

Deccan Education Society's

**Kirti M. Doongursee College of  
Arts, Science and Commerce  
(AUTONOMOUS)**



Affiliated to

**UNIVERSITY OF MUMBAI**

Syllabus for  
Program: Bachelor of Science  
Course: First Year  
Subject: Information Technology

Choice Based Credit System (CBCS)  
with effect from  
Academic Year 2023-2024

## **PROGRAM OUTCOMES**

<b>PO</b>	<b>Description</b>
A student completing Bachelor's Degree in Science (Information Technology) Program will be able to	
PO1	To think analytically, creatively and critically in developing robust, extensible and highly maintainable technological solutions to simple and complex problems.
PO2	To apply their knowledge and skills to be employed and excel in IT professional careers and/or to continue their education in IT and/or related postgraduate programmes.
PO3	To be capable of managing complex IT projects with consideration of the human, financial and environmental factors.
PO4	To work effectively as a part of a team to achieve a common stated goal in a professional manner.
PO5	To adhere to the highest standards of ethics, including relevant industry and organizational codes of conduct.
PO6	To communicate effectively with a range of audiences both technical and non-technical.
PO7	To develop an aptitude to engage in continuing professional development
PO8	To understand corporate approach towards the projects in a research oriented qualitative manner

**Deccan Education Society's  
Kirti M. Doongursee College  
(Autonomous) Proposed**

**Curriculum as per NEP-2020**

**Year of implementation- 2023-2024**

**Name of the Department-Information Technology**

<b>Semester</b>	<b>Course Code</b>	<b>Course Title</b>	<b>Vertical</b>	<b>Credit</b>
<b>I</b>	<b>K23USITMJ111</b>	<b>Programming principles with C</b>	<b>Major</b>	<b>2</b>
	<b>K23USITMJ112</b>	<b>Fundamentals of Database Management Systems</b>	<b>Major</b>	<b>2</b>
	<b>K23USITMJP11</b>	<b>IT practical 1 (Part A+B)</b>	<b>Major</b>	<b>2</b>
	<b>K23USITOE131</b>	<b>Basics of AI &amp; ML</b>	<b>OE</b>	<b>2</b>
	<b>K23USITOE132</b>	<b>Cyber Law</b>	<b>OE</b>	<b>2</b>
	<b>K23USITVC141</b>	<b>Web Programming</b>	<b>VSC</b>	<b>2</b>
	<b>K23USITSC151</b>	<b>Linux Operating System</b>	<b>SEC</b>	<b>2</b>
<b>II</b>	<b>K23USITMJ211</b>	<b>Object Oriented Programming with C++</b>	<b>Major</b>	<b>2</b>
	<b>K23USITMJ212</b>	<b>Data Structures</b>	<b>Major</b>	<b>2</b>
	<b>K23USITMJP21</b>	<b>IT practical 2 (Part A+B)</b>	<b>Major</b>	<b>2</b>
	<b>K23USITMR221</b>	<b>Discrete Mathematics</b>	<b>Minor</b>	<b>2</b>
	<b>K23USITOE231</b>	<b>Introduction to Data Science</b>	<b>OE</b>	<b>2</b>
	<b>K23USITOE232</b>	<b>E-Commerce</b>	<b>OE</b>	<b>2</b>
	<b>K23USITVC241</b>	<b>Advance Web Programming</b>	<b>VSC</b>	<b>2</b>
	<b>K23USITSC251</b>	<b>Python Programming</b>	<b>SEC</b>	<b>2</b>

# **SEMESTER- I**

<b>Course Code</b>	<b>MAJOR SEM – I - Programming principles with C</b>	<b>Credits</b>	<b>Lectures /Week</b>
<b>K23USITMJ111</b>	<b>Paper I</b>	<b>2</b>	<b>2</b>
<b>Course Outcomes:</b>			
After successful completion of this course, students would be able to			
<ul style="list-style-type: none"> <li>• To develop the logical ability of the student.</li> <li>• Basic concepts to be cleared using suitable examples.</li> <li>• Different approach towards the problem.</li> <li>• To handle the errors and find suitable solutions.</li> </ul>			
<b>Unit</b>	<b>Topics</b>	<b>No of Lectures</b>	
<b>I</b>	<p>Introduction: Algorithms, History of C, Structure of C Program. Program Characteristics, Compiler, Linker and preprocessor, pseudo code statements and flowchart symbols, Desirable program characteristics. Program structure. Compilation and Execution of a Program, C Character Set, identifiers and keywords, data types and sizes , constants and its types, variables, Character and character strings, typedef, typecasting</p> <p>Type of operators: Arithmetic operators, relational and logical operators, Increment and decrement operators, assignment operators, the conditional operator, Assignment operators and expression, Precedence and order of Evaluation and C Preprocessor</p> <p>Control Flow: Statements and Blocks, If-Else, Else-If, Switch, Loops- While and For Loops-Do-while, Break and Continue, Goto and Labels. Array concept and programs on array traversal, and multidimensional arrays. Character arrays and working with strings.</p> <p>Functions and Program Structure: Basics of functions. User defined and Library functions, Function parameters, Return values, Recursion</p>	<b>15</b>	
<b>II</b>	External variables, Scope Rules, Standard Input and Output, Formatted Output- printf() and Formatted Input- scanf(), Line Input and Output, Error Handling- StdErr and Exit, Header Files	<b>15</b>	

	<p>Pointer: Pointer and Addresses, Pointer and Function Arguments, Pointer and Arrays, Address, Arithmetic, Character Pointers and Functions, Pointer Arrays: Pointers and Functions, Command-line Arguments, Pointers to Functions, Dynamic memory allocation.</p> <p>Structures: Basics of structures, Structures and Functions, Arrays of Structures, Pointers to Structures, Unions, Bit-fields, File management in C: Defining and Opening file, Closing a file, Input / Output operations on file, Error handling in C, Random access to files, Command line arguments.</p>	
--	---	--

**Textbooks:**

1. Let us C Yashwant P. Kanetkar BPB publication
2. Programming in ANSI C by E.Balagurusamy, Tata McGraw-Hill
3. Programming Language by Brian W. Kernighan and Denis M. Ritchie, PHI publication.

**Additional References:**

Mastering C by K R Venugopal, Tata McGraw-Hill  
Programming with C by Byron Gottfried, Tata McGRAW-Hill

<b>Course Code</b>	<b>MAJOR SEM – I – Fundamentals of Database Management Systems</b>	<b>Credits</b>	<b>Lectures /Week</b>
<b>K23USITMJ112</b>	<b>Paper II</b>	<b>2</b>	<b>2</b>
<b>Course Outcomes:</b>			
After successful completion of this course, students would be able to			
<ul style="list-style-type: none"> <li>• Define and describe the core components of relational database management systems.</li> <li>• Understand creation, manipulation and querying of data in databases using MySQL.</li> <li>• Illustrate ER-models to represent simple database application scenarios.</li> <li>• Relate the fundamental ideas of MySQL, the entity-relationship model, relational database design, relational algebra, and the relational data model.</li> </ul>			
<b>Unit</b>			
<b>Unit</b>	<b>Topics</b>	<b>No of Lectures</b>	
<b>I</b>	<p>Basic Database Concepts, Terminology, and Architecture: Structure of DBMS, Types of Database Management Systems, Differences between Relational and other Database Models.</p> <p>Relational Algebra: Definition of Algebra, Relations as Sets, Operations: SELECT, PROJECT, JOIN, etc.</p> <p>Data and Conceptual Modelling: Entity, Attributes, Relations, Schemas, Constraints, Queries, and Updates, Conceptual vs. Physical Modeling, Database Keys, Data modelling using the Entity Relationship model (ER).</p> <p>Database design: Use of UML diagrams as an Aid to Database Design specification</p>	<b>15</b>	
<b>II</b>	<p>Functional Dependencies and normalization: Types of Functional Dependencies, Normalization Theory: 2NF, 3NF, BCNF, 4NF, 5NF.</p> <p>Introduction to SQL:</p>	<b>15</b>	

	<p>Specifying Tables, Data Types, Constraints; Complex queries, triggers, views, joining database tables and schema modification.</p> <p>Indexing: Query Processing and optimization. File structure, hashing and indexing.</p> <p>Transaction processing and Database Recovery Techniques: Properties and States of Transaction, Deadlock, Methods to handle deadlock in a system, Database Recovery Management.</p>	
--	---	--

**Textbooks:**

- Fundamentals of Database systems, Ramez Elmasri, Shamkant B Navathe, Pearson, 6th Edition
- Database Systems: Design implementation and management, Carlos Coronel, Steven Morris, Peter Rob, Cengage Learning, 9th Edition, 2010
- Database Management Systems, Ramakrishnam, Gehrke, McGraw-Hill, 2007.

**Additional References:**

- Murach's MySQL, Joel Murach, Murach, 2012.
- Beginning MySQL, Robert Sheldon, Geoff Moes, Wrox Press, 2005.



Course Code	MAJOR SEM I Practicals	Credits	Lectures /Week
K23USITMJP11	IT Practical 1 (Part A+B)	2	4

### Part A- Programming Principles with C Practical

#### Course Outcomes:

Learners will be able to,

- Write the programs for Developing applications.
- Understand of a functional hierarchical code organization
- Work with textual information, characters and strings.
- Understand the differences between syntax errors, runtime errors, and logic errors

<b>1</b>	<p>a. Write an algorithm and draw flowchart for Area of circle.</p> <p>b. Write an algorithm and draw flowchart to print the given no. is even or odd.</p> <p>c. Write an algorithm and draw flowchart to print 1 to 10 numbers.</p> <p>d. Write an algorithm and draw flowchart for sum of 1 to 5 numbers.</p> <p>e. Write an algorithm and draw flowchart to compute the addition of digits of a given number</p>
<b>2</b>	<p>a. Write a program using while loop to reverse the digits of a number.</p> <p>b. Write a program to calculate the factorial of a given number.</p> <p>c. Write a program to find the roots of quadratic equation.</p> <p>d. Write a program to print the Fibonacci series..</p>
<b>3</b>	<p>a. Write a program in C to check entered character vowel or consonant</p> <p>b. Write a program to C program to print day name of week using switch-case.</p> <p>c. Write a program to read three values from keyboard and print out the largest of them without using if statement.</p>
<b>4</b>	<p>a. Write a program to print the pattern of asterisks as shown below :</p> <pre>* * * * * * * * * *</pre> <p>b. Write a program to print the pattern of asterisks as shown below :</p> <pre>* * * * * * * * * * * * *</pre> <p>c. Write a program to print Floyd's Triangle.</p>

5	<p>a. Write a program to print area of square using function.</p> <p>b. Write a program using recursive function.</p> <p>c. Write a program to square root, abs() value using function.</p> <p>d. Write a program using goto statement.</p>
6	<p>a. Write a program to print rollno and names of 10 students using array.</p> <p>b. Write a program to read a matrix of size m*n.</p> <p>c. Write a program to sort the elements of array in ascending or descending order.</p>
7	<p>a. Write a program to extract the portion of a character string and print the extracted part.</p> <p>b. Write a program to find the given string is palindrome or not.</p> <p>c. Write a program to using strlen(), strcmp() function.</p>
8	<p>a. Write a program to display the values using different data types and its address using pointer.</p> <p>b. Write a program to perform addition and subtraction using pointer.</p>
9	<p>a. Write a program to copy the contents of the file from one file into other.</p> <p>b. Write a program to print the structure using</p> <ul style="list-style-type: none"> <li>• Title</li> <li>• Author</li> <li>• Subject</li> <li>• Book ID</li> </ul> <p>Print the details of two students.</p>
10	<p>Create a mini project on “Bank management system” . The program should be menu driven.</p>

<b>Part B- Fundamentals of Database Management System Practicals</b>	
<p>Course Outcomes:</p> <p>After successful completion of this course, students would be able to</p> <ul style="list-style-type: none"> <li>• List common database modeling techniques, such as entity-relationship diagrams (ERDs) or UML class diagrams.</li> <li>• Explain the relational model and its key components, such as tables, columns, and relationships.</li> <li>• Write SQL queries to retrieve, manipulate, and analyze data stored in a database.</li> <li>• Examine the effectiveness of database management solutions in terms of data integrity, performance, and scalability.</li> </ul>	
1	<p>Draw E-R diagram and convert entities and relationships to relation table for a given scenario: (Bank, College)</p>
2	<p>Write relational algebra queries for a given set of relations.</p>
3	<p>Defining data Using CREATE, ALTER, DROP, TRUNCATE and RENAME statements</p>

<b>4</b>	Manipulating data Using INSERT, UPDATE, DELETE and SELECT statements
<b>5</b>	Creating and managing the tables Creating table with constraints: NOTNULL, UNIQUE, PRIMARY KEY, FOREIGN KEY
<b>6</b>	Restricting and sorting data Using DISTINCT, IN, AS, SORT, LIKE, ISNULL, OR Using Group By, Having clause, Order By clause
<b>7</b>	Aggregate and Mathematical functions: AVG, MIN, MAX, SUM, COUNT, ABS, SQRT, ROUND, TRUNCATE, SIGN, POWER, MOD, FLOOR, CEIL
<b>8</b>	Views and Joins: For a given set of relation tables perform the following Creating view, Dropping view, Selecting from a view
<b>9</b>	Database trigger Using CREATE OR REPLACE TRIGGER
<b>10</b>	Create and Drop Index

**All the practicals will be completed within 60 Lectures**

<b>Course Code</b>	<b>OPEN ELECTIVE SEM – I – Basics of AI &amp; ML</b>	<b>Credits</b>	<b>Lectures /Week</b>
<b>K23USITOE131</b>	<b>Paper I</b>	<b>2</b>	<b>2</b>
<b>Course Outcomes:</b>			
After successful completion of this course, students would be able to			
<ul style="list-style-type: none"> <li>● Give the risk and benefits of AI</li> <li>● Explain the History of Artificial Intelligence and the future of AI</li> <li>● Apply Machine Learning concept to solve problems</li> <li>● Compare and contrast the Machine Learning task</li> </ul>			
<b>Unit</b>			
<b>Unit</b>		<b>Topics</b>	<b>No of Lectures</b>
<b>I</b>	What Is AI? The Foundations of Artificial Intelligence, The History of Artificial Intelligence, Risks and Benefits of AI, Future of AI		<b>15</b>
<b>II</b>	What is Machine Learning? Motivations for Machine Learning, Why Machine Learning? Job Opportunities for Machine Learning, Future of ML Machine Learning: Machine learning, Examples of Machine Learning Problems, Structure of Learning, learning versus Designing, Training versus Testing, Characteristics of Machine learning tasks, Predictive and descriptive tasks.		<b>15</b>
<b>References:</b>			
<ul style="list-style-type: none"> <li>● Artificial Intelligence: A Modern Approach, 4th US ed., Stuart Russell and Peter Norvig</li> <li>● Artificial Intelligence Elaine Rich, Kevin Knight, Tata McGraw Hill 3rd 2017</li> <li>● Machine Learning: The Art and Science of Algorithms that Make Sense of Data Peter Flach Cambridge University Press 1st 2012</li> <li>● Introduction to Machine Learning, Ethem Alpaydin PHI 2nd 2013</li> </ul>			

<b>Course Code</b>	<b>OPEN ELECTIVE SEM – I – Cyber Law</b>	<b>Credits</b>	<b>Lectures /Week</b>
<b>K23USITOE132</b>	<b>Paper II</b>	<b>2</b>	<b>2</b>
<p><b>Course Outcomes:</b>  After successful completion of this course, students would be able to</p> <ul style="list-style-type: none"> <li>• Make Learner Conversant With The Social And Intellectual Property Issues Emerging From ‘Cyberspace.</li> <li>• Explore The Legal And Policy Developments In Various Countries To Regulate Cyberspace;</li> <li>• Develop The Understanding Of Relationship Between Commerce And Cyberspace</li> <li>• Give Learners In Depth Knowledge Of Information Technology Act And Legal Frame Work Of Right To Privacy, Data Security And Data Protection.</li> </ul>			
<b>Unit</b>	<b>Topics</b>	<b>No of Lectures</b>	
<b>I</b>	Introduction to Cyber World, Introduction to Indian Cyber Law , Distinction between Cyber Crime and Conventional Crime , Cyber Criminals and their Objectives , Kinds of Cyber Crime- cyber stalking; cyber pornography; forgery and fraud; crime related to IPRs; Cyber terrorism; computer vandalism etc.	<b>15</b>	
<b>II</b>	Overview of General Laws and Procedures in India, Penalties & Offences under the IT Act, 2000	<b>15</b>	
<p><b>Textbooks:</b></p> <ul style="list-style-type: none"> <li>• Cyber law –The Indian perspective by Pavan Duggal</li> <li>• The Indian Cyber Law with The Information Technology Act 2000 by Suresh T Viswanathan Edition 2022</li> </ul> <p><b>Additional References:</b></p> <ul style="list-style-type: none"> <li>• CYBER FRAUDS, CYBERCRIMES &amp; LAW IN INDIA by Pavan Duggal</li> </ul>			

<b>Course Code</b>	<b>VOCATIONAL SKILL COURSE SEM – I - Web Programming</b>	<b>Credits</b>	<b>Lectures /Week</b>
<b>K23USITVC141</b>	<b>Paper I</b>	<b>2</b>	<b>2</b>

**Course Outcomes:**

After successful completion of this course, students would be able to

- Recall key concepts and terminology related to web programming and apply them in practical exercises.
- Recognize the fundamental web programming concepts and their application in real-world scenarios.
- Use web programming languages and tools to create dynamic, interactive, and responsive web applications.
- Evaluate the effectiveness of web programming solutions in terms of functionality, usability, and security.

<b>Unit</b>	<b>Topics</b>	<b>No of Lectures</b>
<b>I</b>	<p><b>Internet and the World Wide Web:</b> What is Internet? Applications of Internet, E-mail, Telnet, FTP, Internet Service Providers, Domain Name Server, Internet Address, World Wide Web (WWW): World Wide Web and its Evolution, Uniform Resource Locator (URL), Browsers, Search Engine, Web Server, HTTP Protocol.</p> <p><b>HTML5:</b> Introduction, Formatting Text by using Tags, Using Lists, Creating Hyperlinks and Bookmarks, Defining Metadata about an HTML Document, Redirecting to another URL.</p> <p><b>CSS:</b> Implementing Styles using CSS – Stylesheets, Formatting Text, Paragraphs and Links using CSS, CSS Selectors</p> <p><b>HTML Page Layout:</b> Using Layout Elements, Semantic Elements, Creating, Positioning and Formatting Divisions, Floating Divisions next to each other</p>	<b>15</b>

	<p><b>HTML Media:</b></p> <p>Embedding Images, Creating Client-side and Server-side Image Map, Embedding audio and video on web page.</p> <p><b>Tables and Forms:</b></p> <p>Creating Simple Table, Table Dimension, Merging Table Cells, Formatting Tables: Applying Borders, Background and Foreground fills, Changing Cell Padding, Spacing and Alignment, Collecting user input with HTML Forms.</p>	
<p><b>II</b></p>	<p><b>JavaScript:</b></p> <p>Introduction, Difference between Client-side and Server-side Scripting, JavaScript Variables and Constants, Data Types, JavaScript Operators, Comments, Functions, JavaScript Objects, Dialog Boxes</p> <p><b>Statements:</b></p> <p>Conditional Statements – if else, switch, Loops – while, do while, for, for in, for of, Loop Control – break, continue, labels</p> <p><b>JavaScript Objects:</b> User-defined Objects, with Keyword, Native Objects – Array, String, Date, Math, Number, RegExp</p> <p><b>Events and Event Handlers:</b> HTML Events, DOM Events, DOM Event Listeners</p> <p><b>PHP:</b></p> <p>Introduction, Server-side Scripting, PHP Syntax and Comments, Variables and Constants, Data Types, Looping, Functions, PHP Form Handling, PHP Arrays, PHP Strings, PHP GET and POST, PHP RegEx, Basic PHP Errors.</p> <p><b>Advanced PHP:</b></p> <p>PHP Sessions, PHP Cookies, Validating and Sanitizing Data with PHP Filters, PHP mail function.</p> <p><b>MySQL:</b></p> <p>Why MySQL? Connect to MySQL, Creating Database and Tables, Selecting Data, Updating Data, Deleting Data, Limiting Data.</p>	<p><b>15</b></p>
<p><b>Textbooks:</b></p>		

- The Complete Reference HTML & CSS, Thomas A. Powell, McGraw Hill, 5th edition
- Step by Step HTML5, Faithe Wempen, Microsoft Press, 2011
- The Complete Reference JavaScript, Thomas A. Powell & Fritz Schneider, McGraw Hill, 3rd edition, 2012

**Additional References:**

- Learning PHP, MySQL, JavaScript, CSS & HTML5, Robin Nixon, O'Reilly, 3rd edition, 2018
- PHP & MySQL Novice to Ninja, Tom Butler, SPD, 7th edition, 2022



<b>Course Code</b>	<b>SKILL ENHANCEMENT COURSE SEM – I - Linux Operating System</b>	<b>Credits</b>	<b>Lectures /Week</b>
<b>K23USITSC151</b>	<b>Paper I</b>	<b>2</b>	<b>2</b>
<b>Course Outcomes:</b>			
After successful completion of this course, students would be able to			
<ul style="list-style-type: none"> <li>• Linux Basic Shell Commands and its usage.</li> <li>• Demonstrate the creation and execution of shell scripts in Linux.</li> <li>• Elaborate use the various file system and administration commands to execute shell scripts in linux.</li> <li>• Execute various programming language programs in Linux Platform</li> </ul>			
<b>Unit</b>	<b>Topics</b>	<b>No of Lectures</b>	
<b>I</b>	<p><b>Linux OS Basics:</b> History, Various Linux Distributions, Unix/Linux OS architecture, Features of Unix/Linux, Use of Linux OS.</p> <p><b>Basics of Shell:</b> Starting the shell, Shell prompt, Command structure, File Systems and Directory Structure, man pages, more documentation pages</p> <p><b>Basic Bash shell commands:</b> General purpose utility Commands, Basic commands</p> <p><b>Files Handling in Linux:</b> Type of files in Linux and its file commands, File handling commands like create file/directory, File Content Commands, Linux I/O Redirection</p> <p><b>Advanced Bash shell commands:</b> Simple Filters Commands, Filters using regular expressions.</p> <p><b>Linux environment variable:</b> Default shell environment variables, Using command aliases</p>	<b>15</b>	
<b>II</b>	<p><b>Linux File Security:</b> Types of file ownership, changing ownership of files, Different file Permission, Changing file permission, Working with types of Files Hardlink and Softlink</p>	<b>15</b>	

	<p><b>Linux Security:</b> Understanding Linux Security, Linux Administrator its role and responsibility, uses of root, sudo command, working with passwords</p> <p><b>Basic Networking Concept:</b> Networking Concepts, Working with basic networking commands ipconfig, telnet, ip, hostname, ping, netstat</p> <p><b>Working with Editors:</b> vi editor</p> <p><b>Basic Shell scripting:</b> Using multiple commands, creating script files and executing script files, displaying messages, using variables, basic arithmetic operators, Exiting the script.</p> <p><b>Using structured commands:</b> Working with if-then, if-then-else and nested if statements, test command, Compound condition testing, while command, until command, case command.</p> <p><b>Script and Process control:</b> Handling signals, Running scripts in background mode, Job scheduling commands: ps, nice, renice, at, batch</p>	
--	--	--

**Textbooks:**

- “Linux Command line and Shell Scripting Bible”, Richard Blum, Wiley India.
- “UNIX: Concepts and Applications”, Sumitabha Das, 4th Edition, McGraw Hill.
- “Official Ubuntu Book”, Matthew Helmke & Elizabeth K. Joseph with Jose Antonio Rey and Philips Ballew, 8th Ed.

**Additional References:**

- “Linux Administration: A Beginner's Guide”, Fifth Edition, Wale Soyinka, Tata McGraw-Hill, 2008.
- “Linux: Complete Reference”, Richard Petersen, 6th Edition, Tata McGraw-Hill

# **SEMESTER- II**

<b>Course Code</b>	<b>MAJOR SEM – II - Object Oriented Programming with C++</b>	<b>Credits</b>	<b>Lectures /Week</b>
<b>K23USITMJ211</b>	<b>Paper I</b>	<b>2</b>	<b>2</b>
<b>Course Outcomes:</b>			
After successful completion of this course, students would be able to			
<ul style="list-style-type: none"> <li>• To recall the basic programming principles</li> <li>• Be able to explain the difference between object oriented programming and procedural programming.</li> <li>• Be able to build C++ classes using appropriate encapsulation and design principles</li> <li>• Be able to program using more advanced C++ features such as composition of objects, operator overloads, dynamic memory allocation, inheritance and polymorphism, file I/O, exception handling, etc.</li> </ul>			
<b>Unit</b>	<b>Topics</b>	<b>No of Lectures</b>	
<b>I</b>	<p><b>INTRODUCTION OF OBJECT-ORIENTED DESIGN</b> Introduction, Objects, Class and Instance, Polymorphism, Inheritance, Object-Oriented Analysis, Finding the Objects, Object-Oriented Programming, Object-Oriented Design, Object-Oriented Analysis, Elements of Object Model, Grady Booch Approach.</p> <p><b>STARTING WITH C++:</b> C++ Overview, C++ Character Set, C++ Tokens, Variables, Counting Tokens, Data Types, Qualifiers, Range of Data Types, Your First C++ Program, Structure of a C++ Program, Styles of, Writing C++ Programs, Programming Examples</p> <p><b>FEATURES OF C++:</b> Introduction, Operators and Expressions, Declaring Constants, Type Conversion, Decision Making: An Introduction, Unconditional Branching Using Goto, Introduction to Looping</p> <p><b>OPERATORS AND REFERENCES IN C++:</b> Introduction, Scope Resolution Operator, Reference Variables, The Bool Data Type, The Operator New and Delete, Malloc Vs. New, Pointer Member Operators</p> <p><b>FUNCTION IN C++:</b> Introduction, Function, Declaration/Prototyping,,The Main Function in C++,Recursion, Call by Reference, Call by Reference Vs Call by Address, Return by Reference, Inline Function, Function Overloading, Function with Default Arguments</p>	<b>15</b>	

	<p><b>CLASS AND OBJECTS IN C++:</b> Working with Class, Structure in C++, Accessing Private Data Passing and Returning Object, Array of Object, Friend Function, Static Class Members, Constant Member Function</p> <p><b>WORKING WITH CONSTRUCTOR AND DESTRUCTOR:</b> Introduction, Constructor with Parameters, Implicit and Explicit Call to Constructor, Copy Constructor, Dynamic Initialization of Objects, Dynamic Constructor, Destructor.</p> <p><b>WORKING WITH OPERATOR OVERLOADING:</b> Introduction, Operator Overloading with Binary Operator Overloading Assignment (=) Operator, Overloading Unary Operators, Overloading Using Friend Function, Rules of Operator Overloading, Type Conversion</p>	
<p style="text-align: center;"><b>II</b></p>	<p><b>WORKING WITH INHERITANCE IN C++:</b> Introduction, Types of Inheritance, Public, Private and Protected Inheritance, Multiple Inheritance, Hierarchical Inheritance, Virtual Base Class, Constructor and Destructor in Inheritance, Containership</p> <p><b>POINTERS TO OBJECTS AND VIRTUAL FUNCTIONS:</b> Pointer to Objects, The This Pointer, What is Binding in C++?, Virtual Functions, Working of Virtual Function, Rules for Virtual Function, Pure Virtual Function and Abstract Class, Object Slicing, Some Facts about Virtual Function, Virtual Destructor</p> <p><b>INPUT-OUTPUT AND MANIPULATORS IN C++:</b> Introduction, C++ Stream Classes, Unformatted Input/Output, Formatted Input /Output Operations, Manipulators</p> <p><b>FILE HANDLING IN C++:</b> Introduction, File Streams, Opening and Closing a File, File Opening Modes Checking End of File, Random Access in File, Command Line Arguments, Working with Binary Mode Error Handling</p> <p><b>TEMPLATE PROGRAMMING:</b> Introduction, Function Template, Class Template</p> <p><b>EXCEPTION HANDLING IN C++:</b> Introduction, Basics of Exception Handling, Exception Handling Mechanism,</p>	<p style="text-align: center;"><b>15</b></p>

	Programming Examples, Exception Handling with Class Catching all Exceptions, Specifying Exception for a Function	
--	---	--

**Textbooks:**

- Object-oriented Programming C++ Simplified, Hari Mohan Pandey, University Science Press, 1st Edition 2017
- Object Oriented Programming in C++, E Balagurusamy, Tata McGraw- Hill, 5th Edition 2011
- Object-Oriented Programming in C++, Robert Lafore, Sams, 4th Edition 2002
- Programming with ANSI C++, Bhushan Trivedi, Oxford University Press, 2nd Edition

**Additional References:**

- Demystified Object- Oriented Programming with C++, Dorothy R. Kirk, Packt Publishing Lt, 1st Edition 2021
- C++ Programming: An Object-Oriented Approach, Behrouz A. Forouzan , Richard F. Gilberg, McGraw-Hill Education, 1st edition 2020
- C++ How to Program, Paul Deitel, Harvey Deitel, Pearson Education, 10th Edition 2017

<b>Course Code</b>	<b>MAJOR SEM – II - Data Structures</b>	<b>Credits</b>	<b>Lectures/ Week</b>
<b>K23USITMJ212</b>	<b>Paper II</b>	<b>2</b>	<b>2</b>
<b>Course Outcomes:</b>			
After successful completion of this course, students would be able to			
<ul style="list-style-type: none"> <li>• Describe and understand elementary data structures such as stack, queue, linked list, tree</li> <li>• Understand the various concepts for developing algorithmic solutions</li> <li>• Design, use and write programs for elementary data structures such as stack, queue, linked list, tree</li> <li>• Illustrate the use of different data structures on different types of problems.</li> </ul>			
<b>Unit</b>	<b>Topics</b>	<b>No of Lectures</b>	
<b>I</b>	<p><b>Introduction:</b> Data and Information, Data Structure, Classification of Data Structures, Primitive Data Types, Abstract Data Types</p> <p><b>Simple Data Structures:</b></p> <p><b>Array:</b> Introduction, One Dimensional Array, Memory Representation of One Dimensional Array, Multi-dimensional Arrays, Memory Representation of Two Dimensional Arrays, Advantages and Limitations of Arrays.</p> <p><b>Stack:</b> Introduction, Operations on the Stack, Memory Representation of Stack, Array Representation of Stack, Evaluation of postfix expression, Recursion.</p> <p><b>Queue:</b> Introduction, Operations on the Queue, Memory Representation of Queue, Array representation of Queue, Circular Queue.</p> <p><b>Basic Sorting Techniques</b> - Bubble, Selection, Insertion and Merge Sort and their comparative analysis.</p> <p><b>Searching Techniques</b> - Linear Search, Binary Search and their comparative analysis.</p>	<b>15</b>	
<b>II</b>	Linked List, singly Linked List, Traversal of Linked List, Searching, Memory Allocation and De-allocation, Insertion in Linked List, Deletion from Linked List, Circular Linked List, Doubly Linked List,	<b>15</b>	

	Traversal, Search, Insert and Delete in a doubly Linked List, Implementing other Data Structures.	
--	---	--

**Trees:** Tree terminologies, Binary Tree, Properties of Binary Tree, Memory Representations of Binary Tree, Binary Tree traversals, Binary Search Tree, Operations on Binary Search Tree.

**Textbooks:**

- Schaum's Outlines Data structure Seymour Lipschutz Tata McGraw Hill 2 nd 2005
- Data structure – A Pseudocode Approach with C AM Tanenbaum, Y Langsam and MJ Augustein Prentice Hall India 2 nd 2006

**Additional References:**

- A Simplified Approach to Data Structures Lalit Goyal, Vishal Goyal, Pawan Kumar SPD 1 st 2014
- An Introduction to Data Structure with Applications Jean – Paul Tremblay and Paul Sorenson Tata MacGraw Hill 2 nd 2007
- Data Structure and Algorithm Maria Rukadikar SPD 1 st 2017



Course Code	MAJOR SEM-II Practicals	Credits	Lectures / week
K23USITMJP21	IT Practical 2 (Part A+B)	2	4

### **PART-A Object Oriented Programming with C++**

Course Objectives:

- The student should be able to explain the important characteristics of the C++ programming language.
- The learner must be able to combine components of the C++ programming language to develop structured program.
- The student must demonstrate the skills essential to compile, debug, and test C++ programs correctly.

1a.	Write a C++ program to create a simple calculator.
b.	Write a C++ program to convert seconds into hours, minutes and seconds.
c.	Write a C++ program to find the volume of a square, cone, and rectangle.
2a.	Write a C++ program to find the greatest of three numbers.
b.	Write a C++ program to find the sum of even and odd n natural numbers
c.	Write a C++ program to generate all the prime numbers between 1 and n, where n is a value supplied by the user.
3a.	Write a C++ program using classes and object Student to print name of the student, roll_no. Display the same.
b.	Write a C++ program for Structure bank employee to print name of the employee, account_no. & balance. Display the same also display the balance after withdraw and deposit
c.	Write a Program to find Maximum out of Two Numbers using friend function. Note: Here one number is a member of one class and the other number is member of some other class.
d.	Write a C++ Program using copy constructor to copy data of an object to another object.
4a.	Write a C++ program to overload new/delete operators in a class.
b.	Write a C++ program to access members of a STUDENT class using pointer to object members

c.	Write a C++ Program that illustrates the different types of inheritance.
5a.	Write a C++ Program to design a student class representing student roll no. and a test class (derived class of student) representing the scores of the student in various subjects and sports class representing the score in sports. The sports and test class should be inherited by a result class having the functionality to add the scores and display the final result for a student.
6a.	Write a C++ program to maintain the records of person with details (Name and Age) and find the eldest among them. The program must use this pointer to return the result.
7a.	Write a C++ program illustrating the use of virtual functions in class.
b.	Write a C++ program to design a class representing the information regarding digital library (books, tape: book & tape should be separate classes having the base class as media). The class should have the functionality for adding new item, issuing, deposit etc. the program should use the runtime polymorphism.
8a.	Write a C++ program to show conversion from string to int and vice-versa.
b.	Write a C++ program to copy the contents of one file to another.
c.	Write a C++ program to implement I/O operations on characters. I/O operations includes inputting a string, Calculating length of the string, Storing the String in a file, fetching the stored characters from it, etc.
d.	Write a C++ program to perform read/write binary I/O operation on a file (i.e. write the object of a structure/class to file).
9a.	Write a C++ program to implement the exception handling with multiple catch statements.
10a.	Write a C++ Program to create Simple calculator using Class template.
b.	Write a C++ Program to get maximum of two number using Class template.

## Part B- Data Structures Practicals

### Course Outcomes:

After successful completion of this course, students would be able to

- List the errors and warnings for the given input.
- Explain and demonstrate the execution process of the programs.
- Solve the problems based on each algorithm.
- Write modularized program code for implementing various operating system algorithms and integrate them.

### Data structures – Practical

<b>1</b>	Writing simple programs using one dimensional and two dimensional array
<b>2</b>	Write a program to implement a stack
<b>3</b>	Write a program to evaluate a postfix expression
<b>4</b>	Write a program to implement a linear queue
<b>5</b>	Write a program to implement a circular queue
<b>6</b>	Write a program to perform linear search and binary search on the given elements.
<b>7</b>	Write a program to implement different sorting techniques such as insertion, bubble, merge and selection sort
<b>8</b>	Write a program to implement a singly linked list
<b>9</b>	Write a program to implement a doubly linked list
<b>10</b>	Write a program to create and traverse a binary search tree

**Practicals will be completed within 60 hours.**

<b>Course Code</b>	<b>MINOR SEM – II - Discrete Mathematics</b>	<b>Credits</b>	<b>Lectures /Week</b>
<b>K23USITMR221</b>	<b>Paper I</b>	<b>2</b>	<b>2</b>
<p><b>Course Outcomes:</b>            After successful completion of this course, students would be able to</p> <ul style="list-style-type: none"> <li>• Describe the basic concepts of sets, permutations, relations, graphs, trees</li> <li>• Understand sets and perform operations and algebra on set</li> <li>• Apply the concepts of graphs to solve various problems in day to day life.</li> <li>• Illustrate relationships using directed, undirected, weighted graphs and trees</li> </ul>			
<b>Unit</b>	<b>Topics</b>	<b>No of Lectures</b>	
<b>I</b>	<p>Set Theory Introduction:- Sets and Elements, Subsets, Venn Diagrams, Set Operations, Algebra of Sets, Duality, Finite Sets, Counting Principle, Classes of Sets, Power Sets, Partitions, Mathematical Induction</p> <p>Relations:- Introduction, Product Sets, Relations, Pictorial Representatives of Relations, Composition of Relations, Types of Relations, Closure Properties, Equivalence Relations, Partial Ordering Relations.</p> <p>Functions and Algorithms Introduction, Functions, One-to-One, Onto, and Invertible Functions, Mathematical Functions, Exponential and Logarithmic Functions, Sequences, Indexed Classes of Sets, Recursively Defined Functions, Cardinality, Algorithms and Functions, Complexity of Algorithms</p>	<b>15</b>	
<b>II</b>	<p>Counting Principle Cardinality of sets, Basics of Counting: Addition rule, Product rule, Inclusion and Exclusion Principle, Mathematical Induction: 1st and 2nd principle of induction</p> <p>Introduction to Graphs and Operations on Graphs Definition and examples of graph, Handshaking lemma and its corollaries. Types of graph, Complete graph, bipartite graph, Regular graph, Null graph. Isomorphism of graphs, Adjacency and Incidence Matrix of a Graph. Vertex induced subgraph, Edge induced subgraph, Vertex deleted subgraph, Edge deleted subgraph, Union of two graphs, Intersection of two graphs, Product of two graphs, Ring Sum of two graphs, Fusion of vertices, Complement of a graph.</p>	<b>15</b>	

<p><b>Textbooks:</b></p> <ul style="list-style-type: none"><li>• Kenneth H.Rosen. Discrete Mathematics and its applications. (7th edition) McGraw-Hill Higher Education, 2017.</li><li>• Bernard Kolman, Robert C.Busby , and Sharon cutler Ross. Discrete Mathematical Structures (6th edition). Prentice-Hall, Inc. Upper Saddle River, NJ, USA, 2003.</li></ul> <p><b>Additional References:</b></p> <ul style="list-style-type: none"><li>• John Clark and Derek Holton, a first look at Graph Theory, 2013</li></ul>		

<b>Course Code</b>	<b>OPEN ELECTIVE SEM – II – Introduction to Data Science</b>	<b>Credits</b>	<b>Lectures /Week</b>
<b>K23USITOE231</b>	<b>Paper I</b>	<b>2</b>	<b>2</b>
<b>Course Outcomes:</b>			
After successful completion of this course, students would be able to			
<ul style="list-style-type: none"> <li>• Learn the tools and techniques of data science.</li> <li>• Understand how to use the methods for handling data.</li> <li>• Apply principles of Data Science to study business problems.</li> <li>• Adapt easily to the changes and new demands from industry.</li> </ul>			
<b>Unit</b>			
<b>Unit</b>	<b>Topics</b>	<b>No of Lectures</b>	
<b>I</b>	Introduction to Data Science , Evolution of Data Science, What is Data? Introduction to database Systems, data mining and data warehousing, Kinds of data: e.g. static, spatial, temporal, text, media. Understanding Exploratory Data Analysis.	<b>15</b>	
<b>II</b>	Data Curation, Management and Organization : Introduction to Statistical Models , Simple Linear Regression , Multiple Linear Regression , Logistic Regression , Review of hypothesis testing, confidence intervals, etc. ,Estimation e.g. likelihood principle, Bayes	<b>15</b>	
<b>Textbooks:</b>			
<ul style="list-style-type: none"> <li>• “Doing Data Science”, Cathy O’Neil and Rachel Schutt, O’Reilly, 2015</li> </ul>			
<b>Additional References:</b>			
<ul style="list-style-type: none"> <li>• An Introduction to Statistical Learning with Applications in R Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani: Springer US 2 and 2021</li> </ul>			

<b>Course Code</b>	<b>OPEN ELECTIVE SEM – II - E-Commerce</b>	<b>Credits</b>	<b>Lectures /Week</b>
<b>K23USITOE232</b>	<b>Paper II</b>	<b>2</b>	<b>2</b>
<b>Course Outcomes:</b>			
After successful completion of this course, students would be able to			
<ul style="list-style-type: none"> <li>● To know how the business is carried out through electronic media</li> <li>● To understand the types and ways of doing business over internet</li> <li>● To apply the knowledge after becoming a professional or an entrepreneur</li> <li>● To analyze the security concerns while transacting using electronic media</li> </ul>			
<b>Unit</b>			
<b>Unit</b>	<b>Topics</b>	<b>No of Lectures</b>	
<b>I</b>	Introduction to E-Business and E-Commerce:- Define the e-Commerce and e-Business, Define e-Commerce Types of EC transactions. Define e-Business Models. Internet Marketing and e-Tailing. Elements of e-Business Models. Explain the benefits and limitations of e-Commerce.	<b>15</b>	
<b>II</b>	E-Business applications, E-Procurement and E- Payment Systems:- Integration and e-Business suits. ERP, e-SCM, CRM, E-Payment. E-Procurement definition, processes, methods and benefits. Discuss the categories and users of smart cards. Describe payment methods in B2B EC	<b>15</b>	
<b>Textbooks:</b>			
<ul style="list-style-type: none"> <li>● Electronic Commerce: A Managerial Perspective, Turban, E. et al., Prentice Hall-2008.</li> <li>● Frontiers of e-commerce, Ravi Kalakota, Pearson.</li> </ul>			
<b>Additional References:</b>			
<ul style="list-style-type: none"> <li>● Electronic Business and Electronic Commerce Management, 2nd edition, Dave Chaffey, Prentice Hall, 2006</li> <li>● . e-Learning Tools and Technologies, Horton and Horton, Wiley Publishing</li> </ul>			

Course Code	VOCATIONAL SKILL COURSE SEM – II - Course Title	Credits	Lectures /Week
K23USITVC241	Paper I- Advanced Web Programming	2	2
<p><b>Course Outcomes:</b></p> <p>After successful completion of this course, students would be able to</p> <ul style="list-style-type: none"> <li>• Describe the fundamental concepts of .NET web programming, including ASP.NET, C# and .NET framework</li> <li>• Summarize the concepts of server-side programming with C# and how it interacts with the web interface.</li> <li>• Develop web applications using ASP.NET, C# and the .NET framework.</li> <li>• Evaluate the performance and scalability of a .NET web application.</li> </ul>			
Unit	Topics	No of Lectures	
I	<p><b>Introducing .NET:</b> Introduction to .NET, .NET Framework and its architecture, Overview to C#, The Common Language Runtime, The .NET Class Library.</p> <p><b>The C# Language:</b> C# Language Basics, Variables and Data Types, Object-Based Manipulation, Conditional Logic, Loops, Methods.</p> <p><b>Types, Objects, and Namespaces:</b> The Basics About Classes, Building a Basic Class, Value Types and Reference Types, Understanding Namespaces and Assemblies, Advanced Class Programming.</p> <p><b>Web Form Fundamentals:</b> Writing Code, Adding Event Handlers, Understanding the Anatomy of an ASP.NET Application, Introducing Server Controls, Using the Page Class, Using Application Events, Configuring an ASP.NET Application.</p> <p><b>Form Controls:</b> Stepping Up to Web Controls, Web Control Classes, List Controls, AutoPostBack, Validation Controls, Rich Controls, The Calendar, The AdRotator, Multiview Control, User Controls and Graphics, Website Navigation: Site Maps, The SiteMapPath Control, The TreeView Control, The Menu Control.</p>	15	



	<p><b>Error Handling, Logging, and Tracing</b> : Avoiding Common Errors, Understanding Exception Handling, Handling Exceptions, Throwing Your Own Exceptions, Using Page Tracing</p>	
<p style="text-align: center;"><b>II</b></p>	<p><b>State Management</b> : Understanding the Problem of State, Using View State, Transferring Information Between Pages, Using Cookies, Managing Session State, Configuring Session State, Using Application State</p> <p><b>Styles, Themes, and Master Pages</b> : Styles, Themes, Master Page Basics, Advanced Master Pages</p> <p><b>ADO.NET Fundamentals:</b> Understanding Databases, Configuring Your Database, Understanding SQL Basics, Understanding the Data Provider Model, Using Direct Data Access, Using Disconnected Data Access.</p> <p><b>Data Binding:</b> Introducing Data Binding, Using Single-Value Data Binding, Using Repeated-Value Data Binding, Working with Data Source Controls</p> <p><b>The Data Controls:</b> The GridView, Formatting the GridView, Selecting a GridView Row, Editing with the GridView, Sorting and Paging the GridView, Using GridView Templates, The DetailsView and FormView</p> <p><b>XML:</b> XML Basics, The XML Classes, XML Validation.</p> <p><b>Security Fundamentals:</b> Understanding Security Requirements, Authentication and Authorization, Forms Authentication, Windows Authentication.</p> <p><b>ASP.NET AJAX:</b> Understanding Ajax, Using Partial Refreshes, Using Progress Notification, Working with the ASP.NET AJAX Control Toolkit.</p>	<p style="text-align: center;"><b>15</b></p>
<p><b>Textbooks:</b></p> <ul style="list-style-type: none"> <li>• Beginning ASP.NET 4.5 in C#, Matthew MacDonald Apress 2012,</li> <li>• C# 2015 Anne Bohem and Joel Murach, Murach Third Edition, 2016</li> <li>• Murach’s ASP.NET 4.6 Web Programming in C#2015, Mary Delamater and Anne Bohem, SPD Sixth Edition, 2016</li> </ul>		

**Additional References:**

- ASP.NET 4.0 programming, J. Kanjilal, Tata McGraw-Hill, 2011
- Programming ASP.NET, D.Esposito, Microsoft Press (Dreamtech), 2011
- Beginning Visual C# 2010, K. Watson, C. Nagel, J.H Padderson, J.D., Reid, M.Skinner, Wrox (Wiley), 2010

<b>Course Code</b>	<b>SKILL ENHANCEMENT COURSE SEM – II - Python Programming</b>	<b>Credits</b>	<b>Lectures /Week</b>
<b>K23USITSC251</b>	<b>Paper I</b>	<b>2</b>	<b>2</b>
<b>Course Outcomes:</b>			
After successful completion of this course, students would be able to			
<ul style="list-style-type: none"> <li>• Learn python programming language basics and define its concept</li> <li>• Understand the data types and represent it in various ways</li> <li>• Demonstrate different programs using python language</li> <li>• Integrate MySQL database to python programming language</li> </ul>			
<b>Unit</b>	<b>Topics</b>	<b>No of Lectures</b>	
<b>I</b>	<p><b>Introduction:</b> The Python Programming Language, History, features, Installing Python, Running Python program, Debugging : Syntax Errors, Runtime Errors, Semantic Errors, Experimental Debugging, Formal and Natural Languages, The Difference Between Brackets, Braces, and Parentheses, Variables and Expressions Values and Types, Variables, Variable Names and Keywords, Type conversion, Operators and Operands, Expressions, Interactive Mode and Script Mode, Order of Operations.</p> <p>Conditional Statements: if, if-else, nested if –else</p> <p>Looping: for, while, nested loops Control statements: Terminating loops, skipping specific conditions</p> <p><b>Functions:</b> Function Calls, Type Conversion Functions, Math Functions, Composition, Adding New Functions, Definitions and Uses, Flow of Execution, Parameters and Arguments, Variables and Parameters Are Local, Stack Diagrams, Fruitful Functions and Void Functions, Why Functions? Importing with from, Return Values, Incremental Development, Composition, Boolean Functions, More Recursion, Leap of Faith, Checking Types</p> <p><b>Strings:</b> A String Is a Sequence, Traversal with a for Loop, String Slices, Strings Are Immutable, Searching, Looping and Counting, String Methods, The in Operator, String Comparison, String Operations.</p> <p><b>Lists:</b> Values and Accessing Elements, Lists are mutable, traversing a List, Deleting elements from List, Built-in List Operators, Concatenation, Repetition, In Operator, Built-in List functions and methods</p> <p><b>Tuples and Dictionaries:</b> Tuples, Accessing values in Tuples,</p>	<b>15</b>	

	<p>Tuple Assignment, Tuples as return values, Variable-length argument tuples, Basic tuples operations, Concatenation, Repetition, in Operator, Iteration, Built-in Tuple Functions          Creating a Dictionary, Accessing Values in a dictionary, Updating Dictionary, Deleting Elements from Dictionary, Properties of Dictionary keys, Operations in Dictionary, Built-In Dictionary Functions, Built-in Dictionary Methods</p>	
<p style="text-align: center;"><b>II</b></p>	<p><b>Files:</b> Text Files, The File Object Attributes, Directories  <b>Exceptions:</b> Built-in Exceptions, Handling Exceptions, Exception with Arguments, User-defined Exceptions  <b>Regular Expressions</b> – Concept of regular expression, various types of regular expressions, using match function.  <b>Classes and Objects:</b> Overview of OOP (Object Oriented Programming), Class Definition, Creating Objects, Instances as Arguments, Instances as return values, Built-in Class Attributes, Inheritance, Method Overriding, Data Encapsulation, Data Hiding  <b>Multithreaded Programming:</b> Thread Module, creating a thread, synchronizing threads, multithreaded priority queue  <b>Modules:</b> Importing module, Creating and exploring modules, Math module, Random module, Time module</p> <p><b>Creating the GUI Form and Adding Widgets:</b>  <b>Widgets:</b> Button, Canvas, Checkbutton, Entry, Frame, Label, Listbox, Menubutton, Menu, Message, Radiobutton, Scale, Scrollbar, text, Toplevel, Spinbox, PanedWindow, LabelFrame, tkinter.Messagebox. Handling Standard attributes and Properties of Widgets. <b>Layout Management:</b> Designing GUI applications with proper Layout Management features.  <b>Look and Feel Customization:</b> Enhancing Look and Feel of GUI using different appearances of widgets.  <b>Storing Data in Our MySQL Database via Our GUI:</b>          Connecting to a MySQL database from Python, Configuring the MySQL connection, Designing the Python GUI database, Using the INSERT command, Using the UPDATE command, Using the DELETE command, Storing and retrieving data from MySQL database.</p>	<p style="text-align: center;"><b>15</b></p>
<p><b>Textbooks:</b></p> <ul style="list-style-type: none"> <li>● Python Basics: A Practical Introduction to Python</li> <li>● Programming with Python</li> <li>● Think python 2nd Edition</li> </ul> <p><b>Additional References:</b></p> <ul style="list-style-type: none"> <li>● A Python Book: Beginning Python, Advanced Python, and Python</li> </ul>		

## **Evaluation Scheme for First Year (UG) under NEP (2 credits)**

### **I. Internal Evaluation for Theory Courses – 20 Marks**

1) Continuous Internal Assessment(CIA) Assignment - Tutorial/ Case Study/ Project / Presentations/ Group Discussion / Ind. Visit. – 10 marks

2) Continuous Internal Assessment(CIA) ONLINE Unit Test – 10 marks

### **II. External Examination for Theory Courses – 30 Marks**

**Duration: 1 Hours**

**Theory question paper pattern: All questions are compulsory.**

<b><u>Question</u></b>	<b><u>Based on</u></b>	<b><u>Marks</u></b>
<b><u>Q.1</u></b>	<b><u>Unit I</u></b>	<b><u>15</u></b>
<b><u>Q.2</u></b>	<b><u>Unit II</u></b>	<b><u>15</u></b>

- All questions shall be compulsory with internal choice within the questions.
- Each Question may be subdivided into sub questions as a, b, c, d, etc. & the allocation of Marks depends on the weightage of the topic.

### **III. Practical Examination**

- Each core subject carries 50 Marks.
- Duration: 2 Hours for each practical course.
- Minimum 80% practical from each core subjects are required to be completed.
- Certified Journal is compulsory for appearing at the time of Practical Exam

**NOTE: To pass the examination, attendance is compulsory in both Internal & External (Theory + Practical) Examinations.**