-A5	3	0	0	3	2.5	20	50	70
4	24ICA0202P	Object Oriented Programming using C++ Lab	DSC-A5	0	0	2	1	3	10	20	30
5	24ICA0203T	Data Structures & Algorithms	DSC-A6	3	0	0	3	2.5	20	50	70
6	24ICA0203P	Data Structures & Algorithms Lab	DSC-A6	0	0	2	1	3	10	20	30
Minor Course (MIC)											
7	24MIC0221T	Operating System	MIC2	2	0	0	2	2	15	35	50
Multidisciplinary Course (MDC)											
8	24MDC0210T	Programming for Problem Solving using C	MDC2	3	0	0	3	2.5	25	50	75
Skill Enhancement Course (SEC)											
9	24SEC0210T	Web Designing Using HTML	SEC2	2	0	0	2	2	15	35	50
10	24SEC0210P	Web Designing Using HTML Lab	SEC2	0	0	2	1	3	10	15	25
		**** Internship of 4 credits after 2nd Semester**									

****** **A summer term is for Eight Weeks i.e Two Months during Summer Vacation Internship / Training can be carried out during the summer term specially by the students who wish to exit after Two Semester of study.**

Department of Computer Science & Engineering

Integrated BCA-MCA [Master of Computer Applications]

(Integrated)

Semester-3 (w.e.f Academic Session 2024-25)

(Minor w.e.f. Session 2025-26)

Scheme-D

Subject: Computer Applications

S. No.	Course Code	Nomenclature	Course Type	Hours / Week			Credits	Exam Hours	Int.	Ext.	Total
				L	T	P					
Discipline Specific Courses (DSC)											
1	24ICA0301T	Operating System	DSC-A1	3	0	0	3	2.5	20	50	70
2.	24ICA0301P	Operating System Lab	DSC-A1	0	0	2	1	3	10	20	30
3	24ICA0302T	Software Engineering	DSC-A2	3	0	0	3	2.5	20	50	70
4	24ICA0302P	Software Engineering Lab	DSC-A2	0	0	2	1	3	10	20	30
5	24ICA0303T	Discrete Mathematics	DSC-A3	4	0	0	4	3	30	70	100
Minor Course (MIC)											
6	24MIC0321T	Computer Network	MIC3	3	0	0	3	2.5	20	50	70
7	24MIC0321P	Computer Network Lab	MIC3	0	0	2	1	3	10	20	30
Multidisciplinary Course (MDC)											
8	24MDC0310T	Software Engineering	MDC3	3	0	0	3	2.5	25	50	75
Skill Enhancement Course (SEC)											
9	24SEC0310T	Mobile Application Development	SEC3	2	0	0	2	2	15	35	50
10	24SEC0310P	Mobile Application Development Lab	SEC3	0	0	2	1	3	10	15	25

Department of Computer Science & Engineering

Integrated BCA-MCA [Master of Computer Applications]

(Integrated)

Semester-4 (w.e.f Academic Session 2024-25)

(Minor w.e.f. Session 2025-26)

Scheme-D

Subject: Computer Applications

S. No.	Course Code	Nomenclature	Course Type	Hours / Week			Credits	Exam Hours	Int.	Ext.	Total
				L	T	P					
Discipline Specific Courses (DSC)											
1	24ICA0401T	Computer Architecture	DSC-A1	3	0	0	3	2.5	20	50	70
2.	24ICA0401P	Computer Architecture Lab	DSC-A1	0	0	2	1	3	10	20	30
3	24ICA0402T	Computer Network	DSC-A2	3	0	0	3	2.5	20	50	70
4	24ICA0402P	Computer Network Lab	DSC-A2	0	0	2	1	3	10	20	30
5	24ICA0403T	Computer Graphics	DSC-A3	3	0	0	3	2.5	20	50	70
6	24ICA0403P	Computer Graphics Lab	DSC-A3	0	0	2	1	3	10	20	30
Minor / Vocational Course (MIC)											
7	24VOC0421T	Data Structure & Algorithm	MIC4(V)	2	0	0	2	2	15	35	50
8.	24VOC0421P	Data Structure & Algorithm Lab	MIC4(V)	0	0	4	2	4	15	35	50
	**** Internship of 4 credits after 4th Semester**										

** Four credits of internship, earned by a student during semester internship after 4th semester, will be taken into account in 5th semester of a student who pursues 3 years UG Program without taking exit option.

Department of Computer Science & Engineering

Integrated BCA-MCA [Master of Computer Applications]

(Integrated)

Semester-5 (w.e.f Academic Session 2024-25)

(Minor w.e.f. Session 2025-26)

Scheme-D

Subject: Computer Applications

S. No.	Course Code	Nomenclature	Course Type	Hours / Week			Credits	Exam Hours	Int.	Ext.	Total
				L	T	P					
Discipline Specific Courses (DSC)											
1		To be Decided later on		3	0	0	3	2.5	20	50	70
2.		To be Decided later on		0	0	2	1	3	10	20	30
3		To be Decided later on		3	0	0	3	2.5	20	50	70
4		To be Decided later on		0	0	2	1	3	10	20	30
5		To be Decided later on		3	0	0	3	2.5	20	50	70
6		To be Decided later on		0	0	2	1	3	10	20	30
Minor / Vocational Course (MIC)											
7	24VOC0521T	Data Base Management System	MIC5(V)	2	0	0	2	2	15	35	50
8	24VOC0521P	Data Base Management System Lab	MIC5(V)	0	0	4	2	4	15	35	50

Department of Computer Science & Engineering

Integrated BCA-MCA [Master of Computer Applications]

(Integrated)

Semester-6 (w.e.f Academic Session 2024-25)

(Minor w.e.f. Session 2025-26)

Scheme-D

Subject: Computer Applications

S. No.	Course Code	Nomenclature	Course Type	Hours / Week			Credits	Exam Hours	Int.	Ext.	Total
				L	T	P					
Discipline Specific Courses (DSC)											
1		To be Decided later on		3	0	0	3	2.5	20	50	70
2.		To be Decided later on		0	0	2	1	3	10	20	30
3		To be Decided later on		3	0	0	3	2.5	20	50	70
4		To be Decided later on		0	0	2	1	3	10	20	30
5		To be Decided later on		3	0	0	3	2.5	20	50	70
6		To be Decided later on		0	0	2	1	3	10	20	30
Minor / Vocational Course (MIC)											
7	24MIC0621T	Web Development	MIC6	3	0	0	3	2.5	20	50	70
8	24MIC0621P	Web Development Lab	MIC6	0	0	2	1	3	10	20	30
9	24VOC0621T	Data Analytics using R	MIC7(V)	2	0	0	2	2	15	35	50
10	24VOC0621P	Data Analytics using R Lab	MIC7(V)	0	0	4	2	4	15	35	50

Discipline Specific Courses (DSC):

Essentials of Computer Science

General Course Information:

Course Code: 24ICA0101T Course Credits: 3 Type: Discipline Specific Courses (DSC) Contact Hours: 3 hours/week Mode: Lectures (L) Exam Duration: 2.5 hours	**Course Assessment Methods (Internal: 20; External: 50)** **Internal Assessment: -** Two minor examinations each of 10 marks, Class Performance measured through percentage of lectures attended (5 marks) Assignment and quiz (5 marks), and end semester examination of 50 marks. **External Examination (Marks: 50): -** The syllabus is divided into four units. For the end semester examination, nine questions are to be set by the examiner. Question No. 1 is compulsory and contains five short answer questions (2 marks each) covering entire syllabus. Rest eight more questions (each question of at least two parts) will be set by giving two questions from each unit of the syllabus. A candidate is required to attempt five questions in all by selecting one question from each unit in addition to compulsory Question No.1. All questions will carry equal marks.

Prerequisite: Basic computer Literacy, Basic knowledge of protecting data from malware.

About the Course:

This course has been designed to provide students with an overview of the concepts & fundamentals of computers, information & Communication technology and GUI based operating system. This course describes the data types, its digital representation, security issues and various ways of user's well-being as well as Green IT.

Course Outcomes: By the end of the course students will be able to:

CO1 Lists : key concepts relating to ICT, computers, devices and software.
CO2 Describe: various types of GUI based operating systems which effectively work on the system.
CO3 Apply : file management, efficient organization of files and folders, compress and extract large files.
CO4 Design : various policies of protecting data and devices from malware.
CO5 Analyze : computer accessibility by using voice recognition software and examine the ways of the user well-being to promote green IT.

Course Contents

Unit - I

Computers and Devices: Information and Communication Technology (ICT), ICT services: Internet services, mobile technology, office productivity applications. Types of computers: desktops, laptops, tablets and main types of devices: smartphones, media players, digital cameras. Identify the main types of integrated and external equipment like: printers, screens, scanners, keyboards, mouse/track pad, webcam, speakers, microphone, docking station and common input/output ports: USB, HDMI.

Unit-II

GUI Based Operating System: Operating System: LINUX, WINDOWS. User Interface: Task Bar, Icons, Menu, Applications of operating System: Settings of system Date and Time, Changing Display Properties: adding and removing printers. File and Directory Management: Creating and renaming of files and directories and common utilities.

Unit – III

Data types and its digital representation: Binary numeral system: two binary numbers, representation of real numbers, 2's complement operations: Fixed-point and floating-point numbers. Binary codes: BCD vs. ASCII codes. Error codes: Hamming distance, Parity codes, CRC codes.

Unit – IV

Security and Well-Being: password policies: create with adequate length, character mix, change regularly. Define firewall and outline its purpose. Software updates: anti-virus, application, operating system software. Malware and its type: virus, worm, Trojan, spyware. Health and Green IT: Ways of user's well-being: take regular breaks, ensure appropriate lighting and posture. Device energy saving: turning off, adjusting automatic shutdown, backlight and sleep mode settings. Enhancing accessibilities: voice recognition software, screen reader, screen magnifier, on-screen keyboard, high contrast.

Text and Reference Books:

1. J. Glenn Brookshear," Computer Science: An Overview", Addision-Wesley, Twelfth Edition, 2014.
2. PC Software for Windows 98 made simple, R.K.Taxali, Tata McGraw Hill Publishers, 2015.
3. Fundamental ofdigital systems. Thomas L. Floyd. Prentice Hall. 9ª Ed. 2006.
4. Digital Design and Computer Architecture. David Money Harris y Sarah L. Harris. Morgan Kaufmann. 2007.
5. Fundamental of digital Logic gates and computations. M. Morris Mano y Charles R. Kime. Prentice Hall. 3ª Ed. 2005.

Essentials of Computer Science Lab

General Course Information:

Course Code: 24ICA0101P Course Credits: 1 Type: Discipline Specific Courses (DSC) Contact Hours: 2 hours/week Mode: Lab Exam Duration: 3 hours	**Course Assessment Methods (Internal: 10; External: 20)** The Internal and External assessment is based on the level of participation in Lab sessions and the timely submission of Lab experiments / assignments, the performance in Viva-Voce, the quality of the lab file and ethical practices followed. The Internal examination is conducted by the Course Coordinator. The External examination is conducted by External Examiner appointed by the Controller of Examination in association with the Internal Examiner appointed by the Chairperson of the Department.

Pre-requisite: Students are expected to have basic knowledge of problem solving and Essentials components of system.

About the Course: This course has been designed to provide students with an understanding of the concepts & fundamentals of OS Installation and database concepts. This course analyses the problems and executes DOS and LINUX commands. Students will interact with Problem solving aspect i.e. Security and Well-Being.

Course Outcomes: By the end of the course students will be able to:

CO1. **Implementation :** Creation of database using DOS & LINUX commands.
CO2. **Analysis :** Enforce the uses of ICT services.
CO3. **Compare :** Analyze the interfaces of computer and its peripheral devices.
CO4. **Integrate :** Security and Well-Being.
CO5. **Create :** Execute Lab assignments for various problems.
CO6. **Demonstrate :** Demonstrate ethical practices, self-learning and team spirit.

List of Experiments/ assignments

1. Lab Component- OS installation.
2. At least five basic DOS commands.
3. At least five basic LINUX commands.
4. Create and Renaming of a files.
5. Create and Renaming of a directories.
6. Change the System date & time.
7. Add or Remove a windows component i.e Printer & Scanner.
8. Creation of Email account.
9. Web Browsing, Emails, Searching of Contents.
10. Create of password with adequate length and character mix & symbols.
11. Installation of Firewall.
12. Make settings of Backlight and Sleep Mode in computer system.
13. Installation of Voice Recognition Software.
14. How to use On-screen Keyboard in computer system.
15. Online updated Antivirus Software.

Note: The actual experiment / assignments will be designed by the course coordinator. One assignment to be done in groups of two or three students. The assignments must meet the objectives of the course and the levels of given course outcomes. The list of assignments and schedule of submission will be prepared by course coordinator in the beginning of the semester.

Problem Solving using C

General Course Information:

Course Code: 24ICA0102T Course Credits: 3 Type: Discipline Specific Courses (DSC) Contact Hours: 3 hours/week Mode: Lectures (L) Exam Duration: 2.5 hours	**Course Assessment Methods (Internal: 20; External: 50)** **Internal Assessment: -** Two minor examinations each of 10 marks, Class Performance measured through percentage of lectures attended (5 marks) Assignment and quiz (5 marks), and end semester examination of 50 marks. **External Examination (Marks: 50): -** The syllabus is divided into four units. For the end semester examination, nine questions are to be set by the examiner. Question No. 1 is compulsory and contains five short answer questions (2 marks each) covering entire syllabus. Rest eight more questions (each question of at least two parts) will be set by giving two questions from each unit of the syllabus. A candidate is required to attempt five questions in all by selecting one question from each unit in addition to compulsory Question No.1. All questions will carry equal marks.

Prerequisite:
Students are expected to have basic knowledge of problem solving and programming construct.

About the Course: This course has been designed to provide students with an overview of the basic concepts of programming language, concepts of loops, reading a set of data, stepwise refinement, Function, Control structures, Arrays, File handling. This course describes efficient algorithms in C language to solve real-life problems.

Course Outcomes: By the end of the course students will be able to:

CO1**. List :** simple algorithms for arithmetic and logical problems.
CO2**. Select :** efficient algorithms for solving a problem
CO3. **Apply :** various constructs of a programming language viz., conditional, iteration and recursion.
CO4. **Implement:** algorithm in 'C' language for Structure, Unions and Pointers.
CO5. **Apply :** simple data structures in solving problems and file handling concept.

Course Contents

Unit – I

Introduction to Programming: Basic Model of Computation, Algorithms, Flow-charts, Programming Languages, Need for computer programming languages, Compilation, Linking and Loading, Testing and Debugging, Documentation. Problem solving and algorithms for Problem Solving**:** 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.
Introduction to 'C' Language: History of C, 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.

Unit - II

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.
Arrays: 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.

Unit – III

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 multiple source files: extern and static

Unit – IV

Structures and Unions: Structure variables, initialization, structure assignment, nested structure, structures and functions, structures and arrays: arrays of structures, structures containing arrays, unions.
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.
File Processing: Concept of Files, File opening in various modes and closing of a file, Reading from a file, writing onto a file

T**ext and Reference Books:**

1. Byron S Gottfried "Programming with C" Second edition, Tata McGrawhill, 2007 .
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.
5. E. Balagurusamy, "Programming with ANSI-C", Fourth Edition,2008, Tata McGraw Hill.
6. E. Balagurusamy, "Programming for Problem Solving", First Edition,2018, Tata McGraw Hill.
7. Venugopal K. R and Prasad S. R, "Mastering 'C'", Third Edition, 2008, Tata McGraw Hill.
8. B.W. Kernighan & D. M. Ritchie, "The C Programming Language", Second Edition, 2001, Pearson Education
9. ISRD Group, "Programming and Problem Solving Using C", Tata McGraw Hill,2008.
10. PradipDey ,ManasGhosh, "Programming in C", Oxford University Press, 2007.

Problem Solving using C Lab

General Course Information:

Course Code: 24ICA0102P Course Credits: 1 Type: Discipline Specific Courses (DSC) Contact Hours: 2 hours/week Mode: Lab Exam Duration: 3 hours	**Course Assessment Methods (Internal: 10; External: 20)** The Internal and External assessment is based on the level of participation in Lab sessions and the timely submission of Lab experiments / assignments, the performance in Viva-Voce, the quality of the lab file and ethical practices followed. The Internal examination is conducted by the Course Coordinator .The External examination is conducted by External Examiner appointed by the Controller of Examination in association with the Internal Examiner appointed by the Chairperson of the Department.

Prerequisite: Students are expected to have basic knowledge of problem solving and programming constructs.

About the Course: This course has been designed to provide students with implementation different concepts in C programming language like loops, Control structures, Arrays, File handling etc. This course analyses the Problem solving aspect and develop proper algorithms.

Course Outcomes: By the end of the course students will be able to:

CO1. **Implement**: Effectively use various C programming constructs to control execution based on specified conditions.
CO2. **Analysis**: of the algorithms to programs (in C language) to test and execute the programs and correct syntax and logical errors.
CO3. **Compare**: Assess the performance and suitability of different loop structures and conditional statements for solving programming problems and check the suitability of arrays, pointers and structures to formulate algorithms and programs for various problem situations.
CO4. **Integrate**: Skillfully incorporate conditional statements, loops, into structured programs.
CO5. **Create**: Develop complex algorithms and programs using arrays, functions, structures, pointers, and file processing operations to solve real-world problems.
CO6. **Demonstrate**: Showcase proficiency in applying modular programming principles and best practices to develop software solutions using the C programming language.

Course Contents

List of Experiments/ assignment

1. Basic Programs
A. C program to display hello world message.
B. C program to scan all data type variables as input and print it as output.
C. C program to perform arithmetic operations like +, -, *, /, % on two input variables.
D. C program to perform temperature conversions from Centigrade to Fahrenheit and vice versa.

2. Programs on Operators
A. C program to scan an input and perform pre and post increment operation on it and display the result.
B. C program to perform all bit wise operations.
C. C program to extract the last two digits of a given integer n, where the number of digits should be greater than 2.
D. C program to display the greatest of three numbers using a conditional operator.
E. C program to swap two numbers without using a third variable.

3. Programs on Conditional Statements
A. C program to check whether a given input integer is in between two values x and y.
B. C program to check whether a given character is a vowel or a consonant or a digit or a special symbol.
C. C program to display the nature and roots of a quadratic equation.

D. C program to perform arithmetic operations using switch statement.
E. C program to convert upper case character to lowercase and vice versa.

4. Programs on Loop Statements
A. C program to print odd numbers between specified ranges.
B. C program to display the factors of a given number and check whether it is a prime or not.
C. C program to display the sum of individual digits of a given integer raised to the power of n. Also check whether the given integer is Armstrong or not.
D. C Program to demonstrate the usage of unconditional control statements.
E. C program to display the following pattern.

```
5 4 3 2 1
 4 3 2 1
  3 2 1
   2 1
    1
```

5. Programs on Functions
A. C program to demonstrate the various categories of functions with respect to return type and number of arguments.
B. C program to find the LCM of two numbers using functions.
C. Create a header file which contains the following prototype:
i. int factorial(int) ; // non-recursive function
ii. intfactorial_rec(int); //Recursive function
iii. int prime(int) ; Use the above functions in a C program by including the above header file.
D. C program to display Pascal's triangle using functions.

6. Programs on Arrays
A. C program to read n integer values into an array and display them
B. C program to count and display the number of positive, negative, even and odd numbers in a given array of integers and also display their sum.
C. C program to find the smallest and largest numbers in an array of integers.
D. C program to perform addition, multiplication, transpose of given matrices using functions.
E. C program to check whether a given integer exists in a list of numbers and print its index value if it is present, otherwise print "No".

7. Programs on Strings
A. C program to convert upper case character to lowercase and vice versa in a given string.
B. C program to delete all vowels in a given string and display the remaining string.
C. C program to check whether a given string is palindrome or not.
D. C program that reads two integers as strings and display their sum.

8. Programs on Strings
A. C program to demonstrate the usage of at least 10 predefined string handling functions.
B. C program that implements the following user defined string handling functions
i. To find the length of the given string
ii. To copy the contents of one string to another
iii. To reverse the contents of a string
iv. To compare two strings
v. To concatenate two strings

9. Programs on Pointers and Dynamic Memory Allocation
A. C program to demonstrate the usage of pointers.
B. C program that uses dynamic memory allocation functions to add n elements and display their average.
C. C program that performs pointer arithmetic
D. C program that implements call by reference.

10. Programs on Pointers A.
C program to demonstrate the following
i. Pointers to Pointers
ii. Array of Pointers

iii. Pointer to Array
iv. Pointers to Functions

11. Programs on Structures

A. C program to access and display the members of the structure.
B. C program that demonstrates different ways to access the structure elements using pointers.

12. Programs on Files

A. C program to read the contents of a file and display it on the output screen.
B. C program to copy the contents of one file to another.
C. C program to count and display the number of characters, words and lines in a file.
D. C program to print last n characters of a file by reading file name and n value from command line.

13. C program to replace all the vowels in each string with a given character.

14. C program to perform arithmetic operations using command line arguments.

15. C program that writes the contents to a file and reads the contents from a file using structures.

Note: The actual experiment / assignments will be designed by the course coordinator. One assignment is to be done in the groups of two or three students. The assignments must meet the objectives of the course and the levels of given course outcomes.

Fundamentals of PC Software

General Course Information:

Course Code: 24ICA0103T Course Credits: 3 Type: Discipline Specific Courses (DSC) Contact Hours: 3 hours/week Mode: Lectures (L) Exam Duration: 2.5 hours	**Course Assessment Methods (Internal: 20; External: 50)** **Internal Assessment: -** Two minor examinations each of 10 marks, Class Performance measured through percentage of lectures attended (5 marks) Assignment and quiz (5 marks), and end semester examination of 50 marks. **External Examination (Marks: 50): -** The syllabus is divided into four units. For the end semester examination, nine questions are to be set by the examiner. Question No. 1 is compulsory and contains five short answer questions (2 marks each) covering entire syllabus. Rest eight more questions (each question of at least two parts) will be set by giving two questions from each unit of the syllabus. A candidate is required to attempt five questions in all by selecting one question from each unit in addition to compulsory Question No.1. All questions will carry equal marks.

Prerequisite: Knowledge of Windows, MS-word, MS-Access, MS-Excel, MS-Power point and Internet.

About the Course: This Course includes a detailed coverage of windows, MS-office and Internet. Students learn concepts of windows features, Internet and formatting of data.

Course Outcomes: By the end of the course students will be able to:

CO1. List various functions and design characteristics of PC software.
CO2. Describe various fundamental concepts related to PC software.
CO3. Apply PC software design concepts for solving problems.
CO4. Implement basic mechanics and navigation of Word, Excel & Power point spread sheets
CO5. Design Word documents, Excel workbooks, Templates, Power Point Presentation and websites.

Course Contents

Unit - I

Introduction to MS Windows: Introduction to windows, Components and its features, desktop, taskbar, files and folders start menu operations, my computer, network neighbourhood, recycle-bin, windows explorer, creating copying, moving and deleting files, setting wallpaper, changing the mouse pointer, paint, notepad, Customizing the Desktop, Files and Folders, Understanding the OLE features.

Unit-II

MS-Word& MS-Access: Introduction to word processing software and its features, Advantages of word processing, creating, saving and editing a document, Copying Text to another file. Insert, Formatting Text and Paragraph, Using the font, Dialog Box, Paragraph Formatting using Bullets and Numbering in Paragraph, Use of Smart Art, Checking Spelling, Line spacing, Margins, Space before and after paragraph, Mail merge, customizing the ribbon, Front end and Back-end application, Introduction and Features of DBMS, Create blank Database, Different views in MS-Access, Data types and formatting options.

Unit – III

MS-Excel & Power Point: Introducing Excel, creating new sheet, Saving, Opening and Printing worksheet. Entering Information: Numbers, Formula and editing data in a cell, Excel functions, using a range with SUM, Moving and copying data, Inserting and Deleting Row and Columns in the worksheet, using the format Cells Dialog box, using chart wizard to create a chart. Introduction to MS-Power Point: Introduction to Power point presentation, Slide Show, Formatting, creating a presentation, Inserting Smart Arts, Adding Objects, Applying Transitions, and Animation effects, Adding Tables, Charts and Media files.

Unit – IV

Internet: Meaning, Definitions, History, Internet protocols, TCP\IP, FTP, HTTP, URL. Internet Browsers, WWW Consortium, Search Engines. Introduction to Internet Security terminology-network security, firewall, cryptography, password, biometrics, digital signature, digital certificate. Business applications of internet, email, UseNet, newsgroup, telnet, intranet, e-ticketing, chatting. E-Banking and its Benefits: Smart Card, E-cash, Online financial services stock trading, E-broking, E-business Model, Do-it-yourself model, Made-to-order model, Information Service Model, Emerging hybrid models.

Text and Reference Books:

1. PC Software for Windows 98 made simple, R.K.Taxali, Tata McGraw Hill Publishers, 2015.
2. 3-IN-1 Microsoft Word, Power- point and Excel 2010: A Complete Guide,Gloria Maumee, Create space Independent Publishing Platform, First Edition 2016.
3. Internet: The complete Reference, Margaret Levine Young, Ist edition, 1999.
4. PC Software Made Simple, Jain, BPB Publications,2002.
5. Word 2002 from A to Z, Stephen L. Nelson, FireWall Media, 2009.

Fundamentals of PC Software Lab

General Course Information:

Course Code: 24ICA0103P Course Credits: 1 Type: Discipline Specific Courses (DSC) Contact Hours: 2 hours/week Mode: Lab Exam Duration: 3 hours	**Course Assessment Methods (Internal: 10; External: 20)** The Internal and External assessment is based on the level of participation in Lab sessions and the timely submission of Lab experiments / assignments, the performance in Viva-Voce, the quality of the lab file and ethical practices followed. The Internal examination is conducted by the Course Coordinator. The External examination is conducted by External Examiner appointed by the Controller of Examination in association with the Internal Examiner appointed by the Chairperson of the Department.

Prerequisite: Students are expected to have basic knowledge of Windows, MS-word, MS-Access, MS-Excel, MS-Power point and Internet.

About the Course:A study of the subject matter presented in this course will enable the student to become familiar with concept of Windows, MS-Word. MS-Excel, MS-PowerPoint, MS-Access, Internet.

Course Outcomes: By the end of the course students will be able to:

CO1 **Implement**: Implement Practical knowledge and use of the Windows operating system.
CO2 **Demonstrate:** Creating word documents for office use**.**
CO3 **Format:** Techniques and presentation styles.
CO4 **Integrate:** Use of Basic functions and formulas.
CO5 **Create:** Interactive window, word documents, excels workbooks, power point presentation and website.

Course Contents

List of Experiments/ assignments

1. Install the windows and practice different exercises.
2. Create Edit and Format documents (Prepare a Student Resume) in MS Word.
3. Insert Header and footer in MS Word.
4. Create Macro in MS Word.
5. Design and Create tables to store data in MS-Access.
6. Design and implement forms in MS Access
7. Design queries in MS Access.
8. Use Formula and functions in MS Excel.
9. Create and edit worksheet in MS Excel.
10. Create Charts, pivot chart and pivot table in MS Excel.
11. Manipulate and enhance slides in MS Powerpoint
12. Animations and sounds Sound effect in MS Power point.
13. Create a Power Point Presentation for a given theme.
14. Create a Website using HTML tags.

Note: The actual experiment / assignments will be designed by the course coordinator. One assignment to be done in the groups of two or three students. The assignments must be met the objectives of the course and the levels of given course outcomes. The list of assignments and schedule of submission will be prepared by course coordinator in the beginning of the semester.

Minor Courses:

Essentials of Computer

General Course Information

Course Code:24MIC0121T Course Credits: 2 Hours/Week: 2 Type: Professional Core Category: Theory Mode: Lectures (L) Examination Duration: 2 Hours	**Course Assessment Methods:** **Max. Marks: 50 (Internal: 15; External: 35)** Internal Assessment: - Two minor examinations each of 10 marks, Class Performance measured through percentage of lectures attended (3 marks) Assignment and quiz (2 marks) **External Examinations: -** The end semester examination will be of 35 marks. The examiner is required to set five questions in all.. The first Question will be compulsory consisting of five short questions covering entire syllabus consisting three marks each. In addition to this four more questions (each questions may be of two part) will be set consisting of two questions from each unit. The students is required to attempt three question in all selecting one questing from each unit consisting of 10 marks each in addition to Question No.1 is compulsory.

Prerequisite: Basic understanding of computer science fundamentals.

About the Course:
This course introduces basic constructs of fundamental of computers and programming language like algorithms, conversion of algorithms to programs etc. By studying this course students will get to know about C programming language with its various programming paradigms like branching, looping, arrays.

Course Outcomes: By the end of the course students will be able to:
CO1. List: various functions and characteristics of computer systems
CO2. Explain: basic design and types of computer systems.
CO3. Analyse: algorithms to programs (in C language) to test and execute the programs and correct syntax and logical errors.
CO4. Apply: various constructs of a programming language viz., conditional, iteration and recursion.

Course Content

Unit-I

Introduction to Computers: Introduction, Characteristics of Computers, Block diagram of computer. Types of computers and features, Minicomputers, Micro Computers, Mainframe Computers, Super Computers. Types of Programming Languages (Machine Languages, Assembly Languages, High Level Languages). Types of Memory (Primary and Secondary) RAM, ROM, PROM, EPROM. Secondary Storage Devices (FD, CD, HD, Pen drive) I/O Devices (Scanners, Plotters, LCD, Plasma Display) Number Systems, Introduction to Binary, Octal, Hexadecimal system Conversion.

Unit-II

Programming fundamentals: Planning the computer program, Algorithms: construct, characteristics, Flowchart: symbols, Decision tables: types of decision tables, structured programming.
C language: Data Types, Operators and Expressions: Character set, C identifiers and keywords; variables and constants; basic data type arithmetic, relational, logical and bit-wise operators; increment decrement and ternary operators; typedef, struct, enumerated data types; type conversion.
Control Flow: Various if statement; while, do-white and for loops, switch, break and continue statement; go to statement. **Arrays**: Declaration, initialization and operations on arrays.

Text and Reference:

1. Fundamentals of computers-By P.K.Sinha.
2. Fundamentals of computers-By V.Rajaraman B.P.B Publications

3. Let’s C-Yashwant Kanetkar
4. Programming in C-Balguruswamy
5. The C programming Lang., Pearson Ecl - Dennis Ritchie

Multidisciplinary Course:

Fundamental of Computer Science

General Course Information:

Course Code: 24MDC0110T Course Credits: 3 Type: Discipline Specific Courses (DSC) Contact Hours: 3 hours/week Mode: Lectures (L) Exam Duration: 2.5 hours	**Course Assessment Methods (Internal: 25; External: 50)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (3 marks) Assignment and quiz (2 marks). **External Examinations (50 marks): -**. The syllabus is divided into four units. For the end semester examination, nine questions are to be set by the examiner. Question No.1 is compulsory and contains five short answer questions covering entire syllabus carrying 2 marks each. Rest eight more questions (each question of at least two parts) will be set by giving two questions from each of the unit of the syllabus carrying 10 marks each. A candidate is required to attempt five questions in all by selecting one question from each of unit in addition to compulsory Question No.1.

Prerequisite: Basic computer Literacy, Basic knowledge of protecting data from malware.

About the Course:

This course has been designed with an aim to provide students with an overview of the concepts and fundamentals of computers, information & Communication technology and GUI based operating system. This course describes the data types and its digital representation, security issues and various ways of user's well-being as well as Green IT.

Course Outcomes: By the end of the course students will be able to:

CO1 Lists the key concepts relating to ICT, computers, devices and software.
CO2 Describe the various types of GUI based operating system and effectively work on the system.
CO3 Apply file management and efficiently organize files and folders, compress and extract large files.
CO4 Design& implement the various policies of protecting data and devices from malware.
CO5 Analyze to enhance computer accessibility by using voice recognition software and examine the way of user well-being to promote green IT.

Course Contents

Unit - I

Computers and Devices: Information and Communication Technology (ICT), ICT services: Internet services, mobile technology, office productivity applications. Types of computers: desktops, laptops, tablets and main types of devices: smartphones, media players, digital cameras. Identify the main types of integrated and external equipment like: printers, screens, scanners, keyboards, mouse/track pad, webcam, speakers, microphone, docking station and common input/output ports: USB, HDMI.

Unit-II

GUI Based Operating System: Operating System: LINUX, WINDOWS. User Interface: Task Bar, Icons, Menu, applications of operating System: Settings of system Date and Time, Changing Display Properties: adding and removing printers. File and Directory Management: Creating and renaming of files and directories and common utilities.

Unit – III

Data types and its digital representation: Binary numeral system: two binary numbers, representation of real numbers, 2's complement operations: Fixed-point and floating-point numbers. Binary codes: BCD vs. ASCII codes. Error codes: Hamming distance, Parity codes, CRC codes.

Unit – IV

Security and Well-Being: password policies: create with adequate length, character mix, change regularly. Define firewall and outline its purpose. Software updates: anti-virus, application, operating system software. Malware and its type: virus, worm, Trojan, spyware. Health and Green IT: Ways of user's well-being: take regular breaks, ensure appropriate lighting and posture. Device energy saving: turning off, adjusting automatic shutdown, backlight and sleep mode settings. Enhancing accessibilities: voice recognition software, screen reader, screen magnifier, on-screen keyboard, high contrast.

Text and Reference Books:

1. J. Glenn Brookshear,"Computer Science: An Overview", Addision-Wesley, Twelfth Edition, 2014.
2. PC Software for Windows 98 made simple, R.K.Taxali, Tata McGraw Hill Publishers, 2015.
3. Fundamental ofdigital systems. Thomas L. Floyd. Prentice Hall. 9ª Ed. 2006.
4. Digital Design and Computer Architecture. David Money Harris y Sarah L. Harris. Morgan Kaufmann. 2007.
5. Fundamental of digital Logic gates and computations. M. Morris Mano y Charles R. Kime. Prentice Hall. 3ª Ed. 2005.

Skill Enhancement Course:

PC Hardware and Networking

General Course Information:

Course Code: 24SEC0110T Course Credits: 2 Type: Skill Enhancement **(SEC)** Contact Hours: 2 hours/week Mode: Lectures (L) Exam Duration: 2 hours	**Course Assessment Methods (Internal: 15; External: 35)** Two minor examinations each of 10 marks, Class Performance measured through percentage of lectures attended (3 marks) Assignment and quiz (2 marks), and end semester examination of 35 marks. The syllabus is divided into Two units. For the end semester examination, five questions are to be set by the examiner. Q. No. 1 is compulsory and contains five short questions covering entire syllabus consisting of 3 marks each. Rest four questions are set by giving two questions from each of the unit of the syllabus. A candidate is required to attempt three questions in all selecting one question from each unit consisting of 10 marks each in addition to compulsory Q.No.1

Prerequisite: Basics of Computer.

About the Course: PC hardware & networking is the basic and essential course for every graduate in computer science. This course introduces various hardware components of computer like ROM, Hard Disks, SMPS, UPS etc. and various functionalities performed by these components. It also includes the various networking services with the help of different connectors provided in real world.

Course Outcomes: By the end of the course students will be able to:

CO1. **Describe:** various components of computer and their functionalities.
CO2. **Demonstrate:** the use of various hardware components and their operations.
CO3. **Apply:** various transmission media in network communication.
CO4. **Implement:** how to establish the connection and use it using IP addresses.
CO5. **Define:** the internet and intranet with their services available in now a day.
CO6. **Compare :** the connectors with respect to efficiency of the required operations for solving real live problem.

Course Contents

Unit - I

Introduction to computer hardware: Peripheral devices of a computer system, Add On cards: network interface card, sound card and graphics card, functions of various parts of a PC, SMPS, UPS, CMOS and its types, Mother Board, Types of Ports, Hard Disk and Types of Hard Disk, RAM, Cabinet, Processor and its types.
BIOS: Introduction: Linker, loader and assembler, Connecting & disconnecting computer peripherals and components Mouse, Keyboard, Monitor, Printer, Memory: Types of memory. Cables: Types of Cables, cable design. Drivers: Types of drivers: Device drivers, LAN drivers, sound drivers, graphics drivers.

Unit – II

Introduction to Computer Networks: Types of Computer Networks and their topologies. Transmission media - wired and wireless. Network hardware components: connectors, transceivers & media converters, repeaters, network interface cards and PC cards, repeaters, bridges, switches, routers, gateways, connecting ports.
Introduction to the Internet: concepts of Internet and Intranet; IP addresses, DNS; Internet Services; E-mail, File transfer and FTP, World Wide Web and HTTP. Web Browsers, Search Engines, Uniform Resource Locator (URL),Web Servers. Internet Connections: Dialup, Leased line, Modems, DSL service, Internet Service Provider.

T**ext and Reference Books:**

1. B.Govindarajalu, IBM PC and Clones: Hardware, Troubleshooting and Maintenance, McGraw Hill Education, 2002
2. Digital Logic and Computer Design, M. Morris Mano, PHI, 2000
3. Computer Communications and Networking Technologies, Michael A. Gallo, William M. Hancock, CENGAGE Learning.
4. Foundations of Computing, P.K. Sinha, BPB.

PC Hardware and Networking Lab

General Course Information:

Course Code: 24SEC0110P Course Credits: 1 Type: Skill Enhancement Course (SEC) Contact Hours: 2 hours/week Mode: Labs Exam Hours: 3 Hours	**Course Assessment Methods (Internal: 10; External: 15)** The Internal and External assessment is based on the level of participation in Lab sessions and the timely submission of Lab experiments/assignments, the performance in Viva-Voce, the quality of the lab file and ethical practices followed. The Internal examination is conducted by the Course Coordinator. The External examination is conducted by External Examiner appointed by the Controller of Examination in association with the Internal Examiner appointed by the Chairperson of the Department.

Prerequisite: Students are expected to have basic knowledge of computer device and hardware.

About the Course:

The objectives of this course are to make:

- Understand basic concept of computer hardware, installation of its components like RAM, Hard Drive, SMPS, UPS, graphic card etc.
- Installation of system software like Operating system and application software like MS Office, printer etc.
- It also includes various network design topology, networking components and devices.

Course Outcomes: By the end of the course students will be able to:

CO1. **Implement:** design of cables and installation of components
CO2. **Analyse:** functionality of hardware and networking components
CO3. **Compare:** various devices and their functionalities like hub and switches
CO4. **Integrate:** PC components for working of device
CO5. **Create:** Network topology to understand various types of networks.
CO6. **Demonstrate:** the functionality and components of a PC.

Course Contents

List of Experiments/ assignments

1. To identify and check various components of a PC.
2. Installation/assembling of various PC components.
3. To learn handling and configuration of various hardware like RJ-45 connector, networking cables, crimping tools etc.
4. Design Cross and straight cable for networking.
5. Install windows operating system.
6. Install Linux (Ubuntu) operating system.
7. Install and Configure Dual OS Installation.
8. Install application software on PC (MS-Office)
9. Install and Configure a DVD Writer and a Blu-ray Disc writer and recording DVD and Blu-ray disk.
10. Printer Installation and Servicing and troubleshoot
11. Configuring Hub and switches.
12. Connect the devices in a LAN network.
13. Running and using basic network commands like ping, trace etc.
14. Configure network topology like Star topology using hub.
15. Configure network topology like Star topology using switch.

Note: The actual experiment / assignments will be designed by the course coordinator. One assignment to be done in the groups of two or three students. The assignments must be met the objectives of the course and the levels of given course outcomes. The list of assignments and schedule of submission will be prepared by course coordinator in the beginning of the semester.

Discipline Specific Course:

Database Technologies-1

General Course Information:

Course Code: 24ICA0201T Course Credits: 3 Type: Discipline Specific Courses (DSC) Contact Hours: 3 hours/week Mode: Lectures (L) Exam Duration: 2.5 hours	**Course Assessment Methods (Internal: 20; External: 50)** **Internal Assessment: -** Two minor examinations each of 10 marks, Class Performance measured through percentage of lectures attended (5 marks) Assignment and quiz (5 marks), and end semester examination of 50 marks. **External Examination (Marks: 50): -** The syllabus is divided into four units. For the end semester examination, nine questions are to be set by the examiner. Question No. 1 is compulsory and contains five short answer questions (2 marks each) covering entire syllabus. Rest eight more questions (each question of at least two parts) will be set by giving two questions from each unit of the syllabus. A candidate is required to attempt five questions in all by selecting one question from each unit in addition to compulsory Question No.1. All questions will carry equal marks.

Prerequisite: Elementary Programming skills and knowledge of basic database concepts and database applications such as MS-Access.

About the Course: This course includes detailed coverage of principles of database design and models.

Course Outcomes: By the end of the course students will be able to:

CO1. **List:** fundamental of Database Management System.
CO2. **Select:** principles for basic Database Modelling.
CO3. **Apply:** basic SQL commands for designing queries on Relational Databases.
CO4. **Implement:** various operations of Relational Algebra.
CO5. **Apply:** basic E-R Modelling for E-R diagram on a real-world database application.
CO6. **Compare:** various types of Normalization for real-world database

Course Contents

Unit - I

Basic Introduction: Data, Information, Records, Files and Database. Characteristics of Database Management System. DBMS over File Processing System. Advantages and Disadvantages of DBMS. Database Users and various types of DBMS users. Database Administrator (DBA) and responsibilities of DBA. Schema and Instance. Views of Database. Physical and Logical Data Independence. Database Languages.

Unit – II

DBMS Architecture: 1-Tier, 2-Tier and 3-Tier Database Architecture. Data Models: Hierarchical, Network and Relational Data Models. Entity-Relationship Model: Entity, Entity Sets, Entity Type, Attributes: Type of Attributes, Cardinality, Degree, Domain; Keys: Super Key, Candidate Key, Primary Key, Foreign Key.
ER Diagram: Symbolic Notations for Designing ER Diagram.

Unit – III

Introduction of SQL: characteristics of SQL, Basic Structure, DDL Commands, DML Commands, DQL Commands, TCL Commands. Useful Relational Operators, Aggregate Functions, SET operations.
Relational Algebra: Basic Operations: Select, Project, Union, Intersection, Set Difference, and Cartesian Product, Join. Relational Algebra vs. Relational Calculus.

Unit – IV

Relational Database Design: Functional Dependency, Types of Functional Dependency, Introduction to Normalization, Anomalies of unnormalized database, Normal Form: 1st Normal Form, 2nd Normal Form, 3rd Normal Form. Denormalization.
Distributed Database: Allocation, Replication and Data Fragmentation

T**ext and Reference Books:**

1. Database System Concepts, Sixth edition, Abraham Silberschatz, Henry F. Korth, S. Sudarshan, McGraw-Hill-2010.
2. Database Systems: Models, Languages, Design and Application, Ramez Elmasri, Pearson Education 2014.
3. C. J. Date, A. Kannan and S. Swamynathan, "An Introduction to Database Systems", 8th, Pearson Education, 2006.
4. Ramez Elmasri and Shamkant B. Navathe, "Fundamentals of Database Systems", 7thEdition, Pearson/Addisionwesley, 2016.
5. Raghu Ramakrishnan, "Database Management Systems", Third Edition, McGraw Hill, 2003.

Database Technologies-I Lab

General Course Information:

Course Code: 24ICA0201P Course Credits: 1 Type: Discipline Specific Courses (DSC) Contact Hours: 2 hours/week Mode: Lab Exam Duration: 3 hours	**Course Assessment Methods (Internal: 10; External: 20)** The Internal and External assessment is based on the level of participation in Lab sessions and the timely submission of Lab experiments / assignments, the performance in Viva-Voce, the quality of the lab file and ethical practices followed. The Internal examination is conducted by the Course Coordinator. The External examination is conducted by External Examiner appointed by the Controller of Examination in association with the Internal Examiner appointed by the Chairperson of the Department.

Pre-requisite:
Students are expected to have basic knowledge of problem solving and MS Access

About the Course: This course is designed to implement concepts of database and SQL. Students will analyze the real-life problem and execute DDL, DML , DCL commands and SQL query processing to solve the problem.

Course Outcomes: By the end of the course students will be able to:

CO1. **Implement:** database using DDL commands.
CO2. **Analyse:** integrity constraints on a database using SQL commands.
CO3. **Compare:** design of a relational database.
CO4. **Design:** a database for various given schemas.
CO5. **Execute:** Lab assignments for various problems.
CO6. **Demonstrate:** ethical practices, self-learning and team spirit.

List of Experiments/ assignments

1. Use oracle software and login with valid user id and password.
2. Creation of database schema having different fields and data types.
3. Adding various constraints using DDL commands.
4. Alter the schema of the existing database.
5. Delete the schema of the database.
6. Add records in the database using DML command.
7. SQL query for various views of the database.
8. Update and delete existing records in the database using DML command
9. Using SQL queries for various aggregate function.
10. Using SQL queries Ordering and grouping of records on the database.
11. Using SQL queries for various SET operations on relational database.
12. Using SQL queries for logical operations on relational database.
13. Using various DCL commands on the database.
14. Using various TCL commands on the database.
15. Using SQL queries for creating Join operations.

Note: The actual experiment / assignments will be designed by the course coordinator. One assignment to be done in groups of two or three students. The assignments must meet the objectives of the course and the levels of given course outcomes. The list of assignments and schedule of submission will be prepared by course coordinator in the beginning of the semester.

Object Oriented Programming using C++

General Course Information:

Course Code: 24ICA0202T Course Credits: 3 Type: Discipline Specific Courses (DSC) Contact Hours: 3 hours/week Mode: Lectures (L) Exam Duration: 2.5 hours	**Course Assessment Methods (Internal: 20; External: 50)** **Internal Assessment: -** Two minor examinations each of 10 marks, Class Performance measured through percentage of lectures attended (5 marks) Assignment and quiz (5 marks), and end semester examination of 50 marks. **External Examination (Marks: 50): -** The syllabus is divided into four units. For the end semester examination, nine questions are to be set by the examiner. Question No. 1 is compulsory and contains five short answer questions (2 marks each) covering entire syllabus. Rest eight more questions (each question of at least two parts) will be set by giving two questions from each unit of the syllabus. A candidate is required to attempt five questions in all by selecting one question from each unit in addition to compulsory Question No.1. All questions will carry equal marks..

Prerequisite: Elementary Programming skills in C.

About the Course: The course is intended to imbibe the object-oriented skills in students by making them learn basic OOPS terminology and concepts such as encapsulation, polymorphism, data binding, data hiding, inheritance, exception handling, event handling, function overloading, constructors, destructors, types of functions like friend functions, virtual functions, file handling, stream I/O, etc. The course shall equip students with various data abstraction techniques, data presentation methods and data handling methods. The course shall be beneficial for students as this stream deals with application arena of technologies developed or under development.

Course Outcomes: By the end of the course students will be able to:

CO1. **List:** object-oriented terms describing the OOPS concepts and the C++ language.
CO2. **Illustrate:** significance of OOPS in terms of space, time and algorithmic complexity**.**
CO3. **Apply:** C++ programming constructs to create classes, objects, functions, constructors etc.
CO4. **Implement:** OOPS concepts like polymorphism, inheritance, exception handling, data abstraction,
data hiding, etc. in C++ compiler.
CO5. **Justify:** correctness of OOPS programs by using constructs of C++ language.
CO6. **Compare:** effectiveness, performance and time-constraint applications of C++ programs.

Course Contents

Unit - I

Introduction to Object-Oriented Programming Concepts- Structural Programming Vs Procedural Programming Vs Object -Oriented Programming Vs Object-Based Programming Vs Pure Object-Oriented Programming. Object-Oriented Terminology: object, class, instance, attribute, method, encapsulation, data hiding, data abstraction, data binding: static and dynamic, inheritance, polymorphism, etc. Create a C++ class defining an object of the class created.

Unit – II

C++ programming Language Constructs- data types**,** conditional statements-if. If- else, if-else if ladder, switch-case, looping statements- for, while, do-while, arithmetic operators, relational operators, logical operators, new operator, this pointer, access modifiers, Arrays and Strings, function- friend function, virtual function, function overloading, function overriding, constructor, constructor overloading, destructor, defining data members and member functions of a class, define classes to show the C++ concepts like polymorphism, inheritance, data abstraction, data binding, method overloading, method overriding, constructors, destructors, etc., implement popular algorithms using the constructs of C++ and state their complexities.

Unit – III

Exception Handling- try-catch block, multiple try blocks with one catch block, try with multiple catch blocks, multiple try with multiple catch blocks, nested try blocks, throw, throws, user defined exceptions, handling user-defined exceptions, creating C++ programs to demonstrate exception handling (built-in exceptions and user-defined exceptions).

Unit – IV

File Handling and Event Handling- Stream I/O, Stream Classes, Stream Errors, File handles, File I/O errors, File I/O with member functions, Disk File-I/O with Streams, multi-file programs, Command-line arguments. Event Handling-mouse event handling, keyboard event handling, window creation, menus, buttons, text boxes, check boxes, list boxes. Creating C++ programs to demonstrate file handling and event handling and compare their efficiencies.

Text and Reference Books:

1. Bjarne Stroustrup. 2013. The C++ Programming Language (4th. ed.). Addison-Wesley Professional.
2. Robert Lafore. 2001. Object Oriented Programming in C++ (4th. ed.). Sams, USA.
3. Szalwinski, Chris. 2016. Introduction to Object-Oriented Programming Using C++14.
4. Bala Guruswamy, E. 2020. Object Oriented Programming with C++ (8^{th}. ed.). McGraw-Hill Education (India) Pvt Limited
5. Harvey M. Deitel and Paul J. Deitel. 2002. C++ How to Program, Fourth Edition (4th. ed.). Prentice Hall Professional Technical Reference.

Object Oriented Programming using C++ Lab

General Course Information:

Course Code: 24ICA0202P Course Credits: 1 Type: Discipline Specific Courses (DSC) Contact Hours: 2 hours/week Mode: Lab Exam Duration: 3 hours	**Course Assessment Methods (Internal: 10; External: 20)** The Internal and External assessment is based on the level of participation in Lab sessions and the timely submission of Lab experiments / assignments, the performance in Viva-Voce, the quality of the lab file and ethical practices followed. The Internal examination is conducted by the Course Coordinator. The External examination is conducted by External Examiner appointed by the Controller of Examination in association with the Internal Examiner appointed by the Chairperson of the Department.

Prerequisite:
Students are expected to have basic knowledge of problem-solving using programming constructs of C.

About the Course:
The course includes programming in C++ using an IDE such as visual studio, Borland C++, Code Blocks, Microsoft visual C++ etc. The course would help MCA students to get acquainted with the OOPS concepts (polymorphism, inheritance, data binding, data abstraction, encapsulation, etc) implementation in C++ using constructs such as conditional statements, looping statements, functions, arrays, strings, templates, built-in libraries, file I/O, events, windows, etc. The course also helps solve real-world problems.

The objectives of this course are to:

- Understand basic concepts of programming language, concepts of loops, reading a set of data, stepwise refinement, function, control structures, arrays, file handling
- Analyze the real-life problem and write a program in C++ language to solve the problem.

Course Outcomes: By the end of the course students will be able to:

CO1. **Implementation of OOPS concepts (encapsulation, inheritance, polymorphism, data binding, data abstraction, file handling, event handling, exception handling) using C++.**
CO2. **Analysis of output of program for correctness of solution provided by the C++ compiler against the given problem.**
CO3. **Compare algorithms against time, space and complexity for search of a better one.**
CO4. **Integrate C++ language constructs with real-world assignments for a robust and efficient solution.**
CO5. **Create lab record for the given assignments with problem statement, design of solution and result displayed.**
CO6. **Demonstrate ethical practices while solving problems individually or in groups.**

Course Contents

List of Experiments/ assignments

1. To become well versed in C++ compiler IDE such as Microsoft Visual C++, Code Blocks, Borland C++ etc. by writing a program to create a class in C++ and create its object to initialize data members of class using constructor of the class.
2. Write 2-3 assignments on functions, function overloading.
3. Write 2-3 assignments on operator overloading, constructor overloading.
4. Write 2-3 assignments on using objects of a class as arguments to member functions of a class as copy constructors.
5. Write 3-4 assignments on inheritance, multiple inheritance, hierarchical inheritance, multilevel inheritance, function overriding,
6. Write 3-4 assignments on polymorphism, compile-time and run-time polymorphism, data binding, data abstraction, exception handling.
7. Write 3-4 assignments on arrays, strings, searching. Sorting.
8. Write 2-3 assignments on event handling, mouse events, keyboard events, create a window, creating menus, buttons, radio buttons, combo boxes, list boxes.
9. Write 2-3 assignments on reading data from files, writing data to files, appending data to files, sorting data of a file.
10. Write 2-3 assignments on creation of class and function templates.
11. Write 2-3 assignments on creating header files, using the created header files.
12. Write 2-3 assignments on multiple-file programs, releasing the resources once the files are closed or the session ends.
13. Write 2-3 assignments on using I/O streams.
14. Write 2-3 assignments on using libraries of C++ like AWT, STDC++, io, etc.
15. Write 2-3 assignments on data handling in C++.

Note: The actual experiment / assignments will be designed by the course coordinator. One assignment to be done in the groups of two or three students. The assignments must be meet the objectives of the course and the levels of given course outcomes. The list of assignments and schedule of submission will be prepared by course coordinator in the beginning of the semester.

Data Structures & Algorithms

General Course Information:

Course Code: 24ICA0203T Course Credits: 3 Type: Discipline Specific Courses (DSC) Contact Hours: 3 hours/week Mode: Lectures (L) Exam Duration: 2.5 hours	**Course Assessment Methods (Internal: 20; External: 50)** **Internal Assessment: -** Two minor examinations each of 10 marks, Class Performance measured through percentage of lectures attended (5 marks) Assignment and quiz (5 marks), and end semester examination of 50 marks. **External Examination (Marks: 50): -** The syllabus is divided into four units. For the end semester examination, nine questions are to be set by the examiner. Question No. 1 is compulsory and contains five short answer questions (2 marks each) covering entire syllabus. Rest eight more questions (each question of at least two parts) will be set by giving two questions from each unit of the syllabus. A candidate is required to attempt five questions in all by selecting one question from each unit in addition to compulsory Question No.1. All questions will carry equal marks.

Prerequisite: Elementary Programming skills in C.

About the Course: Data Structure and Algorithms is a core and an essential course for every graduate in Computer Science and Engineering. This course introduces data structures like arrays, linked lists, trees and graphs etc. and various operations to be implemented on these data structures for solving real world problems. It includes various sorting and searching algorithms as well. Further, it incorporates complexity analysis of algorithms implemented on various data structures.

Course Outcomes: By the end of the course students will be able to:

CO1. Describe: various types of data structures and operations on them.
CO2. Demonstrate: use of various data structures and the related operations.
CO3. Apply: data structure to solve computational problems.
CO4. Compare: the suitability of alternative data structures and their operations.
CO5. Defend: solutions with respect to effective storage of data and efficiency of the required operations for solving real world problems.

Course Contents

Unit - I

Introduction to data structures and their types: Abstract data types, Linear lists: Arrays and linked lists: memory representations, implementing operations like traversing, searching, inserting and deleting etc. Applications of arrays and linked lists. Representing sets and polynomials using linked lists.

Unit - II

Stack and Queue: Static and linked implementations, Operations and Applications. Circular queues, Tress, Binary trees and related terminology, Treetraversals (Recursive), Threaded Binary Trees, Binary Search Trees implementation and operations, Priority queues.

Unit - III

Height Balanced or AVL trees and B trees: Graph definitions and related terminology, memory representations and related operations (traversal, insertion, deletion, search), Path Matrix, Warshall's Shortest path algorithm Hashing, Hash tables, hash function and collision resolution.

Unit – IV

Sorting algorithms: Sequential and binary search, Bubble sort, Selection sort, Insertion sort, Quick sort, Merge sort, Count sort, Heap sort, Comparison of searching and sorting techniques based on their complexity analysis, Time and space complexity of algorithms: asymptotic analysis, Big O, Omega, Theta notations.

Text and Reference Books:

1. Aho, A.V., Ullman,J.D., and Hopcroft, J.E., *Data Structures and Algorithms*, Addison-Wesley, 1983.
2. LangsamYedidyah, Augenstein J Moshe, Tenenbaum M Aaron, *Data Structures using C and C++*,3rdedition, PHI,2009.
3. Cormen, T.H., Leiserson, C.E.,Rivest,R.L.andStein,C.,*Introduction to Algorithms*, MITPress,2009.
4. RobertL. Kruse,*Data Structure and Program Design in C*, Pearson Education India, 2007.
5. Weiss, M.A. *Data Structures and Algorithm Analysis in C++*, Addison-Wesley,2007.
6. Sahni, S.*Data Structures, Algorithms, and Applications in C++*, WCB/McGraw-Hill,2001.

Data Structures & Algorithms Lab

General Course Information:

Course Code: 24ICA0203P Course Credits: 1 Type: Discipline Specific Courses (DSC) Contact Hours: 2 hours/week Mode: Lab Exam Duration: 3 hours	**Course Assessment Methods (Internal: 10; External: 20)** The Internal and External assessment is based on the level of participation in Lab sessions and the timely submission of Lab experiments / assignments, the performance in Viva-Voce, the quality of the lab file and ethical practices followed. The Internal examination is conducted by the Course Coordinator. The External examination is conducted by External Examiner appointed by the Controller of Examination in association with the Internal Examiner appointed by the Chairperson of the Department.

Prerequisite: Students are expected to have basic knowledge of problem solving and programming constructs and C language.

About the Course: This lab course involves implementation of basic and advanced data structures and various operations on these data structures. The objective of the lab course is to train the students to solve the problems related to data structures and choose the appropriate data structure for solving computational problem efficiently.

Course Outcomes: By the end of the course students will be able to:

CO1. **Implement:** various data structures and the related operations.

CO2. **Analyze:** space and time complexity of algorithms.
CO3. **Compare:** appropriateness and the efficiency of the operations implemented.
CO4. **Integrate:** knowledge of data structures to solve real world problems related to data structure.
CO5. **Create:** records for the given assignments with problem definition, design and conclusions.
CO6. **Demonstrate:** ethical practices while solving problems individually or in groups.

List of experiments/assignments

1. Two assignments related to creating and manipulating matrices and linear lists.
2. Two assignments associated with linked list, operations on linked lists and their applications.
3. Two assignments on array and linked implementation of stacks.
4. Two assignments on array and linked implementation of queues.
5. Two assignments on trees and their applications.
6. Two assignments on graphs and their applications.
7. Five assignments on different searching and sorting methods along with their complexity analysis.
8. One assignment on challenging problems on data structures to be given in groups.

Note:

The actual experiments/assignments will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.

Minor Courses:

Operating System

General Course Information

Course Code: 24MIC0221T Course Credits: 2 Hours/Week: 2 Type: Professional Core Category: Theory Mode: Lectures (L) Examination Duration: 2 Hours	Course Assessment Methods: **Max. Marks: 50 (Internal: 15; External: 35)** Internal Assessment: - Two minor examinations each of 10 marks, Class Performance measured through percentage of lectures attended (3 marks) Assignment and quiz (2 marks) **External Examinations (Marks: 35):-**The end semester examination will be of 35 marks. The examiner is required to set five questions in all.. The first Question will be compulsory consisting of five short questions covering entire syllabus consisting three marks each. In addition to this four more questions (each questions may be of two part) will be set consisting of two questions from each unit. The students is required to attempt three question in all selecting one questing from each unit consisting of 10 marks each in addition to compulsory Question No.1.

Prerequisite: Basic understanding of computer science fundamentals, including programming concepts, data structures and familiarity with computer architecture and hardware components.

About the Course: This course provides an in-depth understanding of operating systems, covering concepts like process management, CPU scheduling, deadlock handling , focusing on various Operating System types and techniques.

Course Outcomes: By the end of the course students will be able to:

CO1. List: various functions and design characteristics of operating systems.

CO2. Explain: fundamental concepts of operating systems.

CO3. Apply: operating system design concepts for solving problems regarding scheduling, memory management, disk management and deadlocks etc.

CO4. Analyse: the issues related to various operating systems.

CO5. Design: solutions for the memory and process management problems.

Course Content

Unit-I

Introductory Concepts: Operating systems functions and characteristics, Types of Operating systems: Batch operating system, Time-sharing OS, Distributed operating system, Real-time systems. Operating system services and systems calls. System programs, Operating System Structure.

Unit-II

Processes: Process concept, Process Control Block, Operations on processes, cooperating processes. CPU scheduling: Levels of Scheduling, scheduling criteria, multiple processor scheduling.
Deadlock: Deadlock characterization, Methods for handling deadlocks, Deadlock prevention, Deadlock avoidance, Deadlock detection, Recovery from deadlock

Text and Reference:

1. Silberschatz, Peter B. Galvin and Greg Gagne, *Operating System Concepts*, 8th Edition, WileyIndian Edition, 2010.

2. Andrew S Tanenbaum, *Modern Operating Systems,* Third Edition, Prentice Hall India, 2008.

3. D.M. Dhamdhere, *Operating Systems*, 2nd edition, Tata McGraw Hill, 2010.

Multidisciplinary Courses:

Programming for Problem Solving using C

General Course Information:

Course Code: 24MDC0210T Course Credits: 3 Type: Multidisciplinary Course (MDC) Contact Hours: 3 hours/week Mode: Lectures (L) Exam Duration: 2.5 hours	**Course Assessment Methods (Internal: 25; External: 50) Internal Assessment: -** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (3 marks) Assignment and quiz (2 marks), and end semester examination of 50 marks. **External Examination (Marks: 50): -** The syllabus is divided into four units. For the end semester examination, nine questions are to be set by the examiner. Question number one is compulsory and contains five short answer questions of 2 marks each covering entire syllabus. Rest eight more questions (each question of at least two parts) will be set by giving two questions from each of the unit of the syllabus. A candidate is required to attempt five questions in all by selecting one question from each of unit in addition to Question No.1is compulsory.

Prerequisite:
Students are expected to have basic knowledge of problem solving and programming constructs.

About the Course: This course has been designed to provide students with an overview of the basic concepts of programming language, concepts of loops, reading a set of data, stepwise refinement, Function, Control structures, Arrays, File handling. This course describes efficient algorithms in C language to solve real-life problems.

Course Outcomes: By the end of the course students will be able to:

CO1. List: simple algorithms for arithmetic and logical problems.
CO2. Select: efficient algorithms for solving a problem.
CO3. Apply: various constructs of a programming language viz., conditional, iteration and recursion.
CO4. Implement: algorithm in 'C' language for Structure, Unions and Pointers.
CO5. Apply**:** simple data structures in solving problems and file handling concept.

Course Contents

Unit – I

Introduction to Programming: The Basic Model of Computation, Algorithms, Flow-charts, Programming Languages, Need for computer programming languages, Compilation, Linking and Loading, Testing and Debugging, Documentation.
Problem solving and algorithms for Problem Solving: 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.
Introduction to 'C' Language: History of C, 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.

Unit - II

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 .
Arrays: 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

Unit – III

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 multiple source files: extern and static

Unit – IV

Structures and Unions: Structure variables, initialization, structure assignment, nested structure, structures and functions, structures and arrays: arrays of structures, structures containing arrays, unions.
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.
File Processing: Concept of Files, File opening in various modes and closing of a file, reading from a file, writing onto a file

T**ext and Reference Books:**

6. Byron S Gottfried "Programming with C" Second edition, Tata McGrawhill, 2007 .
7. R.G. Dromey, "How to solve it by Computer", Pearson Education, 2008.
8. Kanetkar Y, "Let us C", BPB Publications, 2007.
9. Hanly J R &Koffman E.B, "Problem Solving and Programm design in C", Pearson Education, 2009.
10. E. Balagurusamy, "Programming with ANSI-C", Fourth Edition,2008, Tata McGraw Hill.
11. E. Balagurusamy, "Programming for Problem Solving", First Edition,2018, Tata McGraw Hill.
12. Venugopal K. R and Prasad S. R, "Mastering 'C'", Third Edition, 2008, Tata McGraw Hill.
13. B.W. Kernighan & D. M. Ritchie, "The C Programming Language", Second Edition, 2001, Pearson Education
14. ISRD Group, "Programming and Problem Solving Using C", Tata McGraw Hill,2008.
15. PradipDey ,ManasGhosh, "Programming in C", Oxford University Press, 2007.

Skill Enhancement Courses:

Web Designing Using HTML

General Course Information:

Course Code: 24SEC0210T Course Credits: 2 Type: Skill Enhancement (SEC) Contact Hours: 2 hours/week Mode: Lectures (L) Exam Duration: 2 hours	**Course Assessment Methods (Internal: 15; External: 35)** Two minor examinations each of 10 marks, Class Performance measured through percentage of lectures attended (3 marks) Assignment and quiz (2 marks), and end semester examination of 35 marks. The syllabus is divided into Two units. For the end semester examination, five questions are to be set by the examiner. Q. No. 1 is compulsory and contains five short questions covering entire syllabus consisting of 3 marks each. Rest four questions are set by giving two questions from each of the unit of the syllabus. A candidate is required to attempt three questions in all selecting one question from each unit consisting of 10 marks each in addition to compulsory Q.No.1

Prerequisite: This course requires a basic understanding of computer fundamentals and familiarity with navigating the internet. No prior knowledge of programming or web development is required, making it accessible to beginners.

About the Course: This course covers fundamental concepts in HTML, CSS, and JavaScript during teaching students learn to design and develop interactive websites with multimedia integration and responsive design principles. They learn to create structured HTML documents, style webpages with CSS, and add dynamic functionality using JavaScript, including event handling, and asynchronous programming techniques.

Course Outcomes: By the end of the course students will be able to:

CO1. List or describe: fundamental concepts of HTML, CSS, and JavaScript.

CO2. Apply: appropriate elements, properties, and techniques in web development.

CO3. Create: well-structured, visually appealing, and interactive webpages.

CO4. Implement: forms, styling, and functionality to enhance user experience.

CO5. Utilize: modern techniques like responsive design using JSON, XML and DOM manipulation.

CO6. Compare: and evaluate different development approaches and tools.

Course Contents

Unit - I

HTML5 Basic: Introduction to HTML Document, Elements and Tags, Text Formatting, Headings and Paragraphs, Adding Line Breaks and Horizontal Rules, Lists (Ordered, Unordered, Definition), Anchor Tags, Linking to External Resources (Images, Stylesheets, Scripts). HTML Tables, Forms, Input Types, Textboxes, Radio Buttons, Checkboxes, form submission methods (GET and POST), Dropdown Menus and Select Boxes, Grouping Form Elements with FieldSets and Legends, form Attributes, Embedding Multimedia (Images, Audio, Video).

HTML5 advance: Working with Meta Tags (Description, Keywords, Charset), Comments in HTML, Validation, Responsive Web Design Principles, Semantic Elements (<nav>, <main>, article, section, header, footer), Entities and Special Characters, Document Structure for Search Engine Optimization, Creating Forms with Advanced Features (File Uploads, Hidden Inputs), APIs (Canvas, Geolocation, Local Storage) Working with iframes, Embedding External Content (YouTube Videos, Google Maps), Responsive Images (map area) with srcset and sizes Attributes,

Unit – II

Cascading Style Sheets: Introduction to CSS and Basic Styling, Role of CSS in Web Design, Syntax and Structure of CSS Rules, Selectors and Specificity, Inline Styles, External vs. Internal Stylesheets, Box Model (Margin, Border, Padding, Content), Box Elements (Width, Height, Borders, Margins, Padding), Display Properties (Block, Inline, Inline-Block), Positioning Elements (Static, Relative, Absolute, Fixed).

CSS Advanced Techniques for Styling Text (Font Properties, Text Decoration, Text Alignment), Working with Colors and Backgrounds (Color Values, Background Images, Gradients), Managing Lists and Tables, Styling Links and Navigation Menus, CSS Flexbox/Grid Layouts and Properties, CSS Transitions Properties (Duration, Timing Function, Delay), Debugging and Troubleshooting CSS Issues.

Text and Reference Books:

1. "Learning Web Design: A Beginner's Guide to HTML, CSS, JavaScript, and Web Graphics" by Jennifer Niederst Robbins, 2018, 5th Ed., O'Reilly Media
2. "Head First HTML and CSS" by Elisabeth Robson and Eric Freeman, 2012, O'Reilly Media
3. "Responsive Web Design with HTML5 and CSS" by Ben Frain, 2014, 1st Edition, Packt Publishing.
4. "HTML, CSS, and JavaScript All in One" by Julie C. Meloni and Jennifer Kyrnin, 2020, 3rd Edition, Sams Publishing.
5. “HTML 5 Black Book, Covers CSS 3, JavaScript, XML, XHTML, AJAX, PHP and jQuery”, 2016 2nd Edition, Kindle Edition, Dreamtech Press.

Web Designing Using HTML Lab

General Course Information:

Course Code: 24SEC0210P Course Credits: 1 Type: Skill Enhancement Course (SEC-2) Contact Hours: 2 hours/week Mode: Labs Exam Duration: 3 hours	**Course Assessment Methods (Internal: 10; External: 15)** The Internal and External assessment is based on the level of participation in Lab sessions and the timely submission of Lab experiments / assignments, the performance in Viva-Voce, the quality of the lab file and ethical practices followed. The Internal examination is conducted by the Course Coordinator. The External examination is conducted by External Examiner appointed by the Controller of Examination in association with the Internal Examiner appointed by the Chairperson of the Department

Prerequisite: To learn web design using HTML require basic computer literacy, familiarity with internet usage, and a foundational understanding of HTML fundamentals such as elements, tags, and attributes.

About the Course: This course introduces students to the fundamental principles and practices of web design using HTML. It covers essential concepts such as HTML document structure, elements, tags, text formatting, lists, tables, forms, and multimedia embedding. It also explores basic CSS for styling web pages and introduces responsive design principles.

Course Outcomes: By the end of the course students will be able to:

- **CO1. Implementation:** Students will be able to implement HTML web pages.
- **CO2. Analysis:** students will develop the ability to analyse and interpret HTML and CSS code to understand its structure.
- **CO3. Compare:** students will compare and contrast different approaches to web design, including the use of HTML and CSS, to achieve specific design goals and user requirements.
- **CO4. Integrate:** students will integrate multimedia elements such as images, audio, and video seamlessly into HTML documents.
- **CO5. Create:** able to create well-designed and visually appealing web pages using HTML and basic CSS styling techniques.
- **CO6. Demonstrate:** student will demonstrate responsive web pages and showcasing their ability to create user-friendly.

Course Contents

List of Experiments/ assignments

1. Basic HTML Document Creation: Create a simple HTML document with necessary elements like <!DOCTYPE>, <html>, <head>, <title>, and <body>.
2. Demonstrate various text formatting elements such as , , <u>, <s>, etc.
3. Create ordered, unordered, and definition lists. Include hyperlinks to external resources using <a> tags.
4. Construct a table with rows and columns. Develop a simple form containing input types like textboxes, radio buttons, checkboxes, dropdown menus, and text areas.
5. Embed images, audio, and video files within an HTML document using appropriate tags.
6. Add meta tags for description, keywords, and charset. Include comments within the HTML document for better readability.
7. Validate HTML code using W3C validator. Create a responsive webpage using media queries and viewport meta tags.
8. Implement semantic elements such as <nav>, <main>, <article>, <section>, <header>, and <footer> for better SEO.
9. Implement advanced form features like file uploads and hidden inputs. Utilize HTML5 APIs like Canvas, Geolocation, and Local Storage.
10. Embed external content such as YouTube videos and Google Maps using iframes.

11. Apply basic CSS styles to HTML elements including font properties, colors, backgrounds, and text alignment.
12. Demonstrate the concept of margin, border, padding, and content. Style box elements with width, height, borders, margins, and padding.
13. Understand and implement various display properties like block, inline, and inline-block. Experiment with positioning elements statically, relatively, absolutely, and fixedly.
14. Apply advanced CSS techniques for styling text such as font properties, text decoration, and alignment. Experiment with different color values, background images, and gradients.
15. Style lists and tables with custom designs. Design navigation menus and style hyperlinks accordingly.
16. Implement CSS Flexbox and Grid layouts for better page structuring. Add transitions with properties like duration, timing function, and delay to enhance user experience.
17. Debug and troubleshoot CSS issues using browser developer tools and techniques like validation and code review.

Note: The actual experiment / assignments will be designed by the course coordinator. One assignment is to be done in groups of two or three students. The assignments must meet the objectives of the course and the levels of given course outcomes. The list of assignments and schedule of submission will be prepared by course coordinator in the beginning of the semester.

Discipline Specific Courses:

Operating System

General Course Information

Course Code:24ICA0301T Course Credits: 3 Type: Discipline Specific Courses Contact Hours: 3 hours/week Mode: Lectures (L) Exam Duration: 2.5 hours	**Course Assessment Methods (Internal: 20; External: 50)** **Internal Assessment: -** Two minor examinations each of 10 marks, Class Performance measured through percentage of lectures attended (5 marks) Assignment and quiz (5 marks), and end semester examination of 50 marks. **External Examination (Marks: 50): -** The syllabus is divided into four units. For the end semester examination, nine questions are to be set by the examiner. Question No. 1 is compulsory and contains five short answer questions (2 marks each) covering entire syllabus. Rest eight more questions (each question of at least two parts) will be set by giving two questions from each unit of the syllabus. A candidate is required to attempt five questions in all by selecting one question from each unit in addition to compulsory Question No.1. All questions will carry equal marks.

Prerequisite: Basic understanding of computer science fundamentals, including programming concepts, data structures, algorithms, and familiarity with computer architecture and hardware components.

About the Course: This course provides an in-depth understanding of operating systems, covering concepts like process management, CPU scheduling, deadlock handling, file systems, and storage management, focusing on various Operating System types and techniques.

Course Outcomes: By the end of the course students will be able to:

CO1. **List:** various functions and design characteristics of operating systems.

CO2. **Explain:** fundamental concepts of operating systems.

CO3.**Apply:** operating system design concepts for solving problems regarding scheduling, memory management, disk management and deadlocks etc.

CO4. **Analyse:** the issues related to various operating systems.

CO5. **Design:** solutions for the memory and process management problems.

Course Content

Unit-I

Introductory Concepts: Operating systems functions and characteristics, Types of Operating systems: Batch operating system, Time-sharing OS, Distributed operating system, Real-time systems. Operating system services and systems calls. System programs, Operating System Structure.

Unit-II

Processes: Process concept, Process Control Block, Operations on processes, cooperating processes. CPU scheduling: Levels of Scheduling, scheduling criteria, Comparative study of scheduling algorithms, Algorithm evaluation, multiple processor scheduling.

Unit – III

Deadlock: System model, Deadlock characterization, Methods for handling deadlocks, Deadlock prevention, Deadlock avoidance, Deadlock detection, Recovery from deadlock

Unit – IV

File Systems: Types of Files and their access methods, File allocation methods, Directory Systems: Structured Organizations, directory and file protection mechanisms, disk scheduling and its associated algorithms.

Storage Management: Storage allocation methods: Single contiguous allocation, non-contiguous memory allocation, Paging and Segmentation techniques.

Text and Reference:

1. Silberschatz, Peter B. Galvin and Greg Gagne, *Operating System Concepts*, 8th Edition, WileyIndian Edition, 2010.

2. Andrew S Tanenbaum, *Modern Operating Systems*, Third Edition, Prentice Hall India, 2008.

3. D.M. Dhamdhere, *Operating Systems*, 2nd edition, Tata McGraw Hill, 2010.

4. William Stallings, *Operating Systems– Internals and Design Principles*, 5th Edition, Prentice Hall India, 2000.

Operating System Lab

General Course Information:

Course Code: 24ICA0301P Course Credits: 1 Type: Discipline Specific Courses (DSC) Contact Hours: 2 hours/week Mode: Lab Exam Duration: 3 hours	**Course Assessment Methods (Internal: 10; External: 20)** The Internal and External assessment is based on the level of participation in Lab sessions and the timely submission of Lab experiments / assignments, the performance in Viva-Voce, the quality of the lab file and ethical practices followed. The Internal examination is conducted by the Course Coordinator. The External examination is conducted by External Examiner appointed by the Controller of Examination in association with the Internal Examiner appointed by the Chairperson of the Department.

Pre-requisites: Basic programming skills.

About the Course: The Operating Systems Lab aims to provide practical exposure to the concepts of operating systems, focusing on hands-on experience with Windows and Linux environments. Students will learn system administration, scripting, and essential commands for efficient system management.

Course Outcomes: By the end of the course, students will be able to:

CO1.**Demonstrate:** proficiency in using Windows and Linux operating systems, including their basic features and utilities

CO2.**Apply:** system administration skills in Linux, including user management, disk usage, and process monitoring.

CO3. **Develop** and **debug:** shell scripts to automate tasks involving file operations, loops, conditionals, and text processing.

CO4. **Analyse:** and solve real-world problems using Linux tools such as pipes, filters, sed, and awk.

CO5. **Create:** lab record for assignments that includes problem definitions, design of solutions and conclusions.

CO6. **Demonstrate:** use of ethical practices, self-learning and team spirit.

List of experiments/assignments:

1. Study of WINDOWS and Linux operating systems.
2. Study of vi editor.
3. Administration of LINUX Operating System.
4. Writing shell scripts for basic file operations.
5. Writing shell scripts to demonstrate loop and conditional constructs.
6. Implementation of user and group management commands in Linux.
7. Study and implementation of pipes and filter commands.
8. Writing shell scripts for text processing using sed and awk.
9. Study of disk usage analysis commands in Linux.
10. Setting up a dual-boot system with Windows and Linux.

Note:

The actual experiments/assignments will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.

Software Engineering

General Course Information:

Course Code: 24ICA0302T Course Credits: 3 Type: Professional Core Contact Hours: 3 hours/week Mode: Lectures (L) Exam Duration: 2.5 hours	**Course Assessment Methods (Internal: 20; External: 50)** **Internal Assessment: -** Two minor examinations each of 10 marks, Class Performance measured through percentage of lectures attended (5 marks) Assignment and quiz (5 marks), and end semester examination of 50 marks. **External Examination (Marks: 50): -** The syllabus is divided into four units. For the end semester examination, nine questions are to be set by the examiner. Question No. 1 is compulsory and contains five short answer questions (2 marks each) covering entire syllabus. Rest eight more questions (each question of at least two parts) will be set by giving two questions from each unit of the syllabus. A candidate is required to attempt five questions in all by selecting one question from each unit in addition to compulsory Question No.1. All questions will carry equal marks.

Prerequisite: Basic Programming Skills and Innovative assessment.

About the Course: This course of Software Engineering and Testing will help the learners to understand the systematic approach of all software development phases i.e. from initial stage to final stage of software systems. Learners will gain knowledge about the various processes which are used in software industry for the development of software product and about the testing methods, tools for creating good test cases to improve the quality of software.

Course Outcomes: By the end of the course students will be able to:

CO1. **Define:** the concepts related to software engineering and to comprehend about the stages of Software Development Life Cycle.
CO2. **Demonstrate:** the selection of Software Process Models as per the requirements and to assess the various processes of requirement analysis for software engineering problems.
CO3. **Apply:** the software requirement analysis and design process to model the system as per the requirements and to comprehend the principles, processes of Software Project Management including the Software Configuration Management and Risk Management.
CO4. **Design:** test cases and apply the testing techniques for software engineering problems.
CO5. **Predict:** software quality based on quality parameters and quality models.

Course Contents

Unit – I

Introduction to Software Engineering: Evolution of Software Engineering, Software Crisis, Types of Software Products, Software Development Life Cycle Phases. Validation and Verification, Software Characteristics, Software Engineering Paradigms: Classical Waterfall Model, Iterative Waterfall Model, Prototyping Model, Evolutionary Model, Spiral Model, Selection of Life Cycle Model.
Software Project Management: Software Project Management Plan (SPMP), Metrics for Project Size estimation- Lines of Code, Function Point Metric, Software Cost estimation - COCOMO, Project Scheduling, Personnel Planning, Organization and Team Structures, Software Configuration Management (SCM), Software Risks, Software Risk Management.

Unit – II

Software Requirements: Functional and non-functional Requirements, User and interface requirements, Software Requirement Specification (SRS), Requirement Engineering Process.
Problem Analysis: Structured Analysis, Data Flow Diagrams (DFD), Decision Tables, Decision Trees, Data Dictionary, Structured Charts, Object Oriented Analysis, System Models: Context Models, Data Modelling, Behavioral Modelling, Object Models, Structured Models.
Software Design: Software Design Fundamentals, Design Principles, Function-Oriented Software Design, Object Oriented Design. Characteristics of good user interface, Coding Standards and guidelines, Code

Review.

Unit –III

Software Testing: Software Testing Basics, Necessity and Objectives of Testing, Difference between Inspection and Testing, Testing vs. Debugging, Testing Life Cycle, Test Artifacts, Test Plan, Test Case Design, Software Testing Strategies, The V-Model of Software Testing, Levels of Software Testing- Unit Testing, Integration Testing-Top down Integration Testing and Bottom-up Integration Testing, Regression Testing, Smoke Testing, System Testing- Recovery Testing, Security Testing, Stress Testing, Performance Testing, Acceptance Testing- Alpha Testing, Beta Testing, Gamma Testing, Software Test Report (STR), Software Testing Tools, Static and Dynamic Testing tools.

Computer Aided Software Engineering (CASE): CASE Environment, advantages of CASE, CASE support in Software Life Cycle, Characteristics of CASE tools.

Unit– IV

Software Testing Methods: Black Box Testing Methods: Equivalence class partitioning, Boundary-value analysis, Error guessing, graph- based testing methods, White Box Testing Methods: Statement coverage, Condition coverage, Path testing, Data flow testing. Object Oriented Testing, Web Testing, GUI testing.

Software Quality: Software Quality Concepts, ISO 9126 Quality Factors, McCall's Quality Factors, Software Quality Assurance (SQA) Activities, Software Reviews-Walkthroughs, Formal Technical Review (FTR), Defect Amplification Model, ISO 9000 series Quality Standards, Capability Maturity Model (CMM), Software Reliability.

T**ext and Reference Books:**

1. Rajib Mall,Fundamentals of Software Engineering, PHI Learning Pvt. Ltd., Third Edition, 2009
2. K. K. Aggarwal&Yogesh Singh,Software Engineering Programs Documentation Operating Procedures, A New Age International Publishers.
3. PankajJalote, An Integrated Approach to Software Engineering, Narosa Publications, Third Edition, 2007.
4. Roger S. Pressman, Software Engineering A Practitioner's Approach, McGraw Hill International Edition
5. M. G. Limaye, Software Testing: Principles, Techniques and Tools, TMH, 2009.
6. RenuRajani&Pradeep Oak,Software Testing Effective Methods Tools and Techniques, McGraw Hill Education Pvt. Limited, Second Edition, 2018.
7. Nina S. Godbole, Software Quality Assurance Principles and Practice, Narosa Publications, 2011.
8. Yogesh Singh, Software Testing, Cambridge University Press, 2016.

Software Engineering Lab

General Course information:

Course Code: 24ICA0302P Course Credits: 1 Type: Professional Core Lab. Course Mode: Lab practice and assignments Contact Hours: 2 hours / week	**Course Assessment Methods (Internal: 10; External: 20)** The Internal and External assessment is based on the level of participation in Lab sessions and the timely submission of Lab experiments / assignments, the performance in Viva-Voce, the quality of the lab file and ethical practices followed. The Internal examination is conducted by the Course Coordinator. The External examination is conducted by External Examiner appointed by the Controller of Examination in association with the Internal Examiner appointed by the Chairperson of the Department.

Pre-requisites: Knowledge on "Software Engineering".

About the Course: This lab course provides opportunity to students to implement software engineering methodologies involved in the phases for project development. This includes DFD's, UML Diagram and Case Tools. In addition, students learn to design SRS and implement software engineering methodologies.

Course Outcomes: By the end of the course students will be able to:

CO1. **Understand:** the software engineering methodologies involved in the phases for project development.
CO2. **Demonstrate:** knowledge about open-source tools used for implementing software engineering methods.
CO3. **Compare:** strengths and weaknesses of various software engineering methods.
CO4. **Design:** product-startups implementing software engineering methods.
CO5**. Create:** lab assignment record that includes problem definitions, solutions and conclusions.

Open-source Tools: Star UML / UML Graph / Top cased

Prepare the following documents and develop the software project startup, prototype model, using software engineering methodology for at least two real time scenarios or for the sample experiments.

- Problem Analysis and Project Planning -Thorough study of the problem – Identify Project scope, Objectives and Infrastructure.
- Software Requirement Analysis – Describe the individual Phases/modules of the project and Identify deliverables. Identify functional and non-functional requirements.
- Data Modelling – Use work products – data dictionary.
- Software Designing – Develop use case diagrams and activity diagrams, build and test class diagrams, sequence diagrams and add interface to class diagrams.
- Prototype model – Develop the prototype of the product.
- The SRS and prototype model should be submitted for the end semester examination.

List of experiments/assignments

Do the following 8 exercises for any two projects given in the list of sample projects or any other projects:

1. Development of problem statement.
2. Preparation of Software Requirement Specification Document, Design Documents and Testing Phase related documents.
3. Preparation of Software Configuration Management and Risk Management related documents.
4. Study and usage of any Design phase CASE tool.
5. Performing the Design by using any Design phase CASE tools.
6. Design conceptual model of UML.
7. Develop test cases for unit testing and integration testing.
8. Develop test cases for various white box and black box testing techniques.

Sample Projects:

1. Online Exam Registration
2. Course management system (CMS)
3. E-Bidding
4. Software Personnel Management System
5. Passport automation System
6. Book Bank
7. Stock Maintenance System
8. Online course reservation system
9. E-ticketing
10. Credit Card Processing
11. E-book management System.
12. Recruitment system

Discrete Mathematics

General Course Information

Course Code: 24ICA0303T Course Credits: 4 Type: Professional Core Contact Hours: 4 hours/week Mode: Lectures (L) Exam Duration: 2.5 hours	**Course Assessment Methods (Internal: 30; External: 70)** **Internal Assessment (Marks: 30): -** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (5 marks) Assignment and quiz (5 marks), and end semester examination of 70 marks. **External Examination (Marks: 70): -** The syllabus is divided into four units. For the end semester examination, nine questions are to be set by the examiner. Question number one is compulsory and contains Seven short answer questions of 2 marks each covering entire syllabus. Rest eight more questions (each question of at least two parts) will be set by giving two questions from each of the unit of the syllabus. A candidate is required to attempt five questions in all by selecting one question from each of unit in addition to compulsory Question No.1. All questions will carry equal marks.

Prerequisite: Basic knowledge of Number Theory, Calculus and Algebra.

About the Course: Discrete Mathematics is a core and essential course for every graduate in Computer Science and Application. This branch of mathematics mainly deals with discrete objects (as computer runs on discrete steps). It provides a mathematical language for computers to resolve real world problems by incorporating different methods applicable to various discrete structures. This course introduces set theory, propositional calculus, permutation, combinations and discrete probability, graph theory and recurrence relation.

Course Outcomes: By the end of the course students will be able to:

CO1. **Outline:** various discrete structures and related operations.
CO2. **Illustrate:** different discrete structures with the help of examples.
CO3. **Apply:** appropriate techniques to solve problems related to discrete structures.
CO4. **Justify:** the solutions with the help of proof.
CO5. **Combine:** techniques related to discrete structures for solving real world problems.

Course Content

Unit-1

Set Theory: Introduction to Set Theory, Venn Diagrams, Set Operations, Algebra of Sets, Duality, Finite, Infinite Sets and Counting Principle, Classes of Sets, Power Sets, Partitions, Multi Sets, Relations: Cartesian Product, Representation of Relations, Types of Relation, Equivalence Relations, Functions: Definition, Types of Functions, Composition of Functions, Inverse Function.

Unit-II

Logic and Propositional Calculus: Introduction, Propositions and Compound Propositions, Basic Logical Operations, Propositions and Truth Tables, Tautologies and Contradictions, Logical Equivalence, Algebra of Propositions, Conditional and Bi-conditional Statements. Permutation, Combinations and Discrete Probability: Introduction, The rule of Sum and Product, Permutations, Combinations, Discrete Probability, Conditional Probability, Bayes' Theorem.

Unit-III

Graphs Theory: Introduction to Graphs, Multi Graph, Directed and Undirected Graphs, Subgraphs, Bipartite Graphs, Regular Graphs, Connected Graphs, Homomorphic and Isomorphic Graphs, Cut points and Bridges, Paths and Circuits, Euler Graph, Hamiltonian Graph, Planar Graph, Euler Formula, Weighted Graphs, Dijkstra's Shortest Path Algorithm for Weighted Graphs, Trees, Spanning Trees, Minimum Spanning Tree (Prim's and Kruskal's Algorithm).

Unit-IV

Recurrence Relation: Introduction, Recursion, Recurrence Relations, Linear Recurrence Relations with Constant Coefficients, Homogeneous Solutions, Particular Solutions, Total Solutions.

Text and Reference Books:

1. S. Lipschutz and M. Lipson, *Discrete Mathematics*, Tata McGraw Hill, Third Edition, 2017.
2. C. L. Liu, *Elements of Discrete Mathematics*, Tata McGraw Hill, Fourth Edition, 2017.
3. Kenneth H. Rosen, *Discrete Mathematics and its applications*, Seventh Edition, Tata McGraw Hill, 2017.
4. B. Kolman, R. C. Busby and S. C. Ross, *Discrete Mathematical structures*, Sixth Edition, PHI, 2015.

Minor Courses:

Computer Network

General Course Information

Course Code: 24MIC0321T Course Credits: 3 Type: Minor Course Contact Hours: 3 hours/week Mode: Lectures (L) Examination Duration: 2.5 Hours	**Course Assessment Methods (Internal: 20; External: 50)** **Internal Assessment: -** Two minor examinations each of 10 marks, Class Performance measured through percentage of lectures attended (5 marks) Assignment and quiz (5 marks), and end semester examination of 50 marks. **External Examination (Marks: 50): -** The syllabus is divided into four units. For the end semester examination, nine questions are to be set by the examiner. Question No. 1 is compulsory and contains five short answer questions (2 marks each) covering entire syllabus. Rest eight more questions (each question of at least two parts) will be set by giving two questions from each unit of the syllabus. A candidate is required to attempt five questions in all by selecting one question from each unit in addition to compulsory Question No.1. All questions will carry equal marks.

Pre-requisites: Basic knowledge of Digital and Analog Communication.

About the Course: This course has been designed with aim to provide students with an overview of the concepts and fundamentals of data communication and computer networks. The learner is given an opportunity to grasp various algorithms for routing of data, forwarding data and switching the data from hop to hop. Layered Architecture adds value to the subject contents.

Course Outcomes: By the end of the course students will be able to:

CO1. Outline: various models, topologies and devices of Computer Networks.
CO2. Explain: functions of various layers in Network Reference Model.
CO3. Apply: different network concepts in various network communication protocols.
.CO4. Analyze: performance of various protocols in different scenarios.
CO5. Design: network for an organization.

Course content

Unit I

Data communication: Components, Network: Uses, Topologies, Network Services, OSI and TCP/IP Reference Models; Network categories: LAN, MAN, WAN, Switching Techniques: Circuit Switching, Packet Switching, Message Switching, Networking Devices**:** Hubs, Repeaters, Bridges, Modems, Switches, Routers, and Gateways.

Unit II

Data Link Layer-design issues, Framing & Error Handling: Framing Protocols, Error detection and correction mechanisms; Flow Control Protocols: Stop-and-wait, Sliding Window protocols: Go-back-N and Selective Repeat;, Multiple Access Communication: Random Access-ALOHA,

Unit III

Network Layer-Design issues, store and forward packet switching connection less and connection oriented networks, Distance Vector Routing, Link State Routing, Hierarchical Routing, Congestion control algorithms, Internetworking: IPV4 and IPV6, IP Addressing (Classful Addressing, Private IP Addresses, Classless Addressing, Sub-netting).

Unit IV

Transport Layer: Transport layer Services: Addressing, Multiplexing, Flow control, Buffering and Error control. Internet Transport Protocols: UDP, TCP. Application Layer: Introduction to DNS, HTTP, SMTP, WWW and Multimedia.

Text and Reference Books:

1. Andrew S Tanenbaum, *Computer Networks*, 5th Edition, Pearson publications, 2010.
2. Forouzan, *Data Communication and networking* ,5th Edition, Tata McGrawHill, 2012.
3. William Stalling, *Data & Computer Communication* 6th edition, LPE Pearson Education, 2013.
4. Todd Lammle, *CCNA Study Guide*, 6th Edition, 2013.
5. RFCs and Internet Drafts available from Internet Engineering Task Force.

Computer Network Lab

General Course Information

Course Code: 24MIC0321P Course Credits: 1 Type: Minor Course Lab. Contact Hours: 2 hours/week Mode: Lab. practice and assignments	**Course Assessment Methods (internal: 10; external: 20)** The internal and external assessment is based on the level of participation in lab. sessions and the timely submission of lab experiments/assignments, the quality of solutions designed for the assignments, the performance in VIVA- VOCE, the quality of lab. file and ethical practices followed. The internal examination is conducted by the course coordinator. The external examination is conducted by external examiner appointed by the Controller of Examination in association with the internal examiner appointed by the Chairperson of the Department.

Pre-requisites: knowledge of programming, digital and analog communication.

About the Course: This course has been designed with an aim to provide students with an overview of the concepts and fundamentals of data communication and computer networks. Students learn about various topologies, network devices, routing protocols, firewall amongst other features and devices of Computer Networks.

Course Outcomes: By the end of the course students will be able to:

CO1. **Describe:** various network topologies and networking devices.
CO2. **Justify:** a particular routing protocol for any implemented data communication networks.
CO3. **Construct:** a network and implement various network protocols.
CO4. **Devise:** solutions for various routing and switching problems in Computer Networks.
CO5**. Create:** lab records for the solutions of the assignments.
CO6. **Demonstrate:** ethical practices, self-learning and team spirit.

List of Experiments/assignments:

1. Familiarization with networking components and devices: LAN Adapters - Hubs -Switches -Routers
2. Familiarization with transmission media and Tools: Co-axial cable, UTP Cable, Crimping Tool, Connectors
3. Installation and introduction of simulation tools Packet-Tracer/ GNS3.
4. Preparing the UTP cable for cross and direct connections using crimping tool.
5. Introduction to various interior and exterior routing protocols.
6. Configuration of RIP protocol on routers to configure a network topology.
7. Configuration of a wireless device in simulated environment.
8. Configuration of TELNET protocol on router for remote access.
9. Configuration of access lists on network to stop unwanted traffic on network.
10. Configuration of zone based firewall in network.

Note:

The actual experiments/assignments will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.

Multidisciplinary Courses:

Software Engineering

General Course Information:

Course Code: 24MDC0310T Course Credits: 3 Type: Multidisciplinary Course (MDC) Contact Hours: 3 hours/week Mode: Lectures (L) Exam Duration: 2.5 hours	**Course Assessment Methods (Internal: 25; External: 50) Internal Assessment: -** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (3 marks) Assignment and quiz (2 marks), and end semester examination of 50 marks. **External Examination (Marks: 50): -** The syllabus is divided into four units. For the end semester examination, nine questions are to be set by the examiner. Question number one is compulsory and contains five short answer questions of 2 marks each covering entire syllabus. Rest eight more questions (each question of at least two parts) will be set by giving two questions from each of the unit of the syllabus. A candidate is required to attempt five questions in all by selecting one question from each of unit in addition to Question No.1is compulsory.

Prerequisite: Basic Programming Skills and Innovative assessment.

About the Course: This course of Software Engineering and Testing will help the learners to understand the systematic approach of all software development phases i.e. from initial stage to final stage of software systems. Learners will gain knowledge about the various processes which are used in software industry for the development of software product and about the testing methods, tools for creating good test cases to improve the quality of software.

Course Outcomes: By the end of the course students will be able to:

CO1. Define: the concepts related to software engineering and to comprehend about the stages of Software Development Life Cycle.
CO2. Demonstrate: the selection of Software Process Models as per the requirements and to assess the various processes of requirement analysis for software engineering problems.
CO3. Apply: the software requirement analysis and design process to model the system as per the requirements and to comprehend the principles, processes of Software Project Management including the Software Configuration Management and Risk Management.
CO4. Design: test cases and the testing techniques for software engineering problems.
CO5. Predict: software quality based on quality parameters and quality models.

Course Contents

Unit– I

Introduction to Software Engineering: Evolution of Software Engineering, Software Crisis, Types of Software Products, Software Development Life Cycle Phases. Validation and Verification, Software Characteristics, Software Engineering Paradigms: Classical Waterfall Model, Iterative Waterfall Model, Prototyping Model, Evolutionary Model, Spiral Model, Selection of Life Cycle Model.

Software Project Management: Software Project Management Plan (SPMP), Metrics for Project Size Estimation-Lines of Code, Function Point Metric, Software Cost estimation-COCOMO, Project Scheduling, Personnel Planning, Organization and Team Structures, Software Configuration Management (SCM), Software Risks, Software Risk Management.

Unit –II

Software Requirements: Functional and non-functional Requirements, User and interface requirements, Software Requirement Specification (SRS), Requirement Engineering Process.
Problem Analysis: Structured Analysis, Data Flow Diagrams (DFD), Decision Tables, Decision Trees, Data Dictionary, Structured Charts, Object Oriented Analysis, System Models: Context Models, Data Modelling, Behavioral Modelling, Object Models, Structured Models.
Software Design: Software Design Fundamentals, Design Principles, Function-Oriented Software Design, Object Oriented Design. Characteristics of good user interface, Coding Standards and guidelines, Code Review.

Unit–III

SoftwareTesting: SoftwareTestingBasics,NecessityandObjectivesofTesting,DifferencebetweenInspection and Testing, Testing vs. Debugging, Testing Life Cycle, Test Artifacts, Test Plan, Test Case Design, Software Testing Strategies, The V-Model of Software Testing, Levels of Software Testing- Unit Testing, Integration Testing-Top down Integration Testing and Bottom-up Integration Testing, Regression Testing, Smoke Testing, System Testing- Recovery Testing, Security Testing, Stress Testing, Performance Testing, Acceptance Testing- Alpha Testing, Beta Testing, Gamma Testing, Software Test Report(STR), Software Testing Tools, Static and Dynamic Testing tools.

Computer Aided Software Engineering (CASE): CASE Environment, advantages of CASE, CASE supporting Software Life Cycle, Characteristics of CASE tools.

Unit–IV

Software Testing Methods: Black Box Testing Methods: Equivalence class partitioning, Boundary-value analysis, Error guessing, graph- based testing methods, White Box Testing Methods: Statement coverage, Condition coverage, Path testing, Data flow testing. Object Oriented Testing, Web Testing, GUI testing.

Software Quality: Software Quality Concepts, ISO 9126 Quality Factors, McCall's Quality Factors, Software Quality Assurance (SQA) Activities, Software Reviews-Walkthroughs, Formal Technical Review (FTR), Defect Amplification Model, ISO 9000 series Quality Standards, Capability Maturity Model (CMM), Software Reliability.

T**extandReferenceBooks:**

1. RajibMall, Fundamentals of Software Engineering, PHIL earning Pvt. Ltd.,Third Edition,2009
2. K.K.Aggarwal & Yogesh Singh, Software Engineering Programs Documentation Operating Procedures, A New Age International Publishers.
3. Pankaj Jalote, An Integrated Approach to Software Engineering, Narosa Publications, Third Edition, 2007.
4. RogerS. Pressman, Software Engineering A Practitioner's Approach, McGraw Hill International Edition
5. M.G.Limaye, Software Testing: Principles, Techniques and Tools, TMH,2009.
6. RenuRajani&Pradeep Oak, Software Testing Effective Methods Tools and Techniques, McGraw Hill Education Pvt. Limited, SecondEdition,2018.
7. NinaS. Godbole, Software Quality Assurance Principles and Practice, Narosa Publications,2011.
8. Yogesh Singh, Software Testing, Cambridge University Press,2016.

Skill Enhancement Course:

Mobile Application Development

General Course Information:

Course Code: 24SEC0310T Course Credits: 2 Type: Skill Enhancement Course (SEC) Contact Hours: 2 hours/week Mode: Lectures (L) Exam Duration: 2 hours	**Course Assessment Methods (Internal: 15; External: 35)** Two minor examinations each of 10 marks, Class Performance measured through percentage of lectures attended (3 marks) Assignment and quiz (2 marks), and end semester examination of 35 marks. The syllabus is divided into Two units. For the end semester examination, five questions are to be set by the examiner. Q. No. 1 is compulsory and contains five short questions covering entire syllabus consisting of 3 marks each. Rest four questions are set by giving two questions from each of the unit of the syllabus. A candidate is required to attempt three questions in all selecting one question from each unit consisting of 10 marks each in addition to compulsory Q.No.1

Prerequisite: Java Programming and Object-Oriented programming, Knowledge of RDBMS and OLTP.

About the Course: Mobile Application Development has been introduced as a Professional Elective course for Students keeping in view the Employers' requirements. Android Platform forms the basis for developing Mobile Applications since the last decade as compared to IOS Platform for Apple Products. The Environment requires User Interface to be developed using Buttons, Check-Boxes, Alert Dialog and its kind.

Course Outcomes: By the end of the course students will be able to:

CO1 List: basics of Android, its Evolution and its Architecture.

CO2 Describe & Demonstrate: Lifecycle of Software for Android Mobile Applications.

CO3 Apply: Mobile Applications on the Android Platform.

CO4 Implement: and Compare working with Buttons and other Widgets for Visual Environment

CO5 Design and Develop: Mobile Applications using data storage in SQLite Database.

Course Contents

Unit - I

Android OS Architecture: Architecture of Android based devices, Introduction of Blackberry OS, Firefox OS & IOS, Understanding Android application structure and its updating, ARM and MIPS processor, Internal Details, Dalvik VM, Android Core Building Blocks, Android Emulator, Android Manifest.xml, R.java file, Screen Orientation.

Unit-II

Activity, Intent & Fragment: Activity Lifecycle, Activity Example, Implicit Intent, Explicit Intent.
Android Menu: Option Menu, Context Menu, Popup Menu.
Layout Manager: Relative Layout, Linear Layout, Table Layout, Grid Layout.

Text and Reference Books:

1. Jason Wei, Android Database Programming, packt publishing, 2012
2. Mark.L.Murphy,AndroidProgrammingTutorials,3rdEdition, 2010
3. Bill Phillips et al., Android Programming - The "Big Nerd Ranch" Guide 2017
4. Rick Rogers et al., Android Application Development: Programming with the Google SDK.
5. T1. Lauren Darcey and Shane Conder," Android Wireless Application Development", Peason Education, 2nd ed..(2011).
6. Reto Meier, "Professional Android 2 Application Development", Wiley India Pvt.Ltd.

Mobile Application Development Lab

General Course Information:

Course Code: 24SEC0310P Course Credits: 1 Type: Skill Enhancement Course (SEC) Contact Hours: 2 hours/week Mode: Labs Exam Hours: 3 Hours	**Course Assessment Methods (Internal: 10; External: 15)** The Internal and External assessment is based on the level of participation in Lab sessions and the timely submission of Lab experiments/assignments, the performance in Viva-Voce, the quality of the lab file and ethical practices followed. The Internal examination is conducted by the Course Coordinator. The External examination is conducted by External Examiner appointed by the Controller of Examination in association with the Internal Examiner appointed by the Chairperson of the Department.

Prerequisite: Java Programming and Object-oriented programming, knowledge of XML, JSON and database concepts.

About the Course: This laboratory course on Android Programming helps students to learn how to develop android apps. A study of the subject matter presented in this course will enable the student to become familiar with.

Course Outcomes: By the end of the course students will be able to:

CO1. **Analyse:** the Development Environment and the working of Emulator for android application.
CO2. **Demonstrate and Design:** different activities and layouts of application.
CO3. **Identify:** embedded JSON and XML file in application design.
CO4. **Develop:** application based on SQLite and latest connection providers.
CO5. **Create & Demonstrate:** use of ethical practices, self-learning and team spirit.

Course Contents
List of Experiments/ assignments

1. Setting up development environment, Dalvik Virtual Machine & .apk file extension.
2. Fundamentals: - Basic Building blocks: - Activities, Services, Broadcast Receivers & Content providers.
3. UI Components: – Views & Notifications, Uses of Components for communication -Intents & Intent Filters.
4. Emulator-Android Virtual Device: - Launching emulator, Editing emulator settings, Emulator shortcuts.
5. Develop an app for demonstrating the communication between Intents .Design and implement forms in MS Access.
6. Design a Basic of UI structure:- Form widgets, Text Fields, Layouts, [dip, dp, sip, sp] versus px, Menu, Option menu, Context menu, Sub menu, menu from xml, menu via code.
7. Implementation of Intents: - Explicit Intents, Implicit intents with Examples.
8. Styles & Themes:- styles.xml, draw able resources for shapes, gradients (selectors), style attribute in layout file, Applying themes via code and manifest file.

Note: The actual experiment / assignments will be designed by the course coordinator. One assignment to be done in groups of two or three students. The assignments must be met the objectives of the course and the levels of given course outcomes. The list of assignments and schedule of submission will be prepared by course coordinator in the beginning of the semester.

Discipline Specific Courses:

Computer Architecture

General Course Information

Course Code: 24ICA0401T Course Credits: 3 Type: DSC Contact Hours: 3 hours/week Mode : Lectures (L) Examination Duration: 2.5 Hours.	**Course Assessment Methods (Internal: 20; External: 50)** **Internal Assessment: -** Two minor examinations each of 10 marks, Class Performance measured through percentage of lectures attended (5 marks) Assignment and quiz (5 marks), and end semester examination of 50 marks. **External Examination (Marks: 50): -** The syllabus is divided into four units. For the end semester examination, nine questions are to be set by the examiner. Question No. 1 is compulsory and contains five short answer questions (2 marks each) covering entire syllabus. Rest eight more questions (each question of at least two parts) will be set by giving two questions from each unit of the syllabus. A candidate is required to attempt five questions in all by selecting one question from each unit in addition to compulsory Question No.1. All questions will carry equal marks.

Pre-requisites: Digital Electronics and computer systems.

About the Course: Computer Architecture and organization describes the role of instruction set architecture in digital computer, main memory, and input/output devices. It illustrates the simple data path and control design for processors. It helps to understand the different operations and concept of instructions. It would enable the students to learn the basic function and architecture of modern computer systems.

Course Outcomes: By the end of the course students will be able to:
CO1. Outline: general concepts of digital electronics and computer organization and architecture.
CO2. Discuss: basic components and their interfacing.
CO3. Apply: instructions for performing different operations.
CO4. Analyze: effect of addressing modes on the execution time of a program.
CO5. Contrast: different types of memory, their architecture and access methods.
CO6. Design: different instruction sets.

Course Content

Unit I

Basic Principles: Boolean algebra and Logic gates, Combinational logic blocks (Adders, Subtractors, Multiplexers, Encoders, decoders, demultiplexers, KMaps), Sequential logic blocks (Flip-Flops-SR Flip- Flop, D Flip-Flop, J-K Flip-Flop, T Flip-Flop,, Registers); Flynn's classification of computers (SISD, MISD, MIMD);

Unit II

Computer Organization: Store program control concept, Instruction codes, timing and control, instruction cycle; type of instructions: memory reference, register reference, I/O reference; Implementation of control unit.
CPU Architecture types: computer register, (accumulator, register, stack, memory/ register) detailed data path of a typical register based CPU.

Unit III

Instruction Set Architecture & Parallelism: Instruction set based classification of processors (RISC, CISC, and their comparison); Stack Organization, Instruction Formats; addressing modes: register, immediate, direct, indirect, indexed; Operations in the instruction set: Arithmetic and Logical, Data Transfer, Control Flow; Types of interrupts.

Unit IV

Memory Hierarchy & I/O Techniques: The need for a memory hierarchy (Locality of reference principle, Memory hierarchy in practice: Cache, main memory and secondary memory, Memory parameters: access/ cycle time, cost per bit); Main memory (Semiconductor RAM & ROM organization, memory expansion, Static &

dynamic memory types); Cache memory, mode of transfer, DMA (Direct memory transfer).

Text and Reference Books:

1. Mano, M. Morris, Digital Logic and Computer Design, Prentice Hall of India Pvt. Ltd., 1981.
2. M. Morris Mano, Computer System Architecture, Prentice Hall of India Pvt. Ltd., 1993.
3. Milles J. Murdocca, Vincent P. Heuring, Computer Architecture and Organization, An Integrated Approach,JohnWiley& Sons Inc., 2007.
4. William Stallings, 10th edition, Computer Organization and Architecture, Prentice Hall, 2016.
5. Heuring, V.P., Jordan, H.F., Computer Systems Design and Architecture, Addison Wesley,1997.
6. R.P Jain, Modern Digital Electronics, 3rdEdition , Tata McGraw Hill,, 2003.

Computer Architecture Lab

General Course Information:

Course Code: 24ICA0401P Course Credits: 1 Type: Discipline Specific Courses (DSC) Contact Hours: 2 hours/week Mode: Lab Exam Duration: 3 hours	**Course Assessment Methods (Internal: 10; External: 20)** The Internal and External assessment is based on the level of participation in Lab sessions and the timely submission of Lab experiments /assignments, the performance in Viva-Voce, the quality of the lab file and ethical practices followed. The Internal examination is conducted by the Course Coordinator. The External examination is conducted by External Examiner appointed by the Controller of Examination in association with the Internal Examiner appointed by the Chairperson of the Department.

Pre-lab assignment: Basic knowledge of logic gates, Programming, Computer architecture.

About the Course: This laboratory course on computer architecture helps students to learn how to develop computer architecture. A study of the subject matter presented in this course will enable the student to become familiar with different concept of computer architecture.

Course Outcomes: By the end of the course students will be able to:

CO1. Implement assembly language program in 8085 microprocessors
CO2. Design of simple computer with different instruction sets.
CO3. **Evaluate** different types of memory, their architecture and access methods.

LIST OF EXPERIMENTS

1. Write an 8085 Program to swap two 8-bit numbers.
2. Write a Program to find the largest of two numbers
3. Write an 8085 Program to find the smallest of two numbers
4. Write an 8085 Program to calculate the sum of first ten natural numbers.
5. Write an assembly language program in 8085 microprocessors to find the sum of digits of an 8-bit number.
6. Write an 8085 Program to find the reverse of an 8-bit number
7. Write an 8085 Program to check whether 1-byte number is a palindrome or not. If it is a palindrome display FF otherwise display DD.
8. Write an 8085 Program to check whether a number is ODD or EVEN. If Even no. display DD, if odd no. display FF.
9. Write an 8085 program to count a number of ones in the given 8-bit number.
10. Write an 8085 program to find Addition & Subtraction of two 8 –bit HEX numbers.

Computer Network

General Course Information

Course Code:24ICA0402T Course Credits: 3 Type: DSC Contact Hours: 3 hours/week Mode: Lectures (L) Exam Duration: 3 hours	**Course Assessment Methods (Internal: 20; External: 50)** **Internal Assessment: -** Two minor examinations each of 10 marks, Class Performance measured through percentage of lectures attended (5 marks) Assignment and quiz (5 marks), and end semester examination of 50 marks. **External Examination (Marks: 50): -** The syllabus is divided into four units. For the end semester examination, nine questions are to be set by the examiner. Question No. 1 is compulsory and contains five short answer questions (2 marks each) covering entire syllabus. Rest eight more questions (each question of at least two parts) will be set by giving two questions from each unit of the syllabus. A candidate is required to attempt five questions in all by selecting one question from each unit in addition to compulsory Question No.1. All questions will carry equal marks.

Prerequisite: Basic knowledge of Digital and Analog Communication.

About the Course: This course Computer Networks will help the learners to understand the basic concepts of computer hardware and networking. Learners will gain knowledge about the various topology and techniques which are used in designing networks and communication technology.

Course Outcomes: By the end of the course students will be able to:

CO1. Define: concepts related to network communication and its components.
CO2. Demonstrate: the designing of network models and use of various protocols in their design.
CO3. Apply: network design techniques to develop network models.
CO4. Implement: data transmission and communication between devices and apply the testing of communication process.
CO5. Predict: speed and accuracy of data transmission during communication.

Course Contents

Unit – I

Introduction to Data Communications: Data representation, communication components, data flow; Introduction to computer network: Uses of Computer Networks, Network Devices, Nodes, and Hosts, Types of Computer Networks and their Topologies; Computer Communications and Networking Models: Decentralized and Centralized Systems, Distributed Systems, Client/Server Model; Network Architecture and the OSI Reference Model.

Unit – II

Computer Network Services: Connection-Oriented and Connectionless Services; Switching Techniques: Circuit Switching, Packet Switching, Message Switching; Network Software: Network Design issues and Protocols; Network Applications and Application Protocols; Analog and Digital Communications Concepts: Representing Data as Analog Signals, Representing Data as Digital Signals, Data Rate and Bandwidth, Capacity, Baud Rate; Digital Carrier Systems; Guided and Wireless Transmission Media; Communication Satellites.

Unit –III

Internetworking: Concept of internetworking, IP Addressing: IPV4 and IPV6, Classful Addressing, Classless Addressing, Private IP Addresses, Public IP Addresses, Reserved IP Addresses, Sub-netting; Routing Protocols; ARP, RARP, ICMP, Internet Routing Protocol.

Unit– IV

Network Hardware Components: Connectors, Transceivers, Repeaters, Hubs, Network Interface Cards and PC Cards, Bridge, Switches, Routers, Gateways; **Routing**: Virtual Circuits and Datagrams, Routing Algorithms, Shortest Path Routing, Distance Vector Routing, Link State Routing, Hierarchical Routing, Congestion Control Algorithms;

T**ext and Reference Books:**

1. Michael A. Gallo and William M. Hancock, Computer Communications and Networking Technologies, Course Technology, 2001.
2. Andrew S. Tanenbaum, Computer Networks, 5th Edition, Pearson Education, 2013.
3. James F. Kurose, Keith W. Ross, Computer Networking, Pearson Education, 2013.
4. Behrouz A Forouzan, Data Communications and Networking, McGraw Hill, 2017

Computer Network Lab

General Course Information

Course Code: 24ICA0402P Course Credits: 1 Type: Discipline Specific Courses (DSC) Contact Hours: 2 hours/week Mode: Lab Exam Duration: 3 hours	**Course Assessment Methods (Internal: 10; External: 20)** The Internal and External assessment is based on the level of participation in Lab sessions and the timely submission of Lab experiments /assignments, the performance in Viva-Voce, the quality of the lab file and ethical practices followed. The Internal examination is conducted by the Course Coordinator. The External examination is conducted by External Examiner appointed by the Controller of Examination in association with the Internal Examiner appointed by the Chairperson of the Department.

Prerequisite: Basic knowledge of Programming, Digital and Analog Communication.

About the Course: This course has been designed with aim to provide students with an overview of the concepts and fundamentals of data communication and computer networks. Students learn about various topologies, network devices, routing protocols, firewall amongst other features and devices of Computer Networks.

Course Outcomes: By the end of the course students will be able to:

CO1**.** Demonstrate: various network topologies and networking devices.
CO2. Justify: a particular routing protocol for any implemented data communication networks.
CO3. Construct: network and implement various network protocols.
CO4. Devise: solutions for various routing and switching problems in Computer Networks.
CO5. Create: lab records for the solutions of the assignments.
CO6. Demonstrate: ethical practices, self-learning and team spirit.

List of Experiments/Assignments:

1. Familiarization with networking components and devices: LAN Adapters - Hubs -Switches – Routers etc.
2. Familiarization with transmission media and Tools: Co-axial cable - UTP Cable - Crimping Tool -Connectors etc.
3. Installation and introduction of simulation tools Packet Tracer/ GNS3.
4. Preparing the UTP cable for cross and direct connections using crimping tool.
5. Configuration of network on star topology using Hub.
6. Configuration of network on star topology using Switch.
7. Introduction to various interior and exterior routing protocols.
8. Configuration of RIP protocol on routers to configure a network topology.
9. Implementation EIGRP protocol on router.
10. Implementation OSPF protocol on a larger network.
11. Configuration of TELNET protocol on router for remote access.

Note:
The actual experiments/assignments will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.

Computer Graphics

General Course Information

Course Code: 24ICA0403T Course Credits: 3 Type: DSC Contact Hours: 3 hours/week Mode: Lectures (L) Examination Duration:3 hours	**Course Assessment Methods (Internal: 20; External: 50)** **Internal Assessment: -** Two minor examinations each of 10 marks, Class Performance measured through percentage of lectures attended (5 marks) Assignment and quiz (5 marks), and end semester examination of 50 marks. **External Examination (Marks: 50): -** The syllabus is divided into four units. For the end semester examination, nine questions are to be set by the examiner. Question No. 1 is compulsory and contains five short answer questions (2 marks each) covering entire syllabus. Rest eight more questions (each question of at least two parts) will be set by giving two questions from each unit of the syllabus. A candidate is required to attempt five questions in all by selecting one question from each unit in addition to compulsory Question No.1. All questions will carry equal marks.

Pre-requisites: Programming skills in C/C++ and Data Structures.

About the Course:
This course involves studying graphic techniques, algorithms and imaging models. Moreover, students learn about the techniques for clipping, cropping, representing 2-D and 3-D objects.

Course Outcomes: By the end of the course students will be able to:
CO1. State: basic concepts related to graphics.
CO2. Describe: the principles of creating graphical objects and graphical user interface applications.
CO3. Apply: 2-D and 3-D transformations (rotation, scaling, translation, shearing) on geometric objects.
CO4. Use: different techniques for clipping and filling geometric objects.
CO5. Compare: different graphics algorithms for different geometric objects.
CO6. Create: user-friendly interfaces for computer applications.

Course Content

Unit I

Introduction: History of Computer Graphics (CG), Components of interactive graphics systems, Display devices: Refresh CRT, Color CRT, Plasma Panel displays LCD Panels.
Graphics Primitives: Introduction to computer graphics and application area of Computer Graphics, overview of graphics systems, raster-scan systems, random scan systems and graphics input devices.

Unit II

Output Primitives: Points and Lines, Line Drawing Algorithms: DDA algorithm, Bresenham's Line & circle draw algorithm, Circle drawing algorithms: Polynomial Method. Parametric representation of Cubic Curves, Bezier Curves
2D Transformation: Use of Homogeneous Coordinates Systems, Composite Transformation: Translation, Scaling, Rotation, Mirror Reflection, Rotation about an Arbitrary Point. Clipping and Windowing, Clipping Operations. Line Clipping Algorithms: The Mid-Point subdivision method, Cohen-Sutherland Line Clipping Algorithms, Polygon Clipping, Sutherland Hodgeman Algorithms, Text Clipping.

Unit III

3-D Object Representation: Polygon surfaces, quadric surfaces, spline representation, Hermite curve, Bezier curve and B-Spline curves, Bezier and B-Spline surfaces. Basic illumination models, polygon-rendering methods.

Unit IV

3-D Geometric Transformations: Translation, rotation, scaling, reflection and shear transformations, composite transformations. 3-D Viewing: Viewing pipeline, viewing coordinates, view volume and general projection transforms and clipping.

Text and Reference Books:

1. James D. Foley, Andeies van Dam, Stevan K. Feiner and Johb F. Hughes, Computer Graphics Principles and Practices, second edition, Addision Wesley, 2000.
2. Pradeep K Bhatia, Computer Graphics, 3rd edition, I K International Pub, New Delhi, 2013.
3. Donald Hearn and M. Pauline Baker : Computer Graphics, PHI Publications.
4. Plastock : Theory & Problem of Computer Gaphics, Schaum Series.
5. Foley & Van Dam : Fundamentals of Interactive Computer Graphics, Addison-Wesley.
6. Newman : Principles of Interactive Computer Graphics, McGraw Hill.
7. Tosijasu, L.K. : Computer Graphics, Springer-Verleg.

Computer Graphics Lab

General Course Information

Course Code: 24ICA0403P Course Credits: 1 Type: DSC Lab. Course Mode: Lab practice and assignments Contact Hours: 2 hours / week	**Course Assessment Methods (Internal: 10; External: 20)** The Internal and External assessment is based on the level of participation in Lab sessions and the timely submission of Lab experiments /assignments, the performance in Viva-Voce, the quality of the lab file and ethical practices followed. The Internal examination is conducted by the Course Coordinator. The External examination is conducted by External Examiner appointed by the Controller of Examination in association with the Internal Examiner appointed by the Chairperson of the Department.

Pre-requisites: Knowledge of C/C++ and Data Structures.

About the Course: This lab course provides opportunity to students to implement various algorithms for graphics e.g. drawing lines, circles and ellipses. In addition, students learn about various transformation techniques.

Course Outcomes: By the end of the course students will be able to:

CO1. Implement: various graphics algorithms for drawing and filling of geometric objects.
CO2. Demonstrate: transformation of geometric objects.
CO3. Compare: strengths and weaknesses of various graphics algorithms.
CO4. Design: algorithms for creating scenes like flying a kite and solar eclipse.
CO5. Create: lab assignment record that includes problem definitions, solutions and conclusions.
CO6. Demonstrate: use of ethical practices, self-learning and team spirit.

List of experiments/assignment

List of Programs:

1. Implement a program to define and use basic graphics primitives such as points, lines, and polygons.
2. Implement a program to draw a line using the Digital Differential Analyzer (DDA) algorithm.
3. Create a program to draw a line using Bresenham's line drawing algorithm.
4. Write a program to draw a circle using the Midpoint Circle algorithm.
5. Implement a program to draw a circle using Bresenham's circle drawing algorithm.
6. Create a program to perform 2D transformations: translation, rotation, and scaling on a given set of points.
7. Write a program to perform mirror reflection of a 2D shape across the x-axis and y-axis.
8. Create a program to draw basic two-dimensional objects like rectangles, triangles, and polygons using inbuilt functions.
9. Write a program to apply a shearing transformation to a 2D object (like a square or triangle) and display the original and transformed objects.
10. Write a program to implement line clipping using the Cohen-Sutherland line clipping algorithm.
11. Write a program to perform 3D transformations: translation, rotation, and scaling on a set of 3D points.

Minor / Vocational Courses:

Data Structure & Algorithm

General course information

Course Code: 24VOC0421T Course Credits: 2 Type: Discipline Specific Course (DSC) Contact Hours: 2 hours/week Mode: Lectures (L) Exam Duration: 3 hours	**Course Assessment Methods:** **Max. Marks: 50 (Internal: 15; External: 35)** **Internal Assessment: -** Two minor examinations each of 10 marks, Class Performance measured through percentage of lectures attended (3 marks) Assignment and quiz (2 marks) **External Examinations: -** The end semester examination will be of 35 marks. The examiner is required to set five questions in all. Question No. 1 will be compulsory consisting of five short questions covering entire syllabus consisting 3 marks each. In addition to this four more questions (each questions may be of two part) will be set consisting of two questions from each unit. The students is required to attempt three question in all selecting one questing from each unit consisting of 10 marks each in addition Question No. 1 is compulsory.

Pre-requisites: Elementary Programming skills in C.

About the Course: Data Structure and Algorithms is a core and an essential course for every graduate in Computer Science and Engineering. This course introduces data structures like arrays, linked lists, trees and graphs etc. and various operations to be implemented on these data structures for solving real world problems. It includes various sorting and searching algorithms as well. Further, it incorporates complexity analysis of algorithms implemented on various data structures.

Course Outcomes: By the end of the course, a student is expected to:

CO1. Describe various types of data structures and operations that can be implemented on these data structures.
CO2. Demonstrate the use of various data structures and their related operations.
CO3. Apply data structure to solve computational problems.
CO4. Compare the suitability of alternative data structures and prescribed operations for various problemsituations.
CO5. Defend solutions with respect to effective storage of data and efficiency of the required operations forsolving real world problems.

Course Contents

Unit I

Introduction to data structures and their types, Abstract data types, Linear lists: Arrays and linked lists: memory representations, implementing operations like traversing, searching, inserting and deleting etc. Applications of arrays and linked lists. Representing sets and polynomials using linked lists.

Stack and Queue: Static and linked implementations, Operations and Applications. Circular queues, Tress, Binary trees and related terminology, Tree traversals (Recursive), Threaded Binary Trees, Binary Search Trees implementation and operations, Priority queues.

.

Unit II

Height Balanced or AVL trees and B trees. Sequential and binary search, Sorting algorithms: Bubble sort, Selection sort, Insertion sort, Quick sort, Merge sort, Count sort, Heap sort, Comparison of searching and sorting techniques based on their complexity analysis, Time and space complexity of algorithms: Asymptotic analysis, Big O, Omega, Theta notations.

Text and Reference Books:

1. Aho, A. V., Ullman, J. D., and Hopcroft, J. E., *Data Structures and Algorithms*, Addison-Wesley, 1983.
2. Cormen, T. H., Leiserson, C. E., Rivest, R. L. and Stein, C., *Introduction to Algorithms*, MIT Press, 2009.
3. Robert L. Kruse, *Data Structure and Program Design in C*, Pearson Education India, 2007.
4. Sahni, S., *Data Structures, Algorithms, and Applications in C++*, WCB/McGraw-Hill, 2001.

Data Structure & Algorithm Lab

General Course Information:

Course Code: 24VOC0421P Course Credits: 2 Type: Discipline Specific Courses(DSC) Contact Hours: 4 hours/week Mode: Labs Exam Duration: 3 hours	**Course Assessment Methods (internal: 15; external: 35):** The internal and external assessment is based on the level of participation in lab. Sessions and the timely submission of lab experiments/assignments, the quality of solutions designed for the assignments, the performance in VIVA-VOCE, the quality of lab. File and ethical practices followed. The internal examination is conducted by the course coordinator. The external examination is conducted by external examiner (appointed by the Controller of Examination) in association with the internal examiner appointed by the Chairperson of the Department.

Prerequisite:
Students are expected to have basic knowledge of problem solving and programming constructs and C language.

About the Course:

This lab course involves implementation of basic and advance data structures and various operations on these data structures. The objective of the lab course is to train the students to solve the problems related to data structures and choose the appropriate data structure for solving computational problem efficiently.

Course Outcomes: By the end of the course students will be able to:

CO1. Implement various data structures and the related operations.
CO2. Analyse space and time complexity of algorithms.
CO3. Compare solutions on the basis of the appropriateness of data structure used and the efficiency of the operations implemented.
CO4. Integrate knowledge of data structures to solve real world problems related to data structure and algorithms.
CO5. Create written records for the given assignments with problem definition, design of solution and conclusions.
CO6. Demonstrate ethical practices while solving problems individually or in groups (

List of experiments/assignments

1. Two assignments related to creating and manipulating matrices and linear lists.
2. Two assignments associated with linked list, operations on linked lists and their applications.
3. Two assignments on array and linked implementation of stacks.
4. Two assignments on array and linked implementation of queues.
5. Two assignments on trees and their applications.
6. Two assignments on graphs and their applications.
7. Five assignments on different searching and sorting methods along with their complexity analysis.
8. One assignment on challenging problems on data structures to be given in groups.

Note:

The actual experiments/assignments will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.

Minor Course:

Database Management System

Course Code: 24VOC0521T Course Credits: 2 Type: Discipline Specific Courses (DSC) Contact Hours: 2 hours/week Mode: Lectures (L) Exam Duration: 2.5 hours	**Course Assessment Methods:** **Max. Marks: 50 (Internal: 15; External: 35)** **Internal Assessment: -** Two minor examinations each of 10 marks, Class Performance measured through percentage of lectures attended (3 marks) Assignment and quiz (2 marks) **External Examinations: -** The end semester examination will be of 35 marks. The examiner is required to set five questions in all.. The first Question will be compulsory consisting of five short questions covering entire syllabus consisting three marks each. In addition to this four more questions (each questions may be of two part) will be set consisting of two questions from each unit. The students is required to attempt three question in all selecting one questing from each unit consisting of 10 marks each in addition to Question No. 1 is compulsory.

Prerequisite: Elementary Programming skills and knowledge of basic database concepts and database applications such as SQL, Oracle.

About the Course:
This course includes a detailed coverage of principles of database design and models.

Course Outcomes: By the end of the course students will be able to:

CO1. Describe: Describe fundamental of Database Management System.
CO2. Select: Principles for basic Database Modeling.
CO3. Apply: Apply basic SQL commands for designing queries on Relational Databases.
CO4. Implement: Implement E-R Modeling, E-R diagram on a real-world database application.
CO5. Compare: Contrast various types of Normalization for real-world database

Course Contents

Unit - I

Introduction: Data, Information, Records, Files and Database. Characteristics of Database Management System. DBMS over File Processing System. Advantages and Disadvantages of DBMS. Database Users and various types of DBMS users. Database Administrator (DBA) and responsibilities of DBA. Schema and Instance. Views of Database. Physical and Logical Data Independence. Database Languages. Data Base System Concepts: DBMS Architecture, Various views of Data, Data Models: Hierarchical, Network and Relational Data Models, Entity-Relationship Model: Entity Type, Attributes, Domain, Keys. Integrity Constraints. ER Diagram: Symbolic Notations for Designing ER Diagram, Reduction of E-R Diagram into Tables.

Unit – II

Introduction of SQL: Characteristics of SQL, Basic Structure, DDL Commands, DML Commands, DQL Commands, TCL Commands. Useful Relational Operators, Aggregate Functions, SET operations.
Relational Algebra: Basic Operations: Select, Project, Union, Intersection, Set Difference, and Cartesian product, Join. Relational Algebra vs. Relational Calculus. Relational Database Design: Functional Dependency, Types of Functional Dependency, Database Anomalies, Decomposition, Normalization: 1st Normal Form, 2nd Normal Form, 3rd Normal Form, BCNF Normal Form.

T**ext and Reference Books:**

1. Database System Concepts, Sixth edition, Abraham Silberschatz, Henry F. Korth, S. Sudarshan, McGraw-Hill-2010.
2. Pannerselvam R, Database Management System, 2nd Edition, PHI Learning, 2011.
3. Database Systems: Models, Languages, Design and Application, RamezElmasri, Pearson Education 2014

Database Management System Lab

General Course Information:

Course Code: 24VOC0521P Course Credits: 2 Type: Discipline Specific Courses (DSC) Contact Hours: 4 hours/week Mode: Lab Exam Duration: 3 hours	**Course Assessment Methods (internal: 15; external: 35):** The internal and external assessment is based on the level of participation in lab. Sessions and the timely submission of lab experiments/assignments, the quality of solutions designed for the assignments, the performance in VIVA-VOCE, the quality of lab. File and ethical practices followed. The internal examination is conducted by the course coordinator. The external examination is conducted by external examiner (appointed by the Controller of Examination) in association with the internal examiner appointed by the Chairperson of the Department.

Pre-requisite: Students are expected to have basic knowledge of problem solving and SQL.

About the Course: The Database Management System Lab aims to provide practical exposure to the concepts of Database focusing on hands-on experience with SQL and ORACLE environments. Students will analyze problem solving aspects of the real-life problems and execute DDL, DML and DCL commands to solve the problem.

.**Course Outcomes: By the end of the course students will be able to:**

CO1. Implementation: Creation of database using DDL commands.
CO2. Analysis: Enforce integrity constraints on a database using SQL commands
CO3. Design: Design various database for given schema.
CO4. Execute: Execute Lab assignments for various problems.
CO5. Demonstrate**:** Demonstrate ethical practices, self-learning and team spirit.

List of Experiments/ assignments

1. Use oracle software and login with valid user id and password.
2. Creation of database schema having different fields and data types.
3. Adding various constraints using DDL commands.
4. Alter the schema of the existing database.
5. Delete the schema of the database.
6. Add records in the database using DML command.
7. SQL query for various views of the database.
8. Update and delete existing records in the database using DML command.
9. Using SQL queries for various aggregate functions.
10. Using SQL queries ordering and grouping of records on the database.
11. Using SQL queries for various SET operations on relational database.
12. Using SQL queries for logical operations on relational database.
13. Using various DCL commands on the database.
14. Using various TCL commands on the database.
15. Using SQL queries for creating Join operations.

Note: The actual experiment / assignments will be designed by the course coordinator. One assignment to be done in the groups of two or three students. The assignments must be meet the objectives of the course and the levels of given course outcomes. The list of assignments and schedule of submission will be prepared by course coordinator in the beginning of the semester.

Web Development

General Course Information:

Course Code: 24MIC0621T Course Credits: 3 Type: Skill Enhancement(SEC) Contact Hours: 3 hours/week Mode: Lectures (L) Exam Duration: 2.5 hours	Course Assessment Methods (Internal: 20; External: 50) **Internal Assessment: -** Two minor examinations each of 10 marks, Class Performance measured through percentage of lectures attended (5 marks) Assignment and quiz (5 marks), and end semester examination of 50 marks. **External Examination: -** The syllabus is divided into four units. For the end semester examination, nine questions are to be set by the examiner. Question No. 1 is compulsory and contains five short answer questions of 2 marks each covering entire syllabus. Rest eight more questions (each question of at least two parts) will be set by giving two questions from each of the unit of the syllabus. A candidate is required to attempt five questions in all by selecting one question from each of unit in addition to compulsory Question No.1. All questions will carry equal marks.

Prerequisite:

This course requires a basic understanding of computer fundamentals and familiarity with navigating the internet. No prior knowledge of programming or web development is required, making it accessible to beginners.

About the Course

This course covers fundamental concepts in HTML, CSS, and JavaScript during teaching students learn to design and develop interactive websites with multimedia integration and responsive design principles. They learn to create structured HTML documents, style webpages with CSS, and add dynamic functionality using JavaScript, including event handling, and asynchronous programming techniques.

Course Objective:

CO1. List or describefundamental concepts of HTML, CSS, and JavaScript.
CO2. Apply appropriate elements, properties, and techniques in web development.
CO3. Create well-structured, visually appealing, and interactive webpages.
CO4. Implement forms, styling, and functionality to enhance user experience.
CO5. Utilize modern techniques like responsive design using JSON, XML and DOM manipulation.
CO6. Compare and evaluate different development approaches and tools.

Course Contents

Unit - I

HTML5 Basic: Introduction to HTML Document, Elements and Tags, Text Formatting, Headings and Paragraphs, Adding Line Breaks and Horizontal Rules, Lists (Ordered, Unordered, Definition), Anchor Tags, Linking to External Resources (Images, Style sheets, Scripts). HTML Tables, Forms, Input Types, Textboxes, Radio Buttons, Checkboxes, form submission methods (GET and POST), Dropdown Menus and Select Boxes, Grouping Form Elements with Field Sets and Legends, form Attributes, Embedding Multimedia (Images, Audio, Video).

Unit - II

HTML5 advance: Working with Meta Tags (Description, Keywords, Charset), Comments in HTML, Validation, Responsive Web Design Principles, Semantic Elements (<nav>, <main>, article, section, header, footer), Entities and Special Characters, Document Structure for Search Engine Optimization, Creating Forms with Advanced Features (File Uploads, Hidden Inputs), APIs (Canvas, Geo location, Local Storage) Working with iframes, Embedding External Content (YouTube Videos, Google Maps), Responsive Images (map area) with srcset and sizes Attributes,

Unit – III

Cascading Style Sheets: Introduction to CSS and Basic Styling, Role of CSS in Web Design, Syntax and

Structure of CSS Rules, Selectors and Specificity, Inline Styles, External vs. Internal Style sheets, Box Model (Margin, Border, Padding, Content), Box Elements (Width, Height, Borders, Margins, Padding), Display Properties (Block, Inline, Inline-Block),Positioning Elements (Static, Relative, Absolute, Fixed).

Unit - IV

CSS Advanced Techniques for Styling Text (Font Properties, Text Decoration, Text Alignment), Working with Colors and Backgrounds (Color Values, Background Images, Gradients), Managing Lists and Tables, Styling Links and Navigation Menus, CSS Flexbox/Grid Layouts and Properties, CSS Transitions Properties (Duration, Timing Function, Delay), Debugging and Troubleshooting CSS Issues.

Text and Reference Books:

1. "Learning Web Design: A Beginner's Guide to HTML, CSS, JavaScript, and Web Graphics" by Jennifer Niederst Robbins, 2018, 5th Ed., O'Reilly Media
2. "Head First HTML and CSS" by Elisabeth Robson and Eric Freeman, 2012, O'Reilly Media
3. "Responsive Web Design with HTML5 and CSS" by Ben Frain, 2014, 1st Edition, Packt Publishing.
4. "HTML, CSS, and JavaScript All in One" by Julie C. Meloni and Jennifer Kyrnin, 2020, 3rd Edition, Sams Publishing.
5. “HTML 5 Black Book, Covers CSS 3, JavaScript, XML, XHTML, AJAX, PHP and jQuery”, 2016 2nd Edition, Kindle Edition, Dreamtech Press.

Web Development Lab

General Course Information:

Course Code: 24MIC0621P Course Credits:1 Type: Skill Enhancement Course (SEC-2) Contact Hours: 2 hours/week Mode: Labs Exam Duration: 3 hours	**Course Assessment Methods (Internal: 10; External: 20)** The Internal and External assessment is based on the level of participation in Lab sessions and the timely submission of Lab experiments / assignments, the performance in Viva-Voce, the quality of the lab file and ethical practices followed. The Internal examination is conducted by the Course Coordinator .The External examination is conducted by External Examiner appointed by the Controller of Examination in association with the Internal Examiner appointed by the Chairperson of the Department.

Prerequisite:

To learn web design using HTML require basic computer literacy, familiarity with internet usage, and a foundational understanding of HTML fundamentals such as elements, tags, and attributes.

About the Course:

The objectives of this course are to make:

- This course introduces students to the fundamental principles and practices of web design using HTML.
- It covers essential concepts such as HTML document structure, elements, tags, text formatting, lists, tables, forms, and multimedia embedding.
- It also explores basic CSS for styling web pages and introduces responsive design principles.

Course Outcomes: By the end of the course students will be able to:

CO1. Implementation: Students will be able to implement HTML web pages.
CO2. Analysis: students will develop the ability to analyse and interpret HTML and CSS code to understand its structure.
CO3. Compare: students will compare and contrast different approaches to web design, including the use of HTML and CSS, to achieve specific design goals and user requirements.
CO4. Integrate: students will integrate multimedia elements such as images, audio, and video seamlessly into HTML documents.
CO5. Create: able to create well-designed and visually appealing web pages using HTML and basic CSS styling techniques.
CO6. Demonstrate: student will demonstrate responsive web pages and showcasing their ability to create user-friendly.

Course Contents

List of Experiments/ assignments

1. Basic HTML Document Creation: Create a simple HTML document with necessary elements like <!DOCTYPE>, <html>, <head>, <title>, and <body>.
2. Demonstrate various text formatting elements such as , , <u>, <s>, etc.
3. Create ordered, unordered, and definition lists. Include hyperlinks to external resources using <a> tags.
4. Construct a table with rows and columns. Develop a simple form containing input types like textboxes, radio buttons, checkboxes, dropdown menus, and text areas.
5. Embed images, audio, and video files within an HTML document using appropriate tags.
6. Add meta tags for description, keywords, and charset. Include comments within the HTML document for better readability.
7. Validate HTML code using W3C validator. Create a responsive webpage using media queries and viewport meta tags.
8. Implement semantic elements such as <nav>, <main>, <article>, <section>, <header>, and <footer> for better SEO.
9. Implement advanced form features like file uploads and hidden inputs. Utilize HTML5 APIs like Canvas, Geolocation, and Local Storage.
10. Embed external content such as YouTube videos and Google Maps using iframes.
11. Apply basic CSS styles to HTML elements including font properties, colors, backgrounds, and text alignment.
12. Demonstrate the concept of margin, border, padding, and content. Style box elements with width, height, borders, margins, and padding.
13. Understand and implement various display properties like block, inline, and inline-block. Experiment with positioning elements statically, relatively, absolutely, and fixedly.
14. Apply advanced CSS techniques for styling text such as font properties, text decoration, and alignment. Experiment with different color values, background images, and gradients.
15. Style lists and tables with custom designs. Design navigation menus and style hyperlinks accordingly.
16. Implement CSS Flexbox and Grid layouts for better page structuring. Add transitions with properties like duration, timing function, and delay to enhance user experience.
17. Debug and troubleshoot CSS issues using browser developer tools and techniques like validation and code review.

Note: The actual experiment / assignments will be designed by the course coordinator. One assignment to be done in the groups of two or three students. The assignments must be meet the objectives of the course and the levels of given course outcomes. The list of assignments and schedule of submission will be prepared by course coordinator in the beginning of the semester.

Data Analytics Using R

General Course Information

Course Code: 24VOC0621T Course Credits: 2 Type: DSC Contact Hours: 2 hours/week Mode: Lectures (L) Examination Duration: 3 Hours.	**Course Assessment Methods (Internal: 20; External: 50)** **Internal Assessment: -** Two minor examinations each of 10 marks, Class Performance measured through percentage of lectures attended (5 marks) Assignment and quiz (5 marks), and end semester examination of 50 marks. **External Examination (Marks: 50): -** The syllabus is divided into four units. For the end semester examination, nine questions are to be set by the examiner. Question No. 1 is compulsory and contains five short answer questions (2 marks each) covering entire syllabus. Rest eight more questions (each question of at least two parts) will be set by giving two questions from each unit of the syllabus. A candidate is required to attempt five questions in all by selecting one question from each unit in addition to compulsory Question No.1. All questions will carry equal marks.

Pre-requisites: Basic programming skills, Probability and Statistics

About the Course: Data analytics is a growing and stimulating field that turns data into valuable insights. This course includes programming in R for acquiring, cleaning, visualizing and analyzing data. In addition, it also involves predictive modeling. This course will introduce students to the basic principles, tools and the craft for devising solutions for problems that come in the domain of data science. The emphasis of the course is on integration and synthesis of concepts and their applications for effective engineering solutions.

Course Outcomes: By the end of the course students will be able to:
CO1. outline concepts related to R programming and data analysis. (LOTS: Level 1: Remember)
CO2. explain the basic concepts and tools that are used to solve problems in data analytics. (LOTS: Level 2: Understand)
CO3. interpreting results of descriptive and inferential statistics. (LOTS: Level 2: Understand)
CO4. apply R programming for reading, cleaning, visualizing and analysing data. (LOTS: Level 3: Apply)
CO5. analyse the trends in data through exploratory data analysis. (HOTS: Level 4: Analysc)
CO6. devise solutions for descriptive and predictive modeling. (HOTS: Level 6: Create) Course Content

Unit I

Introduction to R programming: Data types or objects in R. Creating and manipulating objects like factors, vectors and matrices, lists and data frames, Subsetting matrices and data frames, Vectorized operations for vectors and matrices and data frames.
Control structure in R: If-else statements, for and while loops, loop functions like lapply. apply, sapply and mapply. writing user defined functions in R. Getting data in and out of R.

Unit II

Doing basic descriptive statistics: Data types for data analysis and their mapping to R objects, Mean, Median, Mode, Quantiles, Five-point summary, Variance, Correlation and Covariance, normal distribution, uniform distribution using R
Exploratory Data Analysis: Visualizing data through various plots and charts (bar charts, histogram, frequency polygon, scatter plot, box plots etc.

Text and Reference Books:
1. Hadley Wickharn and Garrett Grolemund., R for Data Sclence Import, Tidy. Transform and model Data, O'Reilly, 2017.
2. Roger D. Peng, R Programming for Data Sclence, Lean Publishing, 2015.
3. Paul Teeter, R Cookbook, O'Reilly, 2011
4.W. N. Venables, D. M. Smith and the R core Tean, An introduction to R, Notes on R: A Programming Environment for Data Analysis and Graphics, version 3.3.2, 2016.
5. Michael J. Crawley, Statistics, An Introduction using R, Second edition, John Wiley, 2015

Data Analytics Using R Lab

General Course Information:

Course Code: 24VOC0621P Course Credits: 2 Type: Discipline Specific Courses (DSC) Contact Hours: 2 hours/week Mode: Lab Exam Duration: 3 hours	**Course Assessment Methods (Internal: 10; External: 20)**: The Internal and External assessment is based on the level of participation in Lab sessions and the timely submission of Lab experiments /assignments, the performance in Viva-Voce, the quality of the lab file and ethical practices followed. The Internal examination is conducted by the Course Coordinator .The External examination is conducted by External Examiner appointed by the Controller of Examination in association with the Internal Examiner appointed by the Chairperson of the Department.

Pre-requisites: Basic programming skills and knowledge of statistics

Course Outcomes: By the end of the course students will be able to:

About the Course: This lab course on data science involves a rigorous training on R programming. It incorporates solving problems related to data science in statistical and predictive modeling framework. The objective of the lab course is to equip the students to solve the practical data science problems related to intelligent data analysis using R.

Course Outcomes: By the end of the course students will be able to: 4 Mode: Lab. practice and assignments
CO1. implement R programming concepts for data analysis. (LOTS: Level 3: Apply)
CO2. analyse the trends in data through exploratory data analysis. (HOTS: Level 4: Analyse)
CO3. evaluate the results of descriptive and inferential statistics. (HOTS: Level 5: Evaluate)
CO4. devise solutions for descriptive and predictive modeling. (HOTS: Level 6: Create) 5
CO5. create lab. Record of assignment solutions that include problem definition, solutions and interpretation of results. (HOTS: Level 6: Create) 6
CO6. demonstrate use of ethical practices, independent end enquiry and self-learning, and team spirit to solve unseen problems. (LOTS: Level 3: Apply)

LIST OF EXPERIMENTS
1. Install R studio and explore its GUI. Explore the base R package- datasets. See the list of datasets available in the package.
2. Write description for the following datasets:
Iris Hair Eye Color
Air quality
Mtcars
3 Two assignments on the use of control, looping statements and user defined functions.
4. Three assignments related to creating and manipulating objects like vectors, factors, matrices, lists and data frames.
5.Two assignment on finding descriptive statistics and exploratory data analysis
6.Two assignments on making different charts and writing the finding on the basis of these charts.
7.Two assignments on hypothesis testing for descriptive and inferential statistics,
8.Two assignments on predictive modeling using R packages in groups of two or three students depending on the size of the assignment.