

AC 01.09.23 ITEM NO: 1.1

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: Computer Science

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

PROGRAM OUTCOMES

PO	Description
A student completing Bachelor's Degree in Science 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 CS professional careers and/or to continue their education in CS and/or related postgraduate programmes.
PO3	To be capable of managing complex CS 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-Computer Science**

Semester	Course Code	Course Title	Vertical	Credit
I	K23USCSMJ111	Fundamentals of Algorithms	Major	2
	K23USCSMJ112	Introduction to OOPs using C++	Major	2
	K23USCSMJ11	CS practical 1 (Part A+B)	Major	2
	K23USCSOE131	Basics of AI & ML	OE	2
	K23USCSOE132	Cyber Law	OE	2
	K23USCSVC141	C Programming	VSC	2
	K23USCSSC151	Python	SEC	2
II	K23USCSMJ211	Principles of Operating Systems	Major	2
	K23USCSMJ212	DataBase Systems	Major	2
	K23USCSMJ21	CS practical 2 (Part A+B)	Major	2
	K23USCSMR231	Discrete Mathematics	Minor	2
	K23USCSOE231	Introduction to Data Science	OE	2
	K23USCSOE232	E-Commerce	OE	2
	K23USCSVC241	Linux	VSC	2
	K23USCSSC251	Introduction to SQL	SEC	2

SEMESTER-I

Course Code	MAJOR SEM – I - Fundamentals of Algorithms	Credits	Lectures /Week
K23USCSMJ111	Paper I	2	2
Course Outcomes:			
After successful completion of this course, students would be able to			
<ul style="list-style-type: none"> • Describe different concepts related to algorithms. • Discuss the classification and designing techniques of algorithms on various criteria. • Apply sorting, searching and selection algorithms on the given data. • Apply algorithm design techniques on the given data. • Compare and analyze the performance of different algorithms on different parameters. 			
Unit	Topics	No of Lectures	
I	<p>Introduction to algorithms - What is algorithm, Characteristics of an algorithm, Flowcharts, Running time analysis of algorithm, How to Compare Algorithms, Rate of Growth, Asymptotic Notations: Big-O Notation, Omega-Ω Notation, Theta-Θ Notation, , Estimating running time / number of steps of executions on paper</p> <p>Introduction to Data Structures - What is data structure, its types, Introduction to Array(1-D and 2-D) and Stack, operations on these data structures, advantages and disadvantages</p> <p>Applications: Evaluation of Postfix expression, Recursion</p> <p>Basic Sorting Techniques - Bubble, Selection and Insertion Sort and their comparative analysis</p> <p>Searching Techniques - Linear Search, Binary Search and their comparative analysis</p> <p>Selection Techniques - Selection by Sorting, Finding the Kth Smallest / largest elements with different methods and their comparative analysis</p>	15	
II	<p>Classification of Algorithms: Introduction to various types of classifications / design criteria and design techniques</p> <p>Algorithm Design Techniques -</p> <p>Greedy Technique - Concept, Advantages & Disadvantages, Applications, Implementation using problems like - file merging problem</p> <p>Divide-n-Conquer - Concept, Advantages & Disadvantages, Applications, Implementation using problems like - merge sort, Strassen's Matrix Multiplication</p>	15	

	<p>Algorithm Design Techniques</p> <p>Dynamic Programming - Concept, Advantages & Disadvantages, Applications, Implementation using problems like - Fibonacci series, Factorial of a number, Longest Common subsequence</p> <p>Backtracking Programming - Concept, Advantages & Disadvantages, Applications, Implementation using problems like N-Queen Problem</p>	
<p>Textbooks:</p> <ul style="list-style-type: none"> ● Narasimha Karumanchi, Data Structures and Algorithms made easy, CareerMonk Publications, 2016 ● Narasimha Karumanchi, Data Structures and Algorithmic thinking using Python, CareerMonk Publications, 2016 <p>Additional References:</p> <ul style="list-style-type: none"> ● Cormen, Leiserson, Rivest, Stein, Introduction to Algorithms, The MIT Press, 2nd edition, 2001 ● Anany Levitin, Introduction to the Design and Analysis of Algorithms, Pearson, 3rd Edition, 2011 ● S. Sridhar, Design and Analysis of Algorithms, Oxford University Press, 2014 		

Course Code	MAJOR SEM – I - Introduction to OOPs using C++	Credits	Lectures /Week
K23USCSMJ112	Paper II	2	2
Course Outcomes:			
After successful completion of this course, students would be able to			
<ul style="list-style-type: none"> • Describe various data types and OOPs concepts. • Understand the importance of OOPs approach over procedural language. • Apply the concepts of OOPS like encapsulation, inheritance and polymorphism. • Illustrate basic file operations. 			
Unit	Topics	No of Lectures	
I	<p>Introduction to Programming Concepts: Object oriented programming paradigm, basic concepts of object oriented programming, benefits of object oriented programming, object oriented languages, applications of object oriented programming.</p> <p>Tokens-keywords, identifiers, constants-integer, real, character and string constants, backslash constants, features of C++ and its basic structure, simple C++ program without class, compiling and running C++ program.</p> <p>Data Types, Data Input Output and Operators: Basic data types, variables, rules for naming variables, programming constants, the type cast operator, implicit and explicit type casting, cout and cin statements, operators, precedence of operators.</p> <p>Decision Making, Loops, Arrays and Strings: Conditional statements-if, if...else, switch loops- while, do...while, for, types of arrays and string and string manipulations</p> <p>Classes, Abstraction & Encapsulation: Classes and objects, Dot Operator, data members, member functions, passing data to functions, scope and visibility of variables in function.</p> <p>Constructors and Destructors: Default constructor, parameterized constructor, copy constructor, private constructor, destructors.</p> <p>Working with objects: Accessor - mutator methods, static data and static function, access specifiers, array of objects.</p>	15	
II	<p>Polymorphism - Binding-static binding & overloading, constructor overloading function overloading, operator overloading, overloading unary and binary operators.</p>	15	

	<p>Modeling Relationships in Class Diagrams: Association, Aggregation- Composition and examples covering these principles</p> <p>Inheritance: Defining base class and its derived class, access specifiers, types of inheritance-single, multiple, hierarchical, multilevel, hybrid inheritance, friend function and friend class, constructors in derived classes.</p> <p>Run time Polymorphism - Dynamic Binding, Function overriding, virtual function, pure virtual function, virtual base class, abstract class.</p> <p>Pointers: Introduction to pointers, * and & operators, assigning addresses to pointer variables, accessing values using pointers, pointers to objects & this pointer, pointers to derived classes</p> <p>File Handling: File Stream classes, opening and closing file-file opening modes, text file handling, binary file handling.</p>	
<p>Textbooks:</p> <ul style="list-style-type: none"> ● Object Oriented Programming with C++, Balagurusamy E., 8th Edition, McGraw Hill Education India. ● UML & C++: A Practical Guide to Object Oriented Development, Lee/Tepfenhart, Pearson Education, 2nd Edition 2015 <p>Additional References:</p> <ul style="list-style-type: none"> ● Mastering C++ by Venugopal, Publisher: McGraw-Hill Education, 2017 ● Let Us C++ by Kanetkar Yashwant, Publisher: BPB Publications, 2020 ● Object Oriented Analysis and Design by Timothy Budd TMH, 2001 		

Course Code	Fundamentals of Algorithms – Practical	Credits	Lectures/Week
K23USCSMJ11		2	4

Part A

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.

1	Develop algorithm for simple problems.
2	Draw flowchart for simple problems.
3	Computing step count for different algorithms / code segments.
4	Programs to sort the given elements by using various algorithms like bubble, selection sort and insertion sort.
5	Program to perform linear search and binary search on the given elements.
6	Program to obtain the N th Max/Min element in the given data by using various algorithms.
7	Program to implement merge sort using Divide and Conquer method
8	Programs on recursion like factorial and tower of Hanoi.
9	Program to implement the Fibonacci series using dynamic programming and to compare it with the general recursive algorithm.
10	Write a program to evaluate a postfix expression.

PART B

Introduction to OOPs using C++ - Practical

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 OO concepts.
- Write modularized program code for implementing OO concepts.

1	Program to demonstrate use of data members & member functions.
2	Programs based on branching and looping statements using classes.

3	Program to demonstrate one and two dimensional arrays using classes.
4	Programs to demonstrate various types of constructors and destructors.
5	Programs to demonstrate use of public, protected & private scope specifiers.
6	Programs to demonstrate single and multilevel inheritance.
7	Programs to demonstrate friend function, inline function, this pointer
8	Programs to demonstrate function overloading and overriding.
9	Programs to demonstrate use of pointers.
10	Programs to demonstrate text and binary file handling.

Course Code	OPEN ELECTIVE SEM – I – Basics of AI and ML	Credits	Lectures /Week
K23USCSOE131	Paper I	2	2
Course Outcomes:			
After successful completion of this course, students would be able to			
<ul style="list-style-type: none"> ● Give the risk and benefits of AI ● Explain the History of Artificial Intelligence and the future of AI ● Apply Machine Learning concept to solve problems ● Compare and contrast the Machine Learning task 			
Unit			
Unit	Topics	No of Lectures	
I	What Is AI? The Foundations of Artificial Intelligence, The History of Artificial Intelligence, Risks and Benefits of AI, Future of AI	15	
II	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.	15	
Textbooks:			
<ul style="list-style-type: none"> ● Artificial Intelligence: A Modern Approach, 4th US ed., Stuart Russell and Peter Norvig ● Artificial Intelligence Elaine Rich, Kevin Knight, Tata McGraw Hill 3rd 2017 <p>Additional References:</p> <ul style="list-style-type: none"> ● Machine Learning: The Art and Science of Algorithms that Make Sense of Data ● Peter Flach Cambridge University Press 1st 2012 ● Introduction to Machine Learning, Ethem Alpaydin PHI 2nd 2013 			

Course Code	OPEN ELECTIVE SEM – I – Cyber Law	Credits	Lectures /Week
K23USCSOE132	Paper II	2	2
Course Outcomes:			
After successful completion of this course, students would be able to			
<ul style="list-style-type: none"> • Make Learner Conversant With The Social And Intellectual Property Issues Emerging From 'Cyberspace. • Explore The Legal And Policy Developments In Various Countries To Regulate Cyberspace; • Develop The Understanding Of Relationship Between Commerce And Cyberspace • Give Learners In Depth Knowledge Of Information Technology Act And Legal Frame Work Of Right To Privacy, Data Security And Data Protection. 			
Unit	Topics	No of Lectures	
I	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.	15	
II	Overview of General Laws and Procedures in India, Penalties & Offences under the IT Act, 2000	15	
Textbooks:			
<ul style="list-style-type: none"> • Cyber law –The Indian perspective by Pavan Duggal 			
Additional References:			
<ul style="list-style-type: none"> • CYBER FRAUDS, CYBERCRIMES & LAW IN INDIA by Pavan Duggal 			

Course Code	VOCATIONAL SKILL COURSE SEM – I - C programming (Practical Based)	Credits	Lectures/W eek
K23USCSVC141	Paper I	2	2
Course Outcomes:			
After successful completion of this course, students would be able to			
<ul style="list-style-type: none"> • Define algorithms and flowcharts for given problems in C programming. • Illustrate the use of simple data types, operators, and control structures in C programming. • Implement various standard library functions in C programming • Divide the programs into separate modules by writing user defined functions. 			
Unit	Topics	No of Sessions	
I	<p>Structure of C: Header and body, Use of comments, Compilation of a program. Formatted I/O: printf(), scanf().</p> <p>Data Concepts & Qualifiers: Variables, Constants, data types like: int, float char, double and void. short and long size qualifiers, signed and unsigned qualifiers.</p> <p>Variables: Declaring variables, Scope of the variables according to block, Hierarchy of data types.</p> <p>Types of operators: Arithmetic, Relational, Logical, Compound Assignment, Increment, and decrement, Conditional or ternary, Bitwise and Comma operators. Precedence and order of evaluation, Statements and Expressions. Automatic and Explicit type conversion.</p> <p>Iterations: Control statements for decision making: (i) Branching: if statement, else.. if statement, switch statement. (ii) Looping: while loop, do.. while, for loop. (iii) Jump statements: break, continue and goto</p> <p>Arrays: (One and multidimensional), declaring array variables, initialization of arrays, accessing array elements.</p>	15	
II	<p>Data Input and Output functions: Character I/O format: getch(), getche(), getchar(), getc(), gets(), putchar(), putc(), puts().</p> <p>Strings: Declaring and initializing String variables, Character and string handling functions.</p> <p>Functions: Global and local variables, Function definition, return statement, Calling a function by value.</p>	15	

	<p>Recursion: Definition, Recursion functions.</p>	
	<p>Pointer: Fundamentals, Pointer variables, Referencing and de-referencing, Pointer Arithmetic, Using Pointers with Arrays, Using Pointers with Strings, Array of Pointers, Pointers as function arguments, Functions returning pointers.</p>	
<p>Textbooks:</p> <ul style="list-style-type: none">• Programming in ANSI C (Third Edition) : E Balagurusamy, TMH <p>Additional References:</p> <ul style="list-style-type: none">• Programming with C (Third Edition): Byron S Gottfried (Adapted by Jitender Kumar Chhabra) Schaum's Outlines (TMH)• Let us C by Yashwant Kanetkar, BPB		

Course Code	SKILL ENHANCEMENT COURSE SEM – I – Python	Credits	Lectures/ Week
K23USCSC151	Paper I	2	2
<p>Course Outcomes: After successful completion of this course, students would be able to</p> <ul style="list-style-type: none"> • To learn basic syntax of Python programming language. • Describe To define the structure and components of a Python program. • Understand data storing and processing mechanism using Arrays, String, List, Tuple and Dictionary • Understand sequential control how to write loops and decision statements in Python • Demonstrate inbuilt input/output operations and compound data types in Python • Develop python simple standalone application 			
Unit	Topics	No of Lectures	
I	<p>Introduction to Python Language</p> <p>What is Python, Uses of Python Programming Language / Python Applications, Python for Software development, Python for Networking, Python for Automated Testing, Features of Python Programming Language, Implementations of Python, and Python career opportunities.</p> <p>Download & Install Python</p> <p>Download your operating system compatible Python Interpreter, install Python, set environment variable, customize Python shell, write & execute Python programs using Interactive mode and script mode. Python PyCharm or IDE, set Python for PyCharm IDE, configure PyCharm IDE, write & execute Python programs.</p> <p>Python Language Syntax</p> <p>Modes of Programming in Python, Interactive mode programming, Script mode programming, Creating Python program file, Python Identifiers, Python keywords, Lines and Indentation, Writing code blocks, Comments in Python, and Quotation in Python.</p>	15	

	<p>Python Keywords and Identifiers Python keywords or Reserved words, Python keywords define the syntax and structure of the Python language, Python keywords are case sensitive, Python literals (True, False, Null), Python Identifiers, class names, variable names, function names, method names, and Identifier naming rules.</p> <p>Python Variables What is Variable?, Declaration of Variables, Assign Values to Variables, Initialization, Reading, Variable naming restrictions, and Types of Python Variables.</p> <p>Python Data Types What is Data Type?, Implicit Declaration of Data Types, Python Numbers (Integers, floating-point numbers, and complex numbers), Python Strings, Python boolean data type.</p> <p>Python Operators Arithmetic operators, Assignment operators, Unary minus operator, Relational operators, Logical operators, Bitwise operators, Membership operators, Identity operators, Precedence of Operators, Basic Input and Output Functions.</p>	
<p>II</p>	<p>Control Statements: The if statement, The if ... else Statement, The 'if ... elif ... else' Statement, Loop Statement- while loop, for loop, Infinite loop, Nested loop, The else suite, break statement, continue statement, pass statement, return statement</p> <p>Python Control Flow – Decision Making Decision Making / Conditional Statements in Python, Simple If Structure, if-else structure, if elif structure, and nested If Structure.</p> <p>Python Control Flow – Looping Python Control Flow Statements, Python Loop Statements. Python while loop, Python for loop, Python range(), Python Nested Loop Structures, and Inserting conditions in Loops and vice versa.</p> <p>Python Control Flow – Branching Python Flow Control – Branching Statements, break statement, continue statement, pass statement, return statement. Arrays: Creating Arrays, Indexing and Slicing of Arrays, Basic Array Operations, Arrays Processing, Mathematical Operations on Array, Aliasing Arrays, Slicing and Indexing in Arrays.</p>	<p>15</p>

	<p>Strings: Creating Strings, Functions of Strings, Working with Strings, Length of a String, Indexing and Slicing, Repeating and Concatenating Strings,</p> <p>List and Tuples: Lists, List Functions and Methods, List Operations, List Slices, Nested Lists, Tuples, Functions in Tuple</p> <p>Dictionaries: Creating a Dictionary, Dictionary Methods, Using for Loop with Dictionaries, Operations on Dictionaries, Sorting the Elements of a Dictionary.</p> <p>Functions: Function definition and call, Returning Results, Returning Multiple Values from a Function, Built-in Functions, User defined functions, call by reference, call by value, Parameters and Arguments, Formal and Actual Arguments.</p>	
<p>Textbooks:</p> <ul style="list-style-type: none"> ● Yashwant Kanetkar, Aditya Kanetker, Let Us Python: A Programmer-Friendly Guide- 5th Edition <p>Additional References:</p> <ul style="list-style-type: none"> ● Python: The Complete Reference, Martin C. Brown, McGraw Hill, 2018 ● Beginning Python: From Novice to Professional, Magnus Lie Hetland, Apress, 2017 ● Programming in Python 3, Mark Summerfield, Pearson Education, 2nd Ed, 2018 ● Python Programming: Using Problem Solving Approach, Reema Thareja, Oxford University Press, 2017 		

SEMESTER-II

Course Code	MAJOR SEM – II - Principles of Operating Systems	Credits	Lectures /Week
K23USCSMJ211	Paper I	2	2
Course Outcomes:			
After successful completion of this course, students would be able to			
<ul style="list-style-type: none"> • Describe different concepts of the operating system. • Discuss the concept of file systems and mass storage structure. • Apply process scheduling algorithms on a given scenario. • Apply deadlock handling techniques to determine existence of deadlock and recover it. • Compare and analyze the performance of different algorithms. 			
Unit			
Unit	Topics	No of Lectures	
I	<p>Structures: Definition of Operating System, Functions of Operating System, Operating-System Services, System Calls, Types of System Calls, Operating-System Structure.</p> <p>Process Concept: Process concept, Process State, PCB, Process scheduling, Operations on processes.</p> <p>Process Synchronization: Inter-process Communication, Critical Section Problem, Semaphores: Usage, Implementation.</p> <p>Process Scheduling: Basic Concept - CPU-I/O burst cycle, CPU scheduler, Preemptive, Scheduling, Dispatcher, Scheduling Criteria, Scheduling Algorithms- FCFS, SJF, Priority scheduling, Round-robin scheduling, Multiple queue scheduling, Multilevel feedback queue scheduling.</p> <p>Deadlocks: System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock</p>	15	
II	<p>Main Memory: Background, Logical address space, Physical address space, MMU, Swapping, Contiguous Memory Allocation, Segmentation, Paging, Structure of the Page Table.</p> <p>Virtual Memory: Background, Demand Paging, Copy-on-Write, Page Replacement</p> <p>Mass-Storage Structure: Overview, Disk Structure, Disk Scheduling, Disk Management.</p>	15	

	<p>File-System Interface: File Concept, Access Methods, Directory and Disk Structure.</p> <p>File-System Implementation: Allocation Methods, Free-Space Management.</p>	
<p>Textbooks:</p> <ul style="list-style-type: none">• Abraham Silberschatz, Peter Galvin, Greg Gagne, Operating System Concepts, Wiley, 2021 <p>Additional References:</p> <ul style="list-style-type: none">• Achyut S. Godbole, Atul Kahate, Operating Systems, Tata McGraw Hill, 2017• Naresh Chauhan, Principles of Operating Systems, Oxford Press, 2014• Andrew S Tanenbaum, Herbert Bos, Modern Operating Systems, 4e Fourth Edition, Pearson Education, 2016		

Course Code	MAJOR SEM – II - Database Systems	Credits	Lectures /Week
K23USCSMJ212	Paper II	2	2
Course Outcomes:			
After successful completion of this course, students would be able to			
<ul style="list-style-type: none"> • Describe different concepts of Database System. • Discuss the concept Schema and Relational Algebra. • Apply ER Model and Relational Model concepts. • Compare and analyze the performance of File organizations. 			
Unit	Topics	No of Lectures	
I	<p>Introduction to DBMS – Database, DBMS – Definition, Overview of DBMS, File Systems versus a DBMS, Advantages of DBMS, Levels of abstraction, Data independence, DBMS Architecture</p> <p>Data models - Client/Server Architecture, Object Based Logical Model, Record Based</p> <p>Logical Model (relational, hierarchical, network)</p> <p>INTRODUCTION TO DATABASE DESIGN : Database Design and ER Diagrams, Entities, Attributes, and Entity Sets , Relationships and Relationship Sets, Additional Features of the ER Model(Key Constraints , Participation Constraints , Weak Entities , Class Hierarchies , Aggregation) Conceptual Design With the ER Model (Entity versus Attribute, Entity versus Relationship, Binary versus Ternary Relationships, Aggregation versus Ternary Relationships)</p> <p>Introduction to the Relational Model (Creating and Modifying Relations Using SQL), Integrity Constraints over Relations(Key Constraints, Foreign Key Constraints, General Constraints),Enforcing Integrity Constraints(Transactions and Constraints),Querying Relational Data,</p> <p>Logical Database Design: ER to Relational (Entity Sets to Tables ,Relationship Sets (without Constraints) to Tables, Translating Relationship Sets with Key Constraints, Translating</p>	15	

	Relationship Sets with Participation Constraints ,Translating Weak Entity Sets ,translating Class Hierarchies, Translating ER Diagrams with Aggregation)	
II	<p>Schema refinement and Normal forms: Functional dependencies, first, second, third, and BCNF normal forms based on primary keys, lossless join decomposition.</p> <p>Relational Algebra operations (selection, projection, set operations union, intersection, difference, cross product, Joins –conditional, equi join and natural joins, division)</p> <p>Database Protection: Security Issues, Threats to Databases, Security Mechanisms, Role of DBA, Discretionary Access Control</p> <p>Storage and Indexing: Data on External Storage File Organizations and Indexing (Clustered Indexes, Primary and Secondary Indexes Index Data Structures, Hash-Based Indexing , Tree-Based Indexing)</p> <p>Comparison of File Organizations: (Cost Model,Heap Files,Sorted Files,Clustered Files,Heap File with Unclustered Tree Index, Heap File With Unclustered Hash Index , Comparison of I/O Costs)</p>	15
<p>Textbooks:</p> <ul style="list-style-type: none"> ● Ramez Elmasri & Shamkant B.Navathe, Fundamentals of Database Systems, Pearson Education, Seventh,Edition, 2017 ● Ramakrishnam, Gehrke, Database Management Systems, McGraw-Hill, Third Edition, 2014 <p>Additional References:</p> <ul style="list-style-type: none"> ● Database System Concepts by Abraham Silberschatz, Henry F. Korth, et al., 7th Edition, 21 July 2021 		

Course Code	Principles of Operating Systems – Practical	Credits	Lectures/Week
K23USCSMJ21		2	4

Part A

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.

1	Process Communication:
2	Write a program to give a solution to the Producer–Consumer problem using shared memory.
3	Process Communication:
4	Write a program to give a solution to the Producer–Consumer problem using message passing.
5	Synchronization:
6	Write a program to give a solution to the Bounded Buffer problem.
7	Synchronization:
8	Write a program to give a solution to the Readers–Writer’s problem.
9	Write a program that implements FCFS scheduling algorithm.
10	Write a program that implements (with no preemption) scheduling algorithm.

Part B

Database Systems – Practical

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 concept of Database systems.
- Solve the problems based on each concept of Database System.
- Analyze the output for the given input

1	Perform Following: <ul style="list-style-type: none"> • Create ER model with <ul style="list-style-type: none"> - Key Constraints
---	--

	<ul style="list-style-type: none"> - Participation Constraints - Weak Entities - Class Hierarchies - Aggregation
2	<p>Perform Following:</p> <ul style="list-style-type: none"> • Create Relational Model <ul style="list-style-type: none"> - Create and Modify Relations - Integrity Constraints over Relations(Key Constraints, Foreign Key Constraints, General Constraints)
3	<p>Perform Following:</p> <ul style="list-style-type: none"> • Create Relational Model • Enforcing Integrity Constraints(Transactions and Constraints) • Querying Relational Data
4	<p>Perform Following:</p> <ul style="list-style-type: none"> • Convert ER to Relational (Entity Sets to Tables ,Relationship Sets (without Constraints) to Tables • Translating Relationship Sets with Key Constraints • Translating Relationship Sets with Participation Constraints
5	<p>Perform Following:</p> <ul style="list-style-type: none"> • Convert ER to Relational (Entity Sets to Tables ,Relationship Sets (without Constraints) to Tables • Translating Weak Entity Sets • Translating Class Hierarchies • Translating ER Diagrams with Aggregation
6	Perform Relational algebra operations on above ER Model.
7	Perform Normalization and schema refinement on above ER and Relational Model
8	Perform Following: Create Clustered index
9	Perform Following: File Tables: Create, Alter and drop. Load Files
10	Perform Following: Create database User, Grant and Deny

Course Code	MINOR SEM – II - Discrete Mathematics	Credits	Lectures /Week
K23USCSMR221	Paper I	2	2
Course Outcomes:			
After successful completion of this course, students would be able to			
<ul style="list-style-type: none"> • Describe the basic concepts of sets, permutations, relations, graphs, trees • Understand sets and perform operations and algebra on set • Apply the concepts of graphs to solve various problems in day to day life. • Demonstrate an understanding of relations and functions and be able to determine their properties • Illustrate relationships using directed, undirected, weighted graphs and trees 			
Unit	Topics	No of Lectures	
I	<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>	15	
II	<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>	15	

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

Course Code	OPEN ELECTIVE SEM – II – Introduction to Data Science	Credits	Lectures /Week
K23USCSOE231	Paper I	2	2
Course Outcomes:			
After successful completion of this course, students would be able to			
<ul style="list-style-type: none"> • To learn the tools and techniques of data science. • To understand how to use the methods for handling data. • To apply principles of Data Science to study business problems. • To adapt easily to the changes and new demands from industry. 			
Unit			
Unit		Topics	
No of Lectures			
I	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.		15
II	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		15
Textbooks:			
<ul style="list-style-type: none"> • “Doing Data Science”, Cathy O’Neil and Rachel Schutt, O’Reilly, 2015 			
Additional References:			
<ul style="list-style-type: none"> • An Introduction to Statistical Learning with Applications in R Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani: Springer US 2 and 2021 			

Course Code	OPEN ELECTIVE SEM – II - E-Commerce	Credits	Lectures /Week
K23USCSOE232	Paper II	2	2
Course Outcomes:			
After successful completion of this course, students would be able to			
<ul style="list-style-type: none"> • To know how the business is carried out through electronic media • To understand the types and ways of doing business over internet • To apply the knowledge after becoming a professional or an entrepreneur • To analyze the security concerns while transacting using electronic media 			
Unit			
Unit	Topics	No of Lectures	
I	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.	15	
II	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	15	
Textbooks:			
<ul style="list-style-type: none"> • Electronic Commerce: A Managerial Perspective, Turban, E. et al., Prentice Hall-2008. • Frontiers of e-commerce, Ravi Kalakota, Pearson. 			
Additional References:			
<ul style="list-style-type: none"> • Electronic Business and Electronic Commerce Management, 2nd edition, Dave Chaffey, Prentice Hall, 2006 • . e-Learning Tools and Technologies, Horton and Horton, Wiley Publishing 			

Course Code	VOCATIONAL SKILL COURSE SEM – II - Linux (Practical Based)	Credits	Lectures/ Week
K23USCSVC241	Paper I	2	2
Course Outcomes:			
After successful completion of this course, students would be able to			
<ul style="list-style-type: none"> • Linux Basic Shell Commands and its usage. • Demonstrate the creation and execution of shell scripts in Linux. • Elaborate use the various file system and administration commands to execute shell scripts in linux. • Execute various programming language programs in Linux Platform • Examine the working of different programming language in Linux and work with processes. 			
Unit	Topics	No of Lectures	
I	<p>Linux OS Basics: History, Various Linux Distributions, Unix/Linux OS architecture, Features of Unix/Linux, Use of Linux OS.</p> <p>Basics of Shell: Starting the shell, Shell prompt, Command structure, File Systems and Directory Structure, man pages, more documentation pages</p> <p>Basic Bash shell commands: General purpose utility Commands, Basic commands</p> <p>Files Handling in Linux: 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>Advanced Bash shell commands: Simple Filters Commands, Filters using regular expressions. Linux environment variable: Default shell environment variables, Using command aliases</p>	15	
II	<p>Linux File Security: Types of file ownership, changing ownership of files, Different file Permission, Changing file permission, Working with types of Files Hardlink and Softlink</p> <p>Linux Security: Understanding Linux Security, Linux Administrator its role and responsibility, uses of root, sudo command, working with passwords</p>	15	

	<p>Basic Networking Concept: Networking Concepts, Working with basic networking commands ipconfig, telnet, ip, hostname, ping, netstat</p> <p>Working with Editors:</p> <p>Basic Shell scripting: Using multiple commands, creating script files and executing script files, displaying messages, using variables, basic arithmetic operators, Exiting the script.</p> <p>Using structured commands: Working with if-then, if-then-else and nested if statements, test command, Compound condition testing, while command, until command, case command.</p> <p>Script and Process control: Handling signals, Running scripts in background mode, Job scheduling commands: ps, nice, renice, at, batch</p>	
<p>Textbooks:</p> <ul style="list-style-type: none"> ● “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. <p>Additional References:</p> <ul style="list-style-type: none"> ● “Linux Administration: A Beginner's Guide”, Fifth Edition, Wale Soyinka, Tata McGraw-Hill, 2008. ● “Linux: Complete Reference”, Richard Petersen, 6th Edition, Tata McGraw-Hill ● “Beginning Linux Programming”, Neil Mathew, 4th Edition, Wiley Publishing, 2008. 		

Course Code	SKILL ENHANCEMENT COURSE SEM – II - Introduction to SQL	Credits	Lectures /Week
K23USCSSC251	Paper I	2	2
<p>Course Outcomes: After successful completion of this course, students would be able to</p> <ul style="list-style-type: none"> • Describe different concepts of SQL. • Understand the concepts of SQL. • Apply SQL concepts. • Analyze the performance of File organizations. 			
Unit	Topics	No of Lectures	
I	<p>What is SQL, Benefits of SQL, Downloading and installing SQL Server, Downloading and Installing SQL Server Express, Downloading SQL Server Management Studio(SSMS), Starting the database engine services, Connect to SQL Server with SSMS, Basics of SSMS and its features, Logical process for executing queries in SQL Server.</p> <p>Create Login, Creating SQL Database User (Understanding the Types of Users, Selecting type of User, Create users with SSMS, Additional options, Create User using T-SQL), Create a database schema, Join a Role, Grant permission to a principal role, Create a server Role and Application Role, Create Credentials. Note: Perform above all operations using SSMS and transact-SQL.</p> <p>SQL Operators, SQL Data types</p> <p>DDL Statements - Creating Databases, Using Databases, datatypes, Creating Tables (with integrity constraints – primary key, default, check, not null), Altering Tables, Renaming Tables, Dropping Tables, Truncating Tables, Backing Up and Restoring Databases.</p> <p>DML Statements – Viewing the structure of a table insert, update, delete, Select all columns, specific columns, unique records, conditional select, in clause, between clause, limit, aggregate functions (count, min, max, avg, sum), group by clause, having Clause</p>	15	
II	<p>Functions – String Functions (concat, instr, left, right, mid, length, lcase/lower, ucase/upper, replace, strcmp, trim, ltrim, rtrim), Math Functions (abs, ceil, floor, mod, pow, sqrt, round,</p>	15	

	<p>truncate) Date Functions (adddate, datediff, day, month, year, hour, min, sec, now, reverse)</p> <p>Stored procedures (Create, Execute, create with multiple parameter)</p> <p>Joining Tables – inner join, outer join (left outer, right outer, full outer)</p> <p>Subqueries – subqueries with IN, EXISTS, subqueries restrictions, Nested subqueries, ANY/ALL clause, correlated subqueries</p> <p>Views (creating, altering dropping, renaming and manipulating views)</p> <p>DCL Statements (creating/dropping users, privileges introduction, granting/revoking privileges, viewing privileges)</p> <p>Indexing (Create and drop)</p>	
<p>Textbooks:</p> <ul style="list-style-type: none"> ● Ramakrishnam, Gehrke, Database Management Systems, McGraw-Hill, Third Edition, 2014 ● Ramez Elmasri & Shamkant B.Navathe, Fundamentals of Database Systems, Pearson Education, Sixth Edition, 2010 <p>Additional References:</p> <ul style="list-style-type: none"> ● Learn SQL: A Practical Guide for SQL Server and Database Fundamentals by Jacob Schulz, 2021 ● SQL Programming The Ultimate Guide for Beginner's to Advance by A. Khan, January 2022 ● Querying SQL Server: Run T-SQL operations, data extraction, data manipulation, and custom queries to deliver simplified analytics by Adam Aspin, July 2022 		

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.

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

· All questions shall be compulsory with internal choice within the questions.

· Each Question may be sub-divided 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.