GANDHI INSTITUTE OF TECHNOLOGY AND MANAGEMENT (GITAM)
(Deemed to be University, Estd. u/s 3 of UGC Act 1956)
VISAKHAPATNAM ∗HYDERABAD ∗BENGALURU
Accredited by NAAC with 'A^{++}' Grade

REGULATIONS AND SYLLABUS

of

Master of Computer Applications

w.e.f 2024-2025 batch

Department of Computer Science
GITAM School of Science

Website: www.gitam.edu

Master of Computer Applications (M.C.A)
REGULATIONS
(W.e.f. 2024-2025 admitted batch)

1. ADMISSION

Admission into 2-year M.C.A program of GITAM University is governed by GITAM University admission regulations.

2. ELIGIBILITY CRITERIA

Passed BCA/Bachelor Degree in Computer Science Engineering or Equivalent Degree

OR

Passed B.Sc/B.Com/B.A with Mathematics at 10+2 level or graduation level (with additional bridge Courses as per the norms of the concerned University).

Obtained at least 50% marks (45%marks in case of candidates belonging to reserved category) in the qualifying examination.

Admission into M.C.A (Master Computer Applications) will be based on an All India GITAM Science Admission Test (GSAT) conducted by GITAM University and the rule of reservation, wherever applicable.

3. CHOICE BASED CREDIT SYSTEM

Choice Based Credit System (CBCS) is introduced with effect from the admitted Batch of 2015-16 based on UGC guidelines in order to promote:

Student Centered Learning

Cafeteria approach

Inter-disciplinary learning

Learning goals/ objectives and outcomes are specified leading to what a student should be able to do at the end of the program.

4. STRUCTURE OF THE PROGRAM

The Program Consists of

Foundation Courses (compulsory) which give general exposure to a Student in communication and subject related area.

Core Courses (compulsory).

Discipline centric electives which are supportive to the discipline give expanded scope of the subject give their disciplinary exposure nurture the student skills

Open electives are of general nature either related or unrelated to the discipline.

Practical Proficiency Courses, Laboratory and Project work.

Each course is assigned a certain number of credits depending upon the number of contact hours (lectures/tutorials/practical) per week.

In general, credits are assigned to the courses based on the following contact hours per week per semester.

One credit for each Lecture / Tutorial hour per week.

One credit for two hours of Practical per week.

Eight credits for project.

The curriculum of the Four Semesters M.C.A program is designed to have a total of 88 credits for the award of M.C.A degree.

5. MEDIUM OF INSTRUCTION

The medium of instruction (including examinations and project reports) shall be in English.

6. REGISTRATION

Every student has to register himself / herself for each semester individually at the time specified by the Institute / University.

7. ATTENDANCE REQUIREMENTS

A student whose attendance is less than 75% in all the courses put together in any semester will not be permitted to attend that end - semester examination and he/she will not be allowed to register for subsequent semester of study. He/she has to repeat the semester along with his / her juniors

However, the Vice Chancellor on the recommendation of the Principal / Director of the Institute/School may condone the shortage of attendance to the students whose attendance is between 66% and 74% on genuine grounds and on payment of prescribed fee.

8. EVALUATION

The assessment of the student's performance in a Theory course shall be based on two components: Continuous Evaluation (40 marks) and Semester-end examination (60 marks).

A student has to secure an aggregate of 40% in the course in continuous and semester end examinations the two components put together to be declared to have passed the course, subject to the condition that the candidate must have secured a minimum of 24 marks (i.e. 40%) in the theory component at the semester-end examination.

Practical / Viva voce etc. course are completely assessed under Continuous Evaluation for a maximum of 100 marks and a student has to obtain a minimum of 40% to secure Pass Grade. Details of Assessment Procedure are furnished below in Table 1.

Table 1: Assessment Procedure

S. No.	Component of assessment	Marks allotted	Type of Assessment	Scheme of Examination
1	Theory	40	Continuous	(i) One Mid Exam will be conducted for 30 Marks. (ii) 10 marks are allocated for Assignments.
		60	Semester-end examination	The semester-end examination shall be for a maximum of 60 marks.
	Total	100		
2	Practical	100	Continuous evaluation	60 marks for performance, regularity, record/ and case study. Weightage for each component shall be announced at the beginning of the semester. 40 marks (30 marks for experiment(s) and 10 marks for practical Viva-voce.) for the test conducted at the end of the Semester conducted by the concerned lab Teacher.
	Total	100		
3	Project work	200	Project evaluation	150 marks for evaluation of the project work dissertation submitted by the candidate. 50 marks are allocated for the project Viva-Voce. The project work evaluation and the Viva-Voce shall be conducted by one external examiner outside the University and the internal examiner appointed by the Head of the Department.

9. RETOTALING &REVALUATION

Re-totaling of the theory answer script of the semester-end examination is permitted on request by the student by paying the prescribed fee within one week after the announcement of theresults.

Revaluation of the theory answer scripts of the semester-end examination is permitted on request by the student by paying the prescribed fee within one week after the announcement of theresult.

10. PROVISION FOR ANSWER BOOK VERIFICATION & CHALLENGE EVALUATION:

If a student is not satisfied with his/her grade after revaluation, the student can apply for, answer book verification on payment of prescribed fee for each course within one week after announcement of revaluationresults.

After verification, if a student is not satisfied with revaluation marks/grade awarded, he/she can apply for challenge valuation within one week after announcement of answer book verification result/ two weeks after the announcement of revaluation results, which will be valued by the two examiners i.e., one Internal and one External examiner in the presence of the student on payment of prescribed fee. The challenge valuation fee will be returned, if the student is succeeded in the appeal with a change for a bettergrade.

11. SUPPLEMENTARY EXAMINATIONS & SPECIAL EXAMINATIONS:

The odd semester supplementary examinations will be conducted on daily basis after conducting regular even semester examinations in April/May.

The even semester supplementary examinations will be conducted on daily basis afterconducting regular odd semester examinations during November/December

A student who has completed his/her period of study and still has **"**F**"** grade in final semester courses is eligible to appear for Special Examination normally held during summer vacation.

12. PROMOTION TO THE NEXT YEAR OF STUDY

A student shall be promoted to the next academic year only if he/she completes the academic requirements of 60% of the credits till the previous academic year.

Whenever there is a change in syllabus or curriculum, he/she has to continue the course with new regulations after detention as per the equivalency established by the BoS to continue his/her further studies

13. BETTERMENT OF GRADES

A student who has secured only a pass or second class and desires to improve his/her class can appear for betterment examinations only in **'**n**'** (where **'**n**'** is no.of semesters of the program) theory courses of any semester of his/her choice, conducted in summer vacation along with the Special Examinations.

Betterment of Grades is permitted **'**only once**'**, immediately after completion of the program of study.

14. REPEAT CONTINUOUS EVALUATION

A student who has secured **'F'** grade in a theory course shall have to reappear at the subsequent examination held in that course. A student who has secured **'F'** grade can improve continuous evaluation marks upto a maximum of 50% by attending special instruction classes held during summer.

A student who has secured **'F'** grade in a practical course shall have to attend Special Instruction classes held during summer.

A student who has secured **'F'** grade in a combined (theory and practical) course shall have to reappear for theory component at the subsequent examination held in that course. A student who has secured **'F'** grade can improve continuous evaluation marks upto a maximum of 50% by attending special instruction classes held during summer.

The RCE will be conducted during summer vacation for both odd and even semester students. Student can register a maximum of 4 courses. Biometric attendance of these RCE classes has to be maintained. The maximum marks in RCE be limited to 50% of Continuous Evaluation marks. The RCE marks are considered for the examination held after RCE except for final semester students.

RCE for the students who completed course work can be conducted during the academic semester. The student can register a maximum of 4 courses at a time in slot of 4 weeks. Additional 4 courses can be registered in the next slot.

A student is allowed to Special Instruction Classes (RCE) **'only once'** per course.

15. GRADING SYSTEM

Based on the student performance during a given semester, a final letter grade will be awarded at the end of the semester in each course. The letter grades and the corresponding grade points are as given in Table 2.

Table 2: Grades & Grade Points

Sl.No.	Grade	Grade Points	Absolute Marks
1	O (outstanding)	10	90 and above
2	A+ (Excellent)	9	80 to 89
3	A (Very Good)	8	70 to 79
4	B+ (Good)	7	60 to 69
5	B (Above Average)	6	50 to 59
6	C (Average)	5	45 to 49
7	P (Pass)	4	40 to 44
8	F (Fail)	0	Less than 40
9	Ab. (Absent)	0	-

A student who earns a minimum of 4 grade points (P grade) in a course is declared to have successfully completed the course, subject to securing an average GPA (average of all GPAs in all the semesters) of 5 at the end of the Program to declare pass in the program.

Candidates who could not secure an average GPA of 5 at the end of the program shall be permitted to reappear for a course(s) of their choice to secure the same.

16 GRADE POINT AVERAGE

A Grade Point Average (GPA) for the semester will be calculated according to the formula:

$$GPA = \frac{[C*G]}{C}$$

Where

C = number of credits for the course,
G = grade points obtained by the student in the course.

To arrive at Cumulative Grade Point Average (CGPA), a similar formula is used considering the student's performance in all the courses taken, in all the semesters up to the particular point of time.

CGPA required for classification of class after the successful completion of the program is shown in Table 3.

Table 3: CGPA required for award of Class

Class	CGPA Required
First Class with Distinction	**≥ 8.0***
First Class	**≥ 6.5**
Second Class	**≥ 5.5**
Pass Class	**≥ 5.0**

In addition to the required CGPA of 8.0 or more the student must have necessarily passed all the courses of every semester in first attempt.

17. ELIGIBILITY FOR AWARD OF THE M.C.A DEGREE

Duration of the program: A student is ordinarily expected to complete M.C.A program in Four semesters of two years. However, a student may complete the program in not more than Four years including study period.

However the above regulation may be relaxed by the Vice Chancellor in individual cases for cogent and sufficient reasons.

A student shall be eligible for award of the M.C.A Degree if he / she fulfills all the following conditions.

Registered and successfully completed all the courses and projects.

Successfully acquired the minimum required credits as specified in the curriculum corresponding to the branch of his/her study within the stipulated time.

Has no dues to the Institute, hostels, Libraries, NCC / NSS etc, andNo disciplinary action is pending against him / her.

The degree shall be awarded after approval by the Academic Council.

18. DISCRETIONARY POWER

Not with standing anything contained in the above sections, the Vice Chancellor may review all exceptional cases, and give his decision, which will be final and binding.

19. Vision and Mission of the Department of Computer Science

Vision of the Department

To become a leading hub for education and innovation in computer science, empowering students with emerging technologies for global tech leadership through pioneering research and active community engagement.

Mission of the Department

M1: Foster a new generation of skilled computer science professionals through a well- structured curriculum that encourages continuous learning and prepares students for diverse, dynamic careers in emerging technologies.

M2: Conduct robust research in emerging fields of computer science and engage in strategic collaborations with industry and community partners to make significant contributions to society.

M3: Uphold the highest ethical standards, transparency, and accountability while fostering inclusivity and diversity in pushing the boundaries of technological advancement.

20. Program Educational Objectives, Program Outcomes

Program Educational Objectives

PEO1: Graduates will pursue advanced degrees or engage in research in core and emerging areas of computer science, contributing to the advancement of the field.

PEO2: Graduates will become successful entrepreneurs or excel professionally, applying their skills in core and emerging areas of computer science to address societal challenges.

PEO3: Graduates will become competent professionals in industry, academia, and other organizations, continually adapting to evolving technologies in core and emerging areas of computer science.

Program Outcomes (POs)

PO1: Foundation Knowledge
Apply knowledge of mathematics, programming logic and coding fundamentals for solution architecture and problem solving.

PO2: Problem Analysis
Identify, review, formulate and analyses problems for primarily focusing on customer requirements using critical thinking frameworks.

PO3: Development of Solutions
Design, develop and investigate problems with as an innovative approach for solutions incorporating ESG/SDG goals.

PO4: Modern Tool Usage
Select, adapt and apply modern computational tools such as development of algorithms with an understanding of the limitations including human biases.

PO5: Individual and Teamwork
Function and communicate effectively as an individual or a team leader in diverse and multidisciplinary groups. Use methodologies such as agile.

PO6: Project Management and Finance
Use the principles of project management such as scheduling, work breakdown structure and be conversant with the principles of Finance for profitable project management.

PO7: Ethics
Commit to professional ethics in managing software projects with financial aspects. Learn to use new technologies for cyber security and insulate customers from malware.

PO8:Life-long learning
Change management skills and the ability to learn, keep up with contemporary technologies and ways of working.

Master of Computer Applications (M.C.A.)
Scheme of Instruction
I SEMESTER

SNo	Course Code	Course Title	**Category**	**L**	**T**	**P**	**C**
1	MATH1381	Mathematical Foundations in Computer Science	PC	3	1	0	4
2	CSCI6111	Programming with C and C++	PC	3	1	0	4
3	CSCI6131	Operating Systems: Concepts and Design	PC	3	0	0	3
4	CSCI6151	Front-End Development	PC	3	0	0	3
5	IENT1051	Fundamentals of Entrepreneurship	SSE	2	0	0	2
6	CSCI6121	Programming with C and C++ Lab	PP	0	0	2	1
7	CSCI6141	Fundamentals of Operating Systems lab	PP	0	0	2	1
8	CSCI6161	Front-End Development Lab	PP	0	0	2	1
		Total					19

II SEMESTER

SNo	Course Code	Course Title	Category	L	T	P	C
1	MATH1391	Probability and Statistics with R programming	PC	3	1	0	4
2	CSCI6181	Data Structures	PC	3	0	0	3
3	CSCI6201	Database Systems: Concepts and Design	PC	3	0	0	3
4	CSCI6241	Object Oriented Software Engineering	PC	3	1	0	4
5	CSCI6221 CSCI6251	**Generic Elective -I** (a) Programming With Core Java (b) Programming with Python	GE	3	0	0	3
6	CSCI6271	Technical Communication Skills	PP	2	0	0	2
7	CSCI6191	Data Structures Lab	PP	0	0	2	1
8	CSCI6211	Fundamentals of Database Management Systems Lab	PP	0	0	2	1
9	CSCI6231 CSCI6261	Core Java Lab/ Python Programming Skills Lab	PP	0	0	2	1
		Total					22

III SEMESTER

SNo	Course Code	Course Title	Category	L	T	P	C
1	CSCI7141 CSCI7161	**Generic Elective -II** Server –Side Programming with Java OR Server –Side Programming with Python	GE	3	0	0	3
2	CSCI7181	Design and Analysis of Algorithms	PC	3	0	0	3
3	CSCI7191	Artificial Intelligence and Machine Learning	PC	3	0	0	3
4	CSCI7201 CSCI7211 CSCI7231	**Generic Elective – III** (a)Business Intelligence (b)Agile Software Development (c) Cloud Computing: Concepts and Technologies	GE	3	0	0	3
5	CSCI7251 CSCI7261 CSCI7271 CSCI7281 CSCI7291	**Generic Elective – IV** (a)Computer Networks (b) Data Mining: Concepts and Techniques (c) Introduction to Parallel Computing (d) Cryptography and Network Security (e) Digital Forensics	GE	3	0	0	3
6	CSCI7301 CSCI7311 CSCI7321 CSCI7331	**Skill Enhancement Courses** (a) Digital Marketing (b) E-Commerce Strategies and Technologies (c) Social Network Analysis (d)Compiler Design	SEC	3	0	0	3
7	CSCI7341	Machine Learning Lab	PP	0	0	2	1
8	CSCI7151 CSCI7171	Server-Side Programming with Java Lab OR Server-Side Programming with Python Lab	PP	0	0	2	1
9	CSCI7351 CSCI7221 CSCI7241	(a) Business Intelligence using Tableau/Power BI Lab (b) Agile Software Development Lab (c) Cloud computing Lab	PP	0	0	2	1
10	CSCI7361	Internship/Industrial Training and Seminar	PP	0	0	4	2
		Total					23

IVSEMESTER

SNo	Course Code	Course Title	Category	L	T	P	C
1	CSCI7371 CSCI7391 CSCI7421 CSCI7411 CSCI7451	**Generic Elective – V** (a) Introduction to Big Data Analytics (b) Cyber Security : Concepts and Technologies (c) Blockchain Technologies: Concepts and Applications (d) Deep Learning and Generative AI (e) Fundamentals of Natural Language Processing	GE	3	0	0	3
2	CSCI7461 CSCI7471 CSCI7481 CSCI7491 CSCI7501	**Generic Elective –VI** (a) Fundamentals of Digital Image Processing (b) Introduction to Bioinformatics (c) Introduction to NOSQL Database (c) Software Process Management (d) Software Testing and Quality Assurance	GE	3	0	0	3
3	CSCI7381 CSCI7401 CSCI7421 CSCI7441	(a) Big Data Insights Lab (b) Cyber Security Lab (c) Blockchain Technologies Lab (d) Deep Learning Techniques Lab	PP	0	0	2	1
4	HSMCH102	Universal Human Values 2: Understanding Harmony	PC	2	1	0	3
5	PROJ7999	Project	PP	0	0	3	8
6	VIVA7999	Comprehensive -Viva					2
		Total					20

Total Credits: 19+22+23+20= 84 Credits

DISCRETE MATHEMATICS

L	T	P	S	J	C
3	1	0	0	0	4

Course Description:
Discrete Mathematics provides a foundation for understanding the mathematical underpinnings of computer science. This course covers a wide array of topics crucial to the field, including mathematical logic, set theory, lattices, Boolean algebra, and graph theory. Each unit introduces fundamental concepts and explores their applications, enhancing analytical and problem-solving skills essential for computing and engineering disciplines.

Course Objectives

- To understand the difference between primitive statement and compound statement.
- To learn the basic concept and applications of theory of inference for the statement calculus and predicate calculus.
- To develop an ability to define a set , relation and a function with their properties.
- To attain an ability to implement features of lattices and Boolean algebra.
- To understand the concept of graphs, directed graphs, and trees.

UNIT - I

Mathematical Logic: Statements and Notation, Connectives, Normal Forms, The Theory of Inference for the Statement Calculus, The Predicate Calculus, Inference Theory of the Predicate Calculus. 10Hours

UNIT - II

Set Theory: Basic Concepts of Set Theory, Relations and Ordering.
Functions: Definitions and Introduction, Composite of Functions, Inverse Functions, Binary and n-ary Operations, Characteristic Function of a Set. 10Hours

UNIT - III

Lattices : Definition and Examples, Properties of Lattices, Sub lattices, Direct Product and Homomorphism, Some Special Lattices. 10Hours

UNIT - IV

Boolean Algebra: Definition and Examples, sub algebra, Direct product and Homomorphism, Boolean Functions, Boolean forms and free Boolean Algebras, Values of Boolean expressions and Boolean functions, Representation of Boolean functions, Minimization of Boolean functions, Karnaugh maps. 10Hours

UNIT - V

Graph Theory: Graphs, Multi graphs, Directed Graphs, Complete, Regular and Bipartite Graphs, Planar Graphs, Tree Graphs, Labeled and Weighted Graphs, Basic Definitions, Sub- graphs, Isomorphic Graphs, Paths, Connectivity, The Bridges of Konigsberg, Traversable Multi- graphs. Rooted Trees, Sequential Representation of Directed Graphs, Warshall's Algorithm and Shortest Path (Minima) Algorithm only. 10Hours

Text Books:

1. Discrete Mathematical Structures with applications to computer science by J.P. Tremblay and R. Monohar, Tata McGraw – Hill.
2. Discrete Mathematics by Seymour Lipschutzand Marc Lipson, Schaum's outlines, Tata McGraw Hill.
3. Introductory Methods of Numerical Analysis by S.S Sastry, Prentice – Hall India.

Reference Books:

1. Discrete Mathematics and its Applications by Kenneth H.Rosen, Tata McGraw – Hill.
2. Numerical Methods for Engineers by Steven C. Chopra and Raymond P.Canale, McGraw Hill

Course Outcomes

At the end of the Course, the student is able to

- Demonstrate proficiency in mathematical logic
- Apply set theory in various contexts
- Utilize lattice and Boolean algebra in problem-solving
- Analyze and interpret various types of graphs
- Design and evaluate algorithms using graph theory

CO-PO Mapping:

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1	3	2	1	1	1	-	-	2
CO2	3	2	2	1	1	-	-	2
CO3	3	2	2	1	1	-	-	2
CO4	3	3	2	2	2			3
CO5	3	3	3	3	2			3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handlenecessary mathematical orientation, programming techniques and concept based learning.	

PROGRAMMING WITH C AND C++

L	T	P	S	J	C
3	1	0	0	0	4

Course Description:
This course introduces the fundamentals of programming with a focus on the C and C++ programming languages, offering a comprehensive exploration from basic to advanced programming concepts. It is designed to provide a strong foundation in procedural programming using C and an introduction to object-oriented programming using C++. The course covers a range of topics from basic syntax and control structures in C to the sophisticated features of C++ including classes, inheritance, and exception handling.

Course Objectives
To understand the fundamentals of C and C++ programming
To develop proficiency in using pointers and dynamic memory management
To master object-oriented programming concepts in C++
To apply advanced features of C++ in real-world applications
To design and implement effective software solutions

UNIT –I

Overview of C:History and characteristics of Structure of a C program, Compilation process, **Basic Syntax :**Keywords and identifiers, Data types and variables, Input/output functions **Control Structures**, Conditional statements (`if`, `else`, `switch`),Looping statements (`for`, `while`, `do-while`), Arrays and strings handling. Pointers concepts and functions, Function declaration and definition, Call by value and call by reference, Recursion, Structures 10Hours

UNIT-II

Object-Oriented Programming with C++Concepts
Object-Oriented Programming Features, Difference between Procedure oriented and object-Oriented Programming, input/output statements
Classes and Objects: Defining classes and objects**,** Structures and Classes ,access specifiers, Member functions, Scope Resolution Operator, Friend Functions, Friend Classes, Inline Functions, Static Data Members, Static Member Functions, this pointer ,pointer concepts, Call by value and call by reference, Passing Objects to Functions, Returning Objects, Object Assignment 10Hours

UNIT- III

Constructors, Parameterized Constructors, Constructors with One Parameter,When Constructors and Destructors are executed.

Function Overloading, Copy Constructors, and Default Arguments: Function Overloading, Copy Constructors, and Default Arguments, Overloading Constructor Functions, Default Function Arguments, Default Arguments vs. Overloading.

Operator Overloading: Operator Overloading , Creating a Member Operator Function, Creating Prefix and Postfix Forms of the Increment and Decrement Operators , Overloading the Shorthand Operators, Operator Overloading Restrictions , Operator Overloading Using a Friend Function, Using a Friend to Overload ++ or – – 10Hours

UNIT-IV

Inheritance: Inheritance, Base-Class Access Control, Inheritance and protected Members, Protected Base-Class Inheritance, Inheriting Multiple Base Classes, Constructors, Destructors,and Inheritance, When Constructor and Destructor Functions Are Executed , Passing Parametersto Base-Class Constructors, Granting Access , Virtual Base Classes.

Virtual functions: Virtual Functions and Polymorphism, Virtual Functions, calling a Virtual Function Through a Base Class Reference, The Virtual Attribute Is Inherited, Virtual Functions Are Hierarchical, Pure Virtual Functions, Abstract Classes. 10Hours

UNIT-V

Exception Handling: Exception Handling, Exception Handling Fundamentals, Catching Class Types, Using Multiple catch Statements, Handling Derived-Class Exceptions, Exception Handling, Catching All Exceptions, Restricting Exceptions, Rethrowing an Exception, Understanding terminate( ) and unexpected( ) Setting the, and Unexpected Handlers, The uncaught exception( ) Function. 10Hours

Text Books

1. C: The Complete Reference" by Herbert Schildt
2. C Programming Language" by Brian W. Kernighan and Dennis M. Ritchie
3. The C++ Complete Reference by Herbert Schildt, 4th Edition, 2017ooks

Reference Books

1. Mastering C++ by Venugopal K R, RajkumarBuyya , Tata McGraw Hill, 2nd edition, 2013.
2. C++ Programming Language by BjarneStroustrup, Addison-Wesley Professional, 4th edition, 2013.
3. C++ Primer by Barbara E Moo, Stanley B. Lippman , JoseeLajoie, Pearson Education, 4th edition, 2007.

Course Outcomes

At the end of the Course, the student is able to

- **CO1 :** Understand the fundamental concepts of C programming, including syntax, control structures, functions, pointers, arrays, and strings. *(L2: Understand)*
- **CO2 :** Apply object-oriented programming principles such as classes, objects, access specifiers, and function overloading to solve programming problems. *(L3: Apply)*
- **CO3 :** Analyze the use of constructors, destructors, operator overloading, and copy constructors for effective code design. *(L4: Analyze)*
- **CO4 :** Evaluate inheritance and polymorphism concepts, including virtual functions, to design

modular and reusable code. *(L5: Evaluate)*

- **CO5 :** Develop robust programs using exception handling techniques to ensure error-free execution. *(L6: Create)*

CO-PO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	2	3	1	0	0	2
CO2	3	2	2	3	1	0	0	2
CO3	3	3	2	3	2	0	0	3
CO4	3	3	3	3	2	0	0	3
CO5	3	3	3	3	3	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handle necessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Mr.B.Satyanarayana Verified by Dr.M.Srivenkatesh

OPERATING SYSTEMS: CONCEPTS AND DESIGN

L	T	P	S	J	C
3	0	0	0	0	3

Course Description:
This course provides an in-depth exploration of operating system concepts, architectures, and functionalities. It covers the fundamental roles of an operating system in managing hardware and software resources, process management, memory management, file systems, and I/O systems. The course also addresses advanced topics such as multithreading, synchronization, deadlocks, and virtual memory management. Special attention is given to distributed systems, special-purpose systems, and open-source operating systems

Course Objectives

- To understand the fundamental concepts and architecture of operating systems
- To gain proficiency in process and thread management
- To master synchronization and deadlock handling
- To develop skills in memory and storage management
- To apply knowledge to manage I/O systems and improve system performance

UNIT - I

Introduction: Introduction to Operating Systems, Computer System Architecture, operating System Structure, Operating System Operations, Distributed Systems, Special Purpose Systems, Computing Environments, Open-Source Operating Systems.

System Structures: Operating System Services, User Operating Systems, System Interface, System Calls, Types of System Calls, System Programs, Operating System Design and Implementation Operating System Structure,Virtual Machines. 9 Hours

UNIT - II

Process Concept: Process Concept, Process Scheduling, Operations on Processes, Inter process Communication, Communication in Client Server Systems.

Multithreaded Programming: Overview, Multithreading Models, Thread Libraries, Threading Issues, Operating System Examples.

Process Scheduling: Basic Concepts, Scheduling Criteria and Algorithms, Thread Scheduling, Multiple Processor Scheduling, Real Time CPU Scheduling. 9 Hours

UNIT -III

Synchronization: Background, Critical Section Problem, Peterson's Solution, Synchronization Hardware, Semaphores, Classic Problems of Synchronization, Monitors.

Deadlocks: System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock.

9 Hours

UNIT - IV

Memory Management Strategies: Background, Swapping, Contiguous Memory Allocation, Paging, Structure of the Page Table, Segmentation.

Virtual Memory Management: Background, Demand Paging, Copy on Write, Page replacement, Allocation of Frames, Thrashing, Other Considerations.

9 Hours

UNIT - V

File System: File Concept, Access Methods, Directory and Disk Structure, File-System Mounting, File Sharing, Protection.

Implementing File Systems: File-System Structure, File-System Implementation, Directory Implementation, Allocation Methods, Free-Space Management, Efficiency and Performance.

Mass Storage Structure: Disk Structure, Disk Scheduling, Disk Management.

. 9 Hours

Text Books

1. Operating System Concepts by Abraham Silberschatz, Peter B. Galvin, Greg Gagne, Wiley Publications, 10th edition, 2018.

Reference Books

1.Modern Operating Systems" by Andrew S. Tanenbaum and Herbert Bos, 2nd edition

2.Operating Systems by Achyut S. Godbole, Tata McGraw Hill, 3rd edition,2010.

3. Operating Systems: Internals and Design Principles by William Stallings, PearsonEducation, 9th edition,2011.

4. Operating Systems: A Concept-based Approach by Dhamdhere, D.M.,McGraw Hill, 2nd edition 006.

Course Outcomes

At the end of the Course, the student is

- **CO1:** Understand the basic concepts and structures of operating systems, including system architecture, services, and system calls. *(L2: Understand)*
- **CO2:** Analyse process concepts, multithreading models, and CPU scheduling algorithms to optimize system performance. (L4: Analyse)
- **CO3:** Apply synchronization techniques and deadlock management strategies to ensure smooth system operations. *(L3: Apply)*
- **CO4:** Evaluate memory management strategies and virtual memory mechanisms for efficient utilization of system resources. *(L5: Evaluate)*
- **CO5:** Analyse file system structures and mass storage management techniques to enhance system functionality. *(L4: Analyse)*

CO-PO Mapping table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	1	1	0	0	0	1
CO2	3	3	2	2	1	0	0	1
CO3	3	3	2	2	1	0	0	1
CO4	3	3	2	2	1	0	0	1
CO5	3	3	2	2	1	0	0	1

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handlenecessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Dr.M.Seshashayee

Verified by Dr.M.Srivenkatesh

FRONT-END DEVELOPMENT

L	T	P	S	J	C
3	0	0	0	0	3

Course Description

This course provides a comprehensive introduction to front-end web development, focusing on the fundamental technologies of HTML, CSS, JavaScript, and React. Students will learn to create and style web pages using HTML and CSS, implement dynamic functionality with JavaScript, and build modern, interactive web applications using React. The course covers essential concepts such as document structure, styling, responsive design, scripting, component-based architecture, state management, and routing. Through hands-on lab exercises, students will gain practical experience in developing user interfaces and single-page applications.

Course Objectives

- To master HTML for Web Content Creation
- To utilize CSS for Advanced Styling
- To implement Dynamic Functionality with JavaScript
- To build Interactive Web Applications with React
- To develop Real-World Front-End Projects

UNIT-I

Introduction to HTML: Information files creation, Web Server, Web Client/Browser, Hyper Text Markup Language, Commonly used HTML Commands.

Lists: Types of lists Adding Graphics to HTML Documents: Using the Attributes- Border, Width, and Height, Align and Alt Attributes.

Tables: Introduction, The Caption Tag, Using the width and boarder, Cellpadding, Cellspacing,Using Background-Color property, Using the Colspan and Rowspan Attribute

Input from Elements :Text, button , submit , reset , cancel , file ,date ,email ,checkbox , radio, Select Element , Textarea Element 9Hours

UNIT-II

Cascading style Sheet: Introduction to CSS3 , CSS Syntax, Selectors, Background Properties ,Text Properties ,Fonts Properties , Lists Properties , Tables Properties , Box Model, Display Positioning, **CSS pseudo classes:** Element display state pseudo-classes , Input pseudo-classes , Location pseudo-classes , User action pseudo-classes. 9 Hours

UNIT-III

Introduction To JavaScript: JavaScript in web pages, The Advantages of JavaScript, Writing JavaScript Programming Constructs, Conditional Checking, Super controlled endless loops, Functions in JavaScript, User defined functions, Placing text in a Browser, Dialog Boxes, JavaScript HTML DOM. 9 Hours

UNIT-IV

React: Introduction to installation process of Node.js and npm, Introduction to React, Add React to a website, create a new React App, Introducing JSX, Rendering Elements, Components of Props, State and Lifecycle, Handling events. 9 Hours

UNIT-V

React Routers , REST API , Conditional Rendering, Lists and keys, Forms, Building single page applications with React . 9 Hours

Textbooks

1. "Learning Web Design: A Beginner's Guide to HTML, CSS, JavaScript, and Web Graphics" by Jennifer Robbins, O'Reilly Media,5th Edition, 2018
2. "Learning React: Functional Web Development with React and Redux" by Alex Banks and Eve Porcello, O'Reilly Media, 2nd Edition, 2020

Recommended Textbooks

1. "HTML and CSS: Design and Build Websites" by Jon Duckett**,**Publisher**:** Wiley,1st Edition, 2011

2. React Up and Running: Building Web Applications" by Stoyan Stefanov,Publisher**:** O'Reilly Media,1st Edition, 2016

3. Full stack React: The Complete Guide to ReactJS and Friends" by Anthony Accomazzo, Nathaniel Murray, Ari Lerner, Fullstack.io,1st Edition, 2017

Course Outcomes

After completing the course student will be able to:

- **CO1:** Understand the fundamentals of HTML, including creating lists, adding graphics, creating tables, and using input elements to design structured web pages. (L2: Understand)
- **CO2:** Apply CSS3 properties, including selectors, text, fonts, tables, box model, and pseudo-classes, to style web pages effectively. *(L3: Apply)*
- **CO3:** Develop interactive web pages using JavaScript programming constructs, including loops, functions, dialog boxes, and HTML DOM manipulation. *(L3: Apply)*
- **CO4:** Design and build React components to create dynamic, reusable, and efficient user interface elements for scalable applications. *(L6: Design)*
- **CO5:** Design and implement single-page applications (SPAs) using React features such as React Routers, REST API integration, conditional rendering, lists, and forms. *(L6: Design)*

Co-PO Mapping table

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1	3	2	3	3	1	1	0	2
CO2	3	2	3	3	1	1	0	2
CO3	3	3	3	3	2	1	0	3
CO4	3	3	3	3	2	1	0	3
CO5	3	3	3	3	3	1	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	

4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handlenecessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Mr.B.Satyanarayana

Verified by Dr.M.Srivenkatesh

FUNDAMENTALS OF ENTREPRENEURSHIP

Course Code: IENT1051	**Course Title**: Fundamentals of Entrepreneurship						
Semester: II	**Course Type: Core**	**L**	**T**	**P**	**S**	**J**	**C**
		0	**0**	**4**	**0**	**0**	**2**
Home Programme(s): UG Courses							
Course Leader:							

Introduction

Entrepreneurship is a vital life skill that fosters curiosity, creativity, and a focus on seizing opportunities. By embracing entrepreneurship, individuals can achieve professional independence, tackle complex challenges with innovative solutions, and take calculated risks. This course, "Introduction to Entrepreneurship," is designed to provide students with essential knowledge and practical skills for their entrepreneurial journey. Contrary to popular belief, entrepreneurship can indeed be learned, and this course dispels those myths. It offers a comprehensive understanding of the entire entrepreneurial process, from generating ideas to launching a minimum viable product (MVP). Through a combination of theory and hands-on activities, students will explore various aspects of entrepreneurship, such as identifying opportunities, discovering customers, designing solutions, and employing lean startup methods. To succeed, students must demonstrate self-direction and a genuine enthusiasm for learning, whether independently or in collaboration with peers.

Learning Objectives

S. No.	Learning Objective
1	Understand the fundamental concepts and processes of entrepreneurship.
2	Identify and evaluate business ideas and opportunities.
3	Know the techniques for effective problem-solving.
4	Understand the customer and the customer discovery process and how to develop market insights.
5	Effectively pitch your Venture Idea

Course outline and indicative content

Unit I: Entrepreneurial Process and Mindset **P-12**

Introduction to Entrepreneurship, Pilot Your Purpose, Innovation, Risk-Taking and Value Creation, Myths around Entrepreneurship, Distinct Types of Entrepreneurship, Entrepreneurial vs. Managerial Mindset.

Unit II: Problem Identification and Ideation **P-12**

Entrepreneurship Opportunity identification, Market and Need Analysis, Problem Discovery, Problem Statement Identification and definition, Evaluating and Selecting Ideas

Unit III: Customer Discovery & Market Insights **P-12**

Users and Buyers, Target Group and Persona, Customer Research Methods (People Shadowing,

laddering etc.), Use Cases, Market Sizing & Segmentation, Customer Value Proposition

Unit IV: Solution Design P-12

Principles of Effective Solution Design, Prototyping Methods and Tools, Building and Testing Prototypes, Gathering Feedback on Prototypes, Iterating and Refining Solutions, Building Minimum Viable solution.

Unit V: Crafting your Venture Narrative P-12

How you can launch a successful venture. Tell your venture story

Course Outcomes

On successful completion of this course, students will be able to:

S. No.	Learning Outcome	Assessment
1	To discover skills and competencies needed for entrepreneurial career	A1
2	Effectively utilize frameworks for business planning and development.	A3
3	Implement customer research methods such as shadowing, laddering etc to gather insightful data.	A2
4	Build and refine a minimum viable product (MVP) based on real customer feedback.	A3
5	Present a process pitch that integrates learnings across all units to propose a viable entrepreneurial venture.	A4

Assessment Methods

Task	Task type	Task mode	Weightage (%)
A1	**Class Participation and Activities**: Engagement in class discussions, group activities, and case studies throughout the course.	Individual	20
A2	**Problem Statement and Ideation Report**: A detailed report identifying a market problem, supported by a Problem Statement Canvas.	Group	20
A3	**Customer Discovery Assignment**: A comprehensive analysis of target customers, including persona creation and market sizing.	Group	20
A4	**Process Pitch**: Share your learning from the course	Group	40

**as per grouping made by the course facilitator (no deviation permitted)*

Evaluation pattern

A1: Classroom Participation and Engagement

a) Class Participation – 5 Marks.
b) Group discussions- 5 Marks,
c) Group Activity – 5 Mark
d) Case Study discussion- 5 Marks.

A2: Problem Statement and Ideation.

a) Problem Identification - 5 Marks.
b) Drawings / Prototype Product or Service-5 Marks
c) Discussion on Market Survey-5 Marks
d) Problem Statement Canvas-5 Marks.

A3: Customer Discovery Assignment

a) Analysis on Target Customers - 10 Marks.
b) Report on Market Size - 10 Marks

A4: Process Pitch

a) Presentation from Problem Identification to Launching a Product or Service - 40 Marks.

Learning and Teaching Activities

In classrooms

Reflection videos, Case Discussions, Simulations

Outside classrooms

Field Visits

Teaching and Learning Resources

"Entrepreneurship: Theory, Process, and Practice" by Donald F. Kuratko

https://www.teachingentrepreneurship.org/ Justin Wilcox

Other Books

- The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses by Eric Ries
- Blank, S. and Dorf, B. (2012) The Startup Owner's Manual: The Step-by-Step Guide for Building a Great Company. BookBaby, Pennsauken.
- Neck, Heidi & Greene, Patricia & Brush, Candida. (2014). Teaching entrepreneurship: A practice-based approach. 10.4337/9781782540564.

Documentaries

- Bloomberg Game Changers (e.g. Zuckerberg, Brin & Page; Jobs, Musk, etc.) - YouTube
- Elon Musk: The future we're building and boring | TED – YouTube
- Inspirational series about the entrepreneurial path of 5 of the most admired business entrepreneurs: Cornelius Vanderbilt (Railroads), John D. Rockefeller (Oil), Andrew Carnegie (Steel), J.P. Morgan (Banking) and H. Ford (Automobile)
- 6 Tips on Being a Successful Entrepreneur | John Mullins | TED – YouTube
- Social Entrepreneurship - The Journey of Lakshmi Menon: https://open.spotify.com/episode/3frmNkjUNCZgXCbLsPpfve?si=fd13d7efa85741eb
- Keep the spirit of customer centricity alive with Zoho I YouTube
- Blinkit's Genius Strategy that stunned Amazon and Flipkart | Business Case Study – YouTube
- How is Zerodha's GENIUS Business strategy CRUSHING its competition? Zerodha vs Upstox Case study

Learning articulation (LO – PO mapping and SDG mapping)

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:			
BOS	**:<< date >>**	**ACADEMIC COUNCIL**	**:04-07-2024**
SDG No. & Statement	**: 8**		
Promote sustained, inclusive and sustainable economic growth, full and productive employment and decent work for all			
SDG Justification:			
To achieve sustained per capita income growth and ensure higher economic productivity, focus should be on youth by grooming them to be creative and innovative, have productive employment and quality of life through Skill development and Entrepreneurship			

PROGRAMMING WITH C AND C++ LAB

L	T	P	S	J	C
0	0	2	0	0	1

Course Description:

This laboratory course complements the theoretical knowledge acquired in the "Introduction to Programming in C and C++" course. It focuses on practical implementation of programming concepts using C and C++ languages. Students will engage in hands-on programming exercises that cover a broad spectrum of topics from basic syntax and control structures to advanced concepts such as object-oriented programming, inheritance, and exception handling. Each lab session is designed to strengthen students' understanding and skills in programming with real-world coding exercises.

Course Objectives:

- To apply programming concepts and techniques using C and C++.
- To develop proficiency in using arrays, strings, pointers, and data structures.
- To understand and implement object-oriented programming concepts such as classes, objects, inheritance, and polymorphism.
- To practice advanced features of C++ including operator overloading, exception handling, and templates.
- To enhance problem-solving skills and software development capabilities.

Programs

1. Write a program demonstrates basic syntax, input/output, conditional statements, and looping.
2. Write a program to demonstrate handling arrays and string operations.
3. Write a program covers pointers, pointer arithmetic, and dynamic memory allocation.
4. Write a program demonstrates function usage, including recursion.
5. Write program illustrates the use of structures.
6. Write program to demonstrate classes and objects
7. Write a program to demonstrate different types of Constructors
8. Write program to demonstrate for friend class and friend functions
9. Write program for inline function, Static function, Static Classes
10. Write program to demonstrate for passing objects to a function, returning objects to function
11. Write a program to demonstrate Function Overloading
12. Write a program to demonstrate Copy Constructor
13. Write a program to demonstrate Default Arguments to a Function
14. Write a program to demonstrate different types of operator overloading
15. Write a program to demonstrate single inheritance
16. Write a program to demonstrate Multiple and Multilevel Inheritance
17. Write a program to demonstrate virtual base class
18. Write a program to demonstrate virtual functions
19. Write a program to demonstrate Try and catch block
20. Write a program to demonstrate Multiple catch Statements
21. Write a program to demonstrate Generic Functions
22. Write a program to demonstrate Generic Classes

Text Book

1. C: The Complete Reference" by Herbert Schildt
2. C Programming Language" by Brian W. Kernighan and Dennis M. Ritchie
3. C++ : The Complete Reference by Herbert Schildt 4th Edition, 2017

Reference Books

1. Object Oriented Programming in C++ by E. Balaguruswamy, 4rd Edition, Tata McGraw Hill Publication.
2. Let Us C++ by YashavantP.Kanetkar, 2nd Edition, BPB Publications.

Course Outcomes:

After completing the course student will be able to:

- **CO1:** Understand and implement basic C++ programming concepts such as syntax, input/output operations, control structures, arrays, strings, pointers, and recursion. (*L2: Understand)*
- **CO2:** Apply object-oriented programming principles, including classes, objects, constructors, friend functions, and inline/static functions, to create modular and efficient code. *(L3: Apply)*
- **CO3:** Analyse advanced programming concepts such as function overloading, copy constructors, operator overloading, and default arguments to optimize functionality in C++. *(L4: Analyse)*
- **CO4:** Evaluate inheritance and polymorphism mechanisms, including single, multiple, multilevel inheritance, virtual base classes, and virtual functions, to design reusable and dynamic code. *(L5: Evaluate)*
- **CO5:** Develop robust and reusable programs using exception handling and generic programming features such as templates for functions and classes. *(L3: Develop)*

CO-PO Mapping table

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1	3	2	2	3	1	0	0	2
CO2	3	3	2	3	2	0	0	3
CO3	3	3	3	3	2	0	0	3
CO4	3	3	3	3	2	0	0	3
CO5	3	3	3	3	2	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handlenecessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Dr.M.Srivenkatesh Verified by Dr.M.Srivenkatesh

FUNDAMENTALS OF OPERATING SYSTEMS LAB

L	T	P	S	J	C
0	0	2	0	0	1

Course Description:
This laboratory course is designed to provide hands-on experience with operating system utilities and programming, focusing on Unix/Linux environments. Students will explore the fundamental components of Unix file systems, including file handling, process management, disk management, and networking commands. Key utilities such as the vi editor, shell scripting, and text processing will be central to the coursework.

Course Objectives:
- To Understand the Unix file system and command-line interface
- To Develop proficiency in Unix utilities
- To Implement shell scripts
- To Simulate operating system concepts
- To Explore system calls and processes

Unix Utilities – Introduction to Unix file system, vi editor, file handling utilities, security by file permissions, process utilities, disk utilities, networking commands, cp, mv ln, rm, unlink, mkdir, rmdir, du, df, mount, umount find, unmask, ulimit, ps, who, finger, arp, ftp, telnet, rlogin, text processing utilities and backup utilities, detailed commands to be covered are cat, tail, head, sort, nl,uniq, grep, egrep, fgrep, cut, paste, join, tee, pg, comm., cmp, diff, tr, awk, tar,cpio.

1. Study about the General Purpose Utilities.
   a) Banner b) cal c) date d) calendar e) tty f) bc g)
   spell &fspell2.Write a shell program using if, nested if
3. Write a shell program using switch case
4. Write a shell program to find the Sum of digits in a 3 digit number using while loop.
5. Write a shell program to print first 'n' terms of Fibonacci series using for loop

6. Programs on Processes:
   1. Chain of processes.
   2. Fan of processes.

7. Write a program or script to demonstrate basic operating system commands like file creation, deletion, directory navigation, and permissions.

8. Write a program to demonstrate the use of system calls for file manipulation (open, read, write, close)

9. Write a program to simulate various process scheduling algorithms (FCFS, SJF, Round Robin, Priority Scheduling).

10. Write a program to simulate memory allocation techniques (first fit, best fit, worst fit).

11. Write a program to simulate page replacement algorithms (FIFO, LRU, Optimal).

12. Write a program to perform basic file operations (create, open, read, write, delete) and demonstrate file handling system calls.

13. Write a program to implement directory operations (create, delete, search) and handle directory structure.

14. Write a program to simulate disk scheduling algorithms (FCFS, SSTF, SCAN, C-SCAN).

Text Books

1. Operating System Concepts by Abraham Silberschatz, Peter B. Galvin, Greg Gagne, Wiley Publications, 10th edition, 2018.

2. Unix Concepts and Applications by Sumitabha Das, Tata McGraw Hill, 4th edition,2006.

3. Unix networking program by Stevens W. Richard,2005.

Reference Books

1. Modern Operating Systems" by Andrew S. Tanenbaum and Herbert Bos, 2nd edition

2.Operating Systems by Achyut S. Godbole, Tata McGraw Hill, 3rd edition,2010.

3.Advanced Unix programming by H.J.Rechkind, Pearson Education, 2nd edition,2004.

Course Outcomes

Upon completion of the course, the student is able to

- **CO1:** Understand and utilize basic Unix utilities, file handling commands, text processing utilities, and general-purpose commands to manage Unix systems. *(L2: Understand)*

- **CO2:** Apply shell scripting constructs, including conditionals, loops, and case structures, to automate system tasks. *(L3: Apply)*

- **CO3:** Analyse process management techniques, including process creation, scheduling algorithms, and memory allocation methods, to optimize system performance. *(L4: Analyse)*

- **CO4:** Evaluate file and directory operations, including file manipulation system calls and directory handling techniques, for efficient data management in Unix. *(L5: Evaluate)*

- **CO5:** Develop and simulate advanced system-level algorithms, such as process scheduling and page replacement, to enhance system efficiency. *(L3: Develop)*

CO-PO Mapping table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	1	2	1	1	1	2
CO2	3	3	2	3	2	1	1	2
CO3	3	3	2	3	2	1	1	2
CO4	3	3	2	3	2	1	1	2
CO5	3	3	3	3	2	1	1	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	

SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handlenecessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Dr.M.Seshashayee Verified by Dr.M.Srivenkatesh

FRONT–END DEVELOPMENT LAB

L	T	P	S	J	C
0	0	2	0	0	1

Course Description:
This laboratory course is designed to provide hands-on experience in web development, covering both the front-end and back-end aspects essential for full stack development. It focuses on practical applications of HTML, CSS, JavaScript, Bootstrap, and jQuery to build responsive and interactive web applications. Through a series of lab exercises, students will learn to create web pages, style them, and add dynamic functionality using various web technologies.

Course Objectives:

- To understand the structure and elements of an HTML document to create well-formed web pages.
- To learn to style web pages using CSS, including the box model and positioning techniques.
- To develop interactivity on web pages using JavaScript for operations, DOM manipulation, and event handling.
- To build dynamic web applications using basic Node.js setup and React components.
- To develop Real-World Front-End Projects

1. **Create a Basic HTML Document**: Structure an HTML page with common elements like headers, images,paragraphs, and links.
2. **Implement Various Types of Lists**: Use ordered, unordered, and description lists in an HTML document.
3. **Construct a Simple Table**: Create a table with headers and multiple rows, adjusting border, padding, and spacing, colspan and rowspan, background colors to specific cells and rows
4. **Input Forms**: Create a form with various input types like text, date, email, file, checkboxes, and radio buttons.
5. **Form Submission**: Create a complete form for user registration including handling of different input types. Use HTML attributes to style buttons and inputs, text areas
6. **CSS Basics**: Style a document by changing the font, color, and size of text using CSS.
7. **CSS Box Model**: Experiment with margin, border, padding, and content width and height.
8. **CSS Positioning**: Use different positioning properties like absolute, relative, fixed, and sticky.
9. **Styling Lists and Tables with CSS**: Apply custom styles to lists and tables to improve their visual presentation.
10. **Basic JavaScript Operations**: Perform arithmetic operations and display results in the browser.

11. **Conditional Statements**: Use if-else and switch cases to handle different conditions in a script.
12. **JavaScript Loops**: Implement loops to display numbers from 1 to n on a web page. Various sample examples using loops
13. **DOM Manipulation**: Change the content of HTML elements using JavaScript.
14. **Event Handling**: Attach events to HTML elements like buttons and links to trigger JavaScript functions.
15. Explain Install and configure of Node.js and npm.
16. Create and render a JSX element that includes an expression to display the current date and time.
17. Use the Welcome component in src/App.js to display greetings for at least three different names.
18. Create a class component that displays the current time, updating every second.

19. Create a form with input fields and a submit button, and handle the form submission to display the input values.
20. Create a simple to-do list application that allows users to add, remove, and display to-do items.

Text Book:

1. Web Enable Commercial Application Development Using HTML, Java script, DHTML and PHP by Ivan Bayross, BPB Publications, 4th revised edition, 2010 .

Reference Books:

1. Complete Reference HTML by T. A. Powell, 3rd edition, TMH, 2003.
2. HTML, XHTML, and CSS Bible by Steven M. Schafer, Wiley India, 5th Edition.
3. Beginning CSS: Ca20SCAding Style Sheets for Web Design by Ian Pouncey, Richard York, Wiley India.
4. Web Technology and Design by Xavier, C, New Age International, 2013.

Course Outcomes

Upon completion of the course, the student is able to

CO1: Understand the structure and elements of HTML to create web pages using headers, images, paragraphs, links, lists, tables, and forms. *(L2: Understand)*

CO2: Apply CSS properties, including box model, positioning, and custom styling, to enhance the visual appearance of web elements. *(L3: Apply)*

CO3: Develop interactive web pages using JavaScript programming constructs like loops, conditional statements, DOM manipulation, and event handling. *(L6: Develop)*

CO4: Apply React features such as JSX, components, props, state, lifecycle, and form handling to build dynamic user interfaces. *((L3: Apply)*

CO5: Develop and implement single-page web applications, such as a to-do list, using React and JavaScript to manage and display dynamic content. *(L6: Develop)*

CO-PO Mapping table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	3	3	1	1	0	2
CO2	3	2	3	3	1	1	0	2
CO3	3	3	3	3	2	1	0	3
CO4	3	3	3	3	2	1	0	3
CO5	3	3	3	3	3	1	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	

The topics included in this course are designed to get acquainted with one of the skills that handle necessary mathematical orientation, programming techniques and concept based learning.

Prepared by Mr.B.Satyanarayana

Verified by Dr.M.Srivenkatesh

PROBABILITY AND STATISTICS WITH R PROGRAMMING

L	T	P	S	J	C
3	1	0	0	0	4

Course Description:

This course provides a comprehensive understanding of probability and statistics principles integrated with practical R programming applications. The course covers fundamental probability concepts, various types of random variables, probability distributions, and statistical techniques such as correlation, regression, and hypothesis testing. Students will learn to apply R programming for statistical computations, data visualization, and the implementation of advanced statistical methods. The integration of theoretical knowledge with R programming aims to equip students with the skills to handle real-world data analysis and statistical problem-solving.

Course Objectives:

- Fundamental Understanding of Probability
- Mastery of Probability Distributions
- Correlation and Regression Analysis
- Hypothesis Testing with Large Samples
- Hypothesis Testing with Small Samples

UNIT - I

Probability: Sample Space, Events, Axiomatic Approach to Probability, Conditional Probability Independent Events, Baye's Formula with Applications.

Random Variables: Continuous and Discrete Random Variables, Distribution Function of a random variable, Expectation, Variance, Coefficient of Variation, Moment Generation Function.

R Programming Integration:

Introduction to R and RStudio, Basic R commands for probability computations, Manipulating Vectors, Matrices, Data Frames.

Implementing Conditional Probability and Bayes' theorem in R . 12 Hours

UNIT - II

Probability Distribution: Discrete Distributions, Binomial, Poisson and Geometric Distributions, Continuous Distributions, Uniform, Normal, Exponential.

R Programming Integration:

Using R to generate and visualize discrete distributions

Use in-built Functions in R for Binomial (dbinom() , pbinom() , qbinom() , and rbinom()), Poisson (dpois()`, `ppois()`, `qpois()`, and `rpois()`) , visualizing binomial/Poission distribution using R .

Generating and visualizing continuous distributions using R

Use R in-built functions for Uniform, Normal dnorm(), pnorm, and qnorm(), rnorm()) and visualizing normal distribution using R . 12 Hours

UNIT - III

Correlation and Regression: Correlation Coefficient, Rank Correlation Coefficient of Determination, Linear Regression, Methods of Least Squares, Fitting of the Curve of the Form

ax + b, ax^2 + bx +c, ab^x , ax^b and ae^{bx}.

R Programming Integration:

Calculating and interpreting correlation in R
Performing linear regression in R
Visualizing regression lines and fitting curves using ggplot2 in R
Advanced regression techniques (Quadratic Regression) using R 12 Hours

UNIT - IV

Tests of Significance – Large Sample Tests: Introduction to sampling population, sample, Statistic, Sampling Distributions, Standard error, interval estimation, Critical Region, Two Types of Errors, Level of Significance. Large Sample Tests: Test for Single Proportion & Double Proportion. Test for Single Mean and Double Mean.

R Programming Integration:

Implementing various sampling methods in R
Calculating sampling distributions and standard errors using R
Point and interval estimation in R. 12 Hours

UNIT - V

Tests of Significance – Small Sample Tests: Definition of t-Distribution, Applications & Properties, t-test for Single Mean & Double Mean, Paired Mean t-test, t-test for Correlation coefficient. Definition of F-Distribution, Applications & Properties. Test for Equality of Sample Variances. Definition of Chi-Square Distribution, Applications & Properties. Chi-Square test for Goodness of Fit. Chi-Square test for Chi-Square test for independence.

R Programming Integration:

Hypothesis testing for large samples using R
Conducting t-tests, F-tests, and Chi-square tests in R
Interpreting results from hypothesis tests in R 12 Hours

Text Books:

1. Fundamentals of Mathematical Statistics by S.C. Gupta & V.K. Kapoor, Sultan Chand& Sons, 2002.
2. Sudha G. Purohit, Sharad D. Gore, Shailaja R. Deshmukh, Statistics Using R, Alpha Science Intl Ltd, 2015

Reference Books:

1. Probability and Statistics for Engineers by Irwin Millor and John E.Freund, PHI.
2. Probability and Statistics, Spiegel, TMH.
3. P. G. Hoel, S. C. Port and C. J. Stone, Introduction to Probability Theory, Universal Book Stall, 2003 (Reprint).

Course Outcomes

Upon completion of the course, the student is able to

- **CO1:** Focuses on understanding foundational concepts of probability and applying R for computations *(L2: Understand and L3: Apply)*
- **CO2:** Emphasizes applying knowledge of probability distributions and using R for visualization, aligning *(L3: Apply)*
- **CO3:** Involves analysing relationships between variables using regression and correlation techniques, aligning *(L4: Analyse)*
- **CO4:** Requires evaluating statistical significance using large sample tests and implementing them in R, aligning *(L5: Evaluate)*
- **CO5:** Develop on creating statistical models and performing hypothesis testing using small sample tests and R, aligning (*L3: Develop).*

CO-PO Mapping table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	2	2	1	0	0	2
CO2	3	3	2	3	1	0	0	2
CO3	3	3	2	3	2	0	0	3
CO4	3	3	2	3	2	0	0	3
CO5	3	3	2	3	2	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handlenecessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Dr.A.Sridhar,
Dr. Motahar Reza

Verified by Dr. Motahar Reza

DATA STRUCTURES

L	T	P	S	J	C
3	0	0	0	0	3

Course Description:
This course introduces students to the fundamental concepts of data structures, emphasizing their relationship with algorithms and the efficient handling of data through various data structures. It covers the theoretical foundations as well as practical implementation of data structures in a programming environment, providing students with a deep understanding of how data can be organized, stored, and manipulated efficiently.

Course Objectives:

- To understand Fundamental Concepts
- To master Linear and Non-Linear Data Structures
- To develop Proficiency in Algorithm Analysis
- To apply Data Structures to Solve Real-World Problems
- To **i**mplement and Compare Sorting and Searching Techniques

UNIT -I

Fundamental Concepts: Introduction to Data Structures, Types of Data Structures, Relationship among data, data structures and algorithms, Implementation of data structures, Analysis of Algorithms, Complexity of algorithms: Space complexity, Time complexity Definition. Asymptotic Notation

Linear Data Structure using Arrays: Sequential Organization, Linear Data Structure Using Sequential Organization: Array ADT, Single Variable Polynomial: Representation, evaluation, Addition, Multiplication, Sparse Matrix: Representation, Addition, Transpose.

9 Hours

UNIT - II

Stacks: Primitive operations, Stack Abstract Data Type, Representation of Stacks, Using Sequential Organization, Multiple Stacks, Applications of Stacks – Expression Evaluation and Conversion, Processing of Function Calls.

Queues: Concept of Queues, Queue as Abstract Data Type, Realization of Queues Using Arrays, Circular Queue, Multi queues, Dequeue, Priority Queue, Applications of Queues: Job scheduling. 9 Hours

Hours

UNIT -III

Linked Lists: Introduction, Linked List: Comparison of sequential and Linked Organizations, Terminology, Primitive operations, Realization of Linked Lists using arrays and dynamic memory management, Dynamic memory management in C++, Linked List Abstract Data Type, Linked List Variants, Doubly Linked List: Creation, Deletion, Insertion, Traversal, Circular Linked List, Linked Stack, Linked Queue. 9 Hours

UNIT - IV

Trees: Introduction, Basic terminology, General trees, Representation of a general tree, Types of Trees, Binary Tree, Properties, Binary Tree Abstract Data Type, Realization of a Binary Tree, Insertion of a Node in Binary Tree, Binary Tree Traversal (recursive traversals), Formation of binary tree from its traversals, Binary Search Tree: Inserting a node, Searching for a key, Deleting a node, Binary Tree and Binary Search Tree, AVL Trees. 9 Hours

UNIT – V

Graphs: Introduction, Graph Abstract Data Type, Graph Representation, Graph traversals, Spanning Trees: Prim's, Krushkal's Algorithm.

Searching: Search Techniques: Sequential search, Binary search, Hashed search.
Sorting: Types of sorting, General sort concepts, Bubble sort, Insertion sort, Selection sort, Quick sort, Merge sort, Heap sort. 9 Hours

Text Books:

1. Data Structures using C++ by Varsha H.Patil, Oxford University Press,2012.

Reference Books:

1. Data Structures Algorithms and Applications in C++ by SartajSahani, University Press, 2nd Edition, 2011.
2. Data Structures Using C and C++ by YedidyahLangsam, Moshe J Augenstein and Aaron M Tenenbaum, PHI, 2nd Edition, 2009.
3. Data Structures and Algorithm Analysisin C++by Mark Allen Weiss, Pearson Education, 3rd edition, 2007.
4. Data Structures and Algorithms in C++ by Adam Drozdek, Cengage Learning, 4th Edition, 2013 .

Course Outcomes:

After completing the course student will be able to:

- **CO1:** Understand the fundamental concepts of data structures, including their types, implementation, and algorithm analysis. *(L2: Understand)*
- **CO2:** Apply stack and queue data structures to solve problems such as expression evaluation, conversion, and job scheduling. *(L3: Apply)*
- **CO3:** Analyse the use of linked lists and their variants such as doubly and circular linked lists for dynamic memory management and data organization. *(L4: Analyse)*
- **CO4:** Evaluate tree data structures, including binary trees, binary search trees, and AVL trees, for efficient data storage and retrieval. *(L5: Evaluate)*
- **CO5:** Develop graph algorithms, searching techniques, and sorting methods to optimize data organization and access. *(L6: Develop)*

CO-PO Mapping table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	3	3	2	0	0	2
CO2	3	3	3	3	2	0	0	3
CO3	3	3	3	3	2	0	0	3
CO4	3	3	3	3	2	0	0	3
CO5	3	3	3	3	2	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handlenecessary mathematical orientation, programming techniques and concept based learning.	

DATABASE SYSTEMS: CONCEPTS AND DESIGN

L	T	P	S	J	C
3	1	0	0	0	4

Course Description:

This course provides a comprehensive introduction to the principles of database management systems (DBMS) and explores various data models, schema designs, SQL programming, and advanced database functionalities. Students will gain a solid foundation in both theoretical and practical aspects of database design, implementation, and maintenance.

Course Objectives:

- To understand Database Fundamentals and Architecture
- To master Data Modelling Techniques
- To develop Proficiency in the Relational Model
- To apply Normalization and Database Design Principles
- To explore Advanced Database Topics and Transactions

UNIT - I

Introduction and Conceptual Modelling, Databases and Database Users: Introduction, Characteristics of Database Approach, Actors on the Scene, Workers behind the Scene, Advantages of using DBMS Approach.

Database System, Concepts and Architecture: Data Models, Schemas and Instances, Three Schema Architecture and Data Independence, Database Language and Interfaces, The Database System Environment, Centralized and Client/Server Architecture of Database Management Systems, Classification of Database Management Systems. 9Hours

UNIT - II

Data Modeling Using The ER Model: High Level Conceptual Data Models for Database Design, Entity Types, Entity Sets, Attributes and Keys, Relationship Types, Relationship Sets, Roles and Structural Constraint, Weak Entity Types, ER Diagrams, Naming Conventions and Design Issues. The Enhanced Entity Relationship model, UML Class Diagrams, Relationship Types of Degree Higher Than Two.

The Relational Data Model and Relational Database Constraints: Relational Model Concepts, Relational Model Constraints and Relational Database Schemas, Update Operations, Transactions and Dealing with Constraints Violations. 9 Hours

UNIT – III

The Relational Algebra: Unary Relational Operations - SELECT and PROJECT, Relational Algebra Operations from Set Theory, Binary Relational Operations - JOIN and DIVISION; Additional Relational Operations, Examples of Queries in Relational Algebra.

Relational Database Design By ER And EER To Relational Mapping: Relational Database Design using ER to Relational Mapping, Mapping EER Model Constructs to Relations. 9 Hours

UNIT – IV

SQL-Schema Definition, Constraints, Queries and Views: SQL Data Definition and Data types, Specifying Constraints in SQL, Schema Change Statements in SQL, Basic Queries in SQL, More Complex SQL Queries, INSERT, DELETE, UPDATE Statements in SQL, Sub Queries, Nested Queries

Functional Dependencies and Normalization for Relational Databases: Informal Design Guidelines for Relational Schemas, Functional Dependencies, Normal Forms Based on Primary Keys, General Definitions of 2nd and 3rd Normal Forms, Boyce-Codd Normal Form

Relational Database Design Algorithms and Further Dependencies: Properties of relational Decomposition, Algorithms for Relational Database Schema design, Multi valued Dependencies and Fourth Normal Form, Join Dependencies and Fifth Normal Form. 9 Hours

UNIT – V

Transaction Processing Concepts: Introduction to Transaction Processing, Transaction and System Concepts, Desirable properties of transactions, Characterizing Schedules based on recoverability, Characterizing Schedules based on serializability

Concurrency Control Techniques: Two Phase Locking, Time Stamp Ordering, Multi version concurrency control

Database Recovery Techniques: Recovery Concepts, No-Undo/Redo Recovery Based Deferred Update, Recovery Techniques Based on Immediate Update 9 Hours

Text Books:

1. Fundamentals of Database Systems by Ramez Elmasri and Shamkant B. Navathe, Pearson education, 5th edition, 2009.

Reference Books:

1. Database Concepts by Abraham Silberschatz, Henry F Korth, S.Sudarshan, TMH, 6th edition,2014.
2. An Introduction to Database Systems by C.J. Date, Addison Wesley, 8thedition, 2008.
3. Database Management Systems by Raghu Ramakrishnan, Johannes Gehrke, TMH, 2nd edition, 2000.

Course Outcomes:

After completing the course student will be able to:

- **CO1:** Understand the foundational concepts of database systems, including database characteristics, architectures, and classification of database management systems. *(L2: Understand)*
- **CO2:** Apply data modelling techniques using ER and EER models to design high-level conceptual schemas and analyse relational database constraints. *(L3: Apply)*
- **CO3:** Analyse relational algebra operations and map ER and EER models to relational schemas for effective database design. *(L4: Analyse)*
- **CO4:** Evaluate SQL queries, normalization techniques, and relational schema design algorithms to ensure efficient and consistent database structures. (L5: Evaluate)
- **CO5:** Design robust transaction processing systems and implement concurrency control and recovery techniques to maintain database integrity. *(L3: Design)*

CO-PO Mapping table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	2	3	2	1	1	1	-	3
CO2	3	3	3	1	1	1	-	3
CO3	3	3	3	1	1	1	-	3
CO4	2	3	3	1	1	1	-	3
CO5	3	3	3	1	1	1	-	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handle necessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Dr.V.Nagalakshmi Verified by Dr.M.Srivenkatesh

OBJECT ORIENTED SOFTWARE ENGINEERING

L	T	P	S	J	C
3	1	0	0	0	4

Course Description:

This course provides a comprehensive exploration of object-oriented software engineering (OOSE), focusing on the application of software engineering principles within an object-oriented framework. It covers all phases of the software development lifecycle, from initial modeling using Unified Modeling Language (UML) to requirements elicitation, system and object design, and testing. The course integrates theoretical concepts with practical applications, including case studies and projects that simulate real-world software development scenarios.

- To understand the Fundamentals of Software Engineering
- To develop Proficiency in Modelling with UML
- To understand Software Project Organization and Communication
- To design and Manage Software Systems
- To apply Testing and Project Management Practices

UNIT-I

Software Engineering: Software related problems, software engineering, concepts, development activities.
Modelling: Concepts, Modelling with UML. 10 Hours

UNIT-II

Project Organization & Communication: Project Organization & communication concepts and their activities.
Requirements: Requirements elicitation & its activities and managing requirements elicitation, ARENA Case Study.
Analysis: Analysis overview, concepts, activities and managing analysis, ARENA Case Study.
10 Hours

UNIT-III

System Design: Decomposing the System: System Design overview, System design concepts, and System design Activities, and Managing System Design.
System design: Addressing design goals: An overview of system design activities and concepts UML Development diagram, System design goals, Managing system design, ARENA Case Study
10 Hours

UNIT-IV

Object Design: Reusing Pattern Solutions: An overview of object design Reuse Concepts, Solution objects, inheritance and design patterns.
An Object Design: Specifying Interfaces: An overview of interface specification, interface specifications concepts & its activities and Managing object design, ARENA Case Study

10 Hours

UNIT-V

Testing: Introduction, An overview of Testing**,** Testing concepts, activities and managing testing.

10 Hours

Text Book:

1. Object-Oriented Software Engineering: Using UML, Patterns and Java, Bernd Bruegge and Allen H. Dutoit, 2nd Edition, Pearson Education Asia.

Reference Books:

1. Object-Oriented Software Engineering: Practical software development using UML and Java Timothy C. Lethbridge and Robert Laganiere , McGraw-Hill Higher education
2. An Introduction to Object Oriented Systems Analysis and Design with UML and the Unified Process, Stephen R Schach, Tata McGraw-Hill

Course Outcomes:

After completing the course student will be able to:

- **CO1:** Understand the fundamental concepts of software engineering, development activities, and modeling with UML to address software-related problems. ***(****L2: Understand****)***

- **CO2:** Apply project organization, communication strategies, and requirements elicitation techniques to analyze and manage software requirements effectively. ***(****L3: Apply****)***

- **CO3:** Analyse system design activities, including decomposing systems and addressing design goals, to create structured system architectures using UML. ***(****L4: Analyse****)***

- **CO4:** Evaluate object design solutions by reusing design patterns, specifying interfaces, and managing object design activities for efficient system development. ***(****L5: Evaluate****)***

- **CO5:** Design and manage software testing activities, including test planning, execution, and validation, to ensure the reliability and functionality of software systems. ***(****L3: Design ****)***

CO-PO Mapping table

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1	3	3	3	2	2	0	0	3
CO2	3	3	3	3	2	0	0	3
CO3	2	3	3	3	3	0	0	3
CO4	3	3	3	3	3	0	0	3
CO5	3	3	3	3	3	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handlenecessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Dr.M.Seshashayee

Verified by Dr.M.Srivenkatesh

PROGRAMMING WITH CORE JAVA

L	T	P	S	J	C
3	0	0	0	0	3

Course Description:

This comprehensive course on Java programming covers a wide range of topics from the basics of object-oriented programming to advanced Java features and GUI development using Swing. It is designed to equip students with a thorough understanding of Java as a programming language and its applications in software development.

Course Objectives:

- To understand Object-Oriented Programming Concepts
- To implement Inheritance and Polymorphism in Java
- To develop Skills in Exception Handling and Multithreading
- To utilize Java's Collections Framework and Utility Classes
- To implement GUI Applications Using Swing

UNIT – I

Introduction to Object-oriented concepts- Object-Oriented concepts, An Overview of Java, Data types, Variables and Arrays, operators, expressions, control statements, Classes, and Methods, Constructors, this keyword, static modifier (static block , static method , static variable) 9 Hours

UNIT – II

Inheritance and Polymorphism: Inheritance ,Creating Multilevel hierarchy, super keyword, using final with inheritance, Polymorphism, abstract classes, abstract methods, Interfaces- defining an interface, implementing interfaces, Nested interfaces, applying interfaces, variables in interfaces and extending interfaces.

Packages- Defining a Package, Access protection, importing packages, sub packages. 9 Hours

UNIT - III

Exception handling - Fundamentals of exception handling, Exception types, Termination or resumptive exception models, Uncaught exceptions, using try and catch, multiple catch clauses, nested try statements, throw, throws and finally, built- in exceptions, creating own exception sub classes.

Multithreading- Differences between thread-based multitasking and process-based multitasking, Java thread model, creating threads, thread priorities, synchronizing threads, inter thread communication.

9 Hours

UNIT – IV

The Collections Framework (java.util)- Collections overview, Collection Interfaces, The Collection classes- Array List, Linked List, Hash Set, Tree Set, Priority Queue, Array Dequeue. Accessing a Collection via an Iterator, Using an Iterator, The For-Each alternative, Map Interfaces and Classes, Comparators, Collection algorithms, Arrays, The Legacy Classes and Interfaces- Dictionary, Hashtable , Properties, Stack, Vector More Utility classes, String Tokenizer, Random,Scanner. 9 Hours

UNIT - V

GUI Programming with Swing– Introduction Swing, limitations of AWT, components, containers. Understanding Layout Managers.

Swing Components : JButton , JLabel, JTextField,JRadioButton ,JCheckBox, JComboBox ,JList, JScrollPane , JTextArea, JTable, JMenu, JTabbedPane, JFrame, JOptionPane ,JPanel

Event Handling- The Delegation event model- Events, Event sources, Event Listeners, Event classes, Handling mouse and keyboard events, Adapter classes, Inner classes, Anonymous Inner classes.

9 Hours

Text Books:

1. Java The complete reference, 9th edition, Herbert Schildt, McGraw Hill Education (India) Pvt. Ltd.
2. Understanding Object-Oriented Programming with Java, updated edition, T. Budd, Pearson Education.

Reference Books:

1. An Introduction to programming and OO design using Java, J. Nino and F.A. Hosch, John Wiley & sons.
2. Introduction to Java programming, Y. Daniel Liang, Pearson Education.
3. Object Oriented Programming through Java, P. Radha Krishna, Universities Press.
4. Programming in Java, S. Malhotra, S. Chudhary, 2nd edition, Oxford Univ. Press.
5. Java Programming and Object oriented Application Development, R. A. Johnson, Cengage Learning.

Course Outcomes

After completing the course student will be able to:

- **CO1:** Understand object-oriented programming concepts, including classes, methods, constructors, and static modifiers, to develop foundational Java programs. (**L2: Understand**)
- **CO2:** Apply inheritance, polymorphism, interfaces, and packages to create reusable and modular Java programs. (**L3: Apply**)
- **CO3:** Analyse exception handling mechanisms and multithreading techniques to ensure error-free and efficient Java applications. (**L4: Analyse**)
- **CO4:** Evaluate the usage of the Java Collections Framework, including collection classes, interfaces, and utility classes, to manage data efficiently. (**L5: Evaluate**)
- **CO5:** Develop graphical user interfaces and implement event handling using Swing components to develop interactive Java applications. (**L6: Develop**)

CO-PO Mapping table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	1	3	2	1	1	2
CO2	3	3	2	3	2	1	1	2
CO3	3	3	3	3	2	2	1	2
CO4	2	2	3	3	2	2	1	3
CO5	3	3	3	3	3	3	2	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handlenecessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Mr.B.Satyanarayana Verified by Dr.M.Srivenkatesh

PROGRAMMING WITH PYTHON

L	T	P	S	J	C
3	0	0	0	0	3

Course Description:

This comprehensive course introduces the fundamentals of Python programming, emphasizing both procedural and object-oriented programming techniques. It is designed to equip students with the skills needed to utilize Python's versatile features across various applications, from basic scripting to complex system developments.

Course Objectives

- To understand and Apply Python Basics
- To master Control Structures and Data Handling
- To develop Proficiency in Regular Expressions and File Operations
- To implement Object-Oriented Programming (OOP) Principles in Python
- To handle Exceptions and Write Python Scripts

UNIT – I

Python Basics: Introduction, Data Types in Python, Mutable versus Immutable, Type Casting (also called Type Conversion) in Python, Input to a Python Program. Operators in Python: Introduction Assignment (and reassignment), Overview of Operators

Functions: Introduction, Need of Functions, Basics of Functions, defining your own functions andfunction syntax, Passing variables in Function Call, Function Arguments, some special functions

Flow control: Introduction, using "if", while loop, for loop, range function, Common Errors in FlowControl. 9 Hours

UNIT – II

Strings: Introduction, Creating, Initializing and Accessing Elements of a string, Traversing a String, String Operations, Difference between Functions, Methods and Attributes, String Functions versus String Methods, A Short Note on String Module.

Lists: Introduction, Some basic concepts of Lists, Creating, Traversing, and Slicing Lists, List Functions and Methods, Nested Lists and using them as matrix. 9 Hours

UNIT – III

Dictionaries: Introduction, Basics of Dictionaries, Dictionary Functions and Methods,various operations on Dictionaries

Tuples: Introduction, Some basic concepts regarding Tuples, some Additional Topics,

Regular Expression: Introduction, basic concepts of Expressions, Special Characters, Groups of Characters and Anchors, Understanding Re Module, Some Important methods of the Re Module.

9 Hours

UNIT - IV

Object-Oriented Programming with Python: Introduction, basic concepts of Object-Oriented Programming, OOP concepts related specifically to Python, some common "Built in" Attributes andMethods of a Python Modules and classes.

Inheritance and Namespace: Introduction, Basics of Inheritance of Python, Single Inheritance, Multiple Inheritance, Concept of Namespace. 9 Hours

UNIT-V

File Operations in Python: Introduction, basics of file Operations in Python, Reading and Writing aFile, Some more Advanced concepts in File Operations, some useful Methods of the OS Module, Writing small scripts for Inserting Data in a File.

Python Exceptions: Introduction, basic concepts of Exceptions in Python, User-defined Exceptions, Built-in Exceptions. 9 Hours

Text Book:

1. Python Programming by Anurag Gupta , G.P.BIswas, McGraw Hill, 2019.

Course Outcomes:

After completing the course student will be able to:

- **CO1:** Understand Python basics, including data types, operators, functions, and flow control, to write simple and structured Python programs. *(L2: Understand)*
- **CO2:** Apply string operations and list handling techniques, including slicing, functions, and methods, to manipulate and process data efficiently. *(L3: Apply)*
- **CO3:** Analyze data structures like dictionaries and tuples, and implement regular expressions to handle advanced data manipulation and pattern matching tasks. *(L4: Analyse)*
- **CO4:** Evaluate object-oriented programming concepts in Python, including inheritance, namespaces, and module usage, to design reusable and modular code. *(L5: Evaluate)*
- **CO5:** Develop Python programs using file operations, OS module methods, and exception handling to manage data storage and error conditions effectively. *(L3: Develop)*

CO-PO Mapping table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	2	3	1	0	0	2
CO2	3	2	2	3	1	0	0	2
CO3	3	2	2	3	1	0	0	2
CO4	3	3	3	3	2	0	0	3
CO5	3	3	3	3	2	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handle necessary mathematical orientation, programming techniques and concept-based learning.	

Prepared by Dr.T.Uma Devi

Verified by Dr.M.Srivenkatesh

TECHNICAL COMMUNICATION SKILLS

L	T	P	S	J	C
2	0	0	0	0	2

Course Description:

This course aims to enhance learners' sentence formation and overall communication skills. It familiarizes learners with the use of English in various communicative contexts, helping them expand their linguistic resources and enhance their communicative competence. The course develops listening, speaking, reading, and writing skills, enabling learners to exchange information and express ideas with clarity and confidence. Additionally, it prepares learners for interviews and increases their employability by improving their communicative efficiency. Critical thinking and problem-solving skills are also cultivated through effective communication strategies.

Course Objectives:

- Make the learners proficient in forming sentences and improving communication skills.
- Familiarize learners with English usage and help them expand their linguistic resources for enhanced communicative competence.
- Develop listening, speaking, reading, and writing skills for clear and confident information exchange and idea expression.
- Prepare learners for interviews and improve their employability through enhanced communication efficiency.
- Cultivate critical thinking and problem-solving skills through effective communication.

UNIT -I

Basics of English Language: Tenses , Correction of sentences, idiomaticexpressions & Vocabulary Building

Assessment Activities:
Grammar/vocabulary Test - 10 M
Describing people, places, things and objects -10 M

UNIT -II

Effective Communication: Features of good communication, Different flows of communication, Verbal and Nonverbal communication, Barriers to effective communication & ways to overcome the barriers.

Assessment Activities:
Role Plays - 10 M
Project work (Communicating with a group of new people on & off campus and documenting the details) - 10 M

UNIT-III

LSRW Skills: Introduction to phonetics, pitch, tone, Clarity of voice & correct pronunciation of a few words, Importance of Reading, Reading for Joy, Writing styles, expressing thoughts and ideas in written words in different contexts.

Assessment Activities: Group Discussion / Listening Test -10 M
Book Review / Essay or Letter Writing - 10 M

UNIT-IV

Interview Skills: Ambiance and polemics, interviewing in different settings and for different purposes, Preparing Resume & Preparation for a job interview.

Assessment Activities: Resume Writing- 10 M
Mock interview- 10 M

UNIT -V

Presentation Skills: Types of presentations, knowing the audience, Features of impactful presentations, Preparation & Delivery.

Assessment Activities: Presentation -10 M
Assignment -10 M

Textbooks:

1. Essentials of Business Communication by Rajendra Pal & JS Korlahalli, S. Chand & Sons.
2. Advanced Communication Skills by V. Prasad, Atma Ram Publications.

Course Outcomes:

Upon completing the course, learners will be able to:

- **CO1:** Understand the basics of English grammar, tenses, idiomatic expressions, and vocabulary building to improve sentence construction and communication. *(L2: Understand)*
- **CO2:** Apply effective communication techniques, including verbal and nonverbal methods, to overcome barriers and engage in clear interactions. *(L3: Apply)*
- **CO3:** Analyze LSRW (Listening, Speaking, Reading, Writing) skills to enhance phonetic clarity, reading comprehension, and written expression in various contexts. *(L4: Analyse)*
- **CO4:** Evaluate personal and professional interview skills, including resume preparation and mock interviews, for successful performance in diverse settings. *(L5: Evaluate)*
- **CO5:** Develop impactful presentations by understanding audience needs, designing content, and delivering effectively in professional contexts. *(L3: Develop)*

CO-PO Mapping table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	2	2	2	2	3	0	0	3
CO2	2	2	2	2	3	0	0	3

CO3	2	2	2	2	3	0	0	3
CO4	2	2	2	2	3	0	0	3
CO5	2	2	2	2	3	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handlenecessary mathematical orientation, programming techniques and concept based learning.	

DATA STRUCTURES USING C++ LAB

L	T	P	S	J	C
0	0	2	0	0	1

Course Description:

This laboratory course is designed to complement the theoretical knowledge gained in Data Structures with practical, hands-on programming experiences using C++. The lab focuses on the implementation and understanding of various data structures and algorithms, crucial for software development and computational problem solving.

Course Objectives

- To Implement Search and Sort Algorithms
- To Develop Skills with Arrays and Matrices
- To Master Linear Data Structures
- To Explore Linked Lists
- To Understand Tree-Based Structures and graph–based structures

1. Implementation of Linear Search, Binary Search.
2. Implementing the following sorting methods.
   a.Bubble sort , b. Insertion sort, c. Selection Sort, d. Quick Sort, e. Merge Sort
3. Implementation of Polynomial Addition using Arrays.
4. Implementation of Sparse Matrix addition using Arrays
5. Array implementation of stack.
6. Array implementation of Queue.
7. Implementation to convert infix expressions to post fix notation, prefix notation.
8. Simple expression evaluator that can handle +,-, /,*.
9. Singly Linked List operations – insertion, deletion, display, reversal
10. Implementation of Linked Stack Operations.
11. Implementation of Linked Queue Operations.
12. Implementation of circular queue ADT using an array.
13. Implementation of Doubly LinkedList.
14. Implementation of Circular LinkedList.
15. Implementation of Binary Tree, Binary Search Tree creation, traversals.
16. Implementation of Binary Search Tree operations, insertion all cases and deletion all cases.
17. Implementation of Graph Traversals.

Reference Books:

1. Data Structures through C++ by Varsha H Patil, Oxford University Press, New Edition, 2011.
2. Data Structures through C in depth by S.K. Srivastva and DeepaliSrivastva,BPB publications,2004.
3. Data Structures and Algorithms in C++ by Adam Drozdek, Cengage Learning, 4th edition, 2013.

Course Outcomes:

After completing the course student will be able to:

- **CO1:** Apply searching and sorting techniques, including linear and binary search and sorting algorithms, to organize and retrieve data efficiently. *(L3: Apply)*
- **CO2:** Implement basic data structures like arrays, stacks, and queues to solve problems such as polynomial addition, sparse matrix operations, and expression evaluation. *(L3: Apply)*
- **CO3:** Analyse linked list operations, including singly, doubly, and circular linked lists, to manage dynamic memory efficiently. *(L4: Analyse)*
- **CO4:** Evaluate tree data structures, including binary trees and binary search trees, by implementing traversal methods and various operations. *(L5: Evaluate)*
- **CO5:** Design graph algorithms and implement graph traversal techniques to explore and manipulate graph-based data. ***(L3: Design)***

CO-PO Mapping table

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1	2	3	2	1	1	1	-	3
CO2	3	3	3	1	1	1	-	3
CO3	3	3	3	1	1	1	-	3
CO4	2	3	3	1	1	1	-	3
CO5	3	3	3	1	1	1	-	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handlenecessary mathematical orientation, programming techniques and concept based learning.	

FUNDAMENTALS OF DATABASE MANAGEMENT SYSTEMS LAB

L	T	P	S	J	C
0	0	2	0	0	1

Course Description:

This course is designed to provide students with a comprehensive understanding and hands-on experience in SQL and PL/SQL programming. Covering both basic and advanced concepts, the course focuses on data manipulation, query writing, schema management, and procedural programming in SQL databases. Students will learn to create, manipulate, and maintain database objects using SQL commands, and to write procedural code in PL/SQL for more complex database interactions.

Course Objectives

- To Learn SQL Commands
- To Implement Data Integrity and Transactions
- To Understand Advanced Query Writing
- To develop Schema Management
- To Learn about to PL/SQL

1. Creation, altering and dropping of tables using SQL.
2. Implementing integrity Constraints on tables.
3. Implementing DML Operations using SQL- Insert, Delete, Update.
4. Simple Queries to access data from Tables using Select Statement and where condition using Distinct, And, Or, Not and Order By Operators.
5. Queries Using Built in Functions:
    i. Arithmetic Functions: Sign, Abs, Ceil, Floor, Exp, Power, Log, Sqrt,
    ii. String Functions: Concat, Lpad, Rpad, Ltrim, Rtrim, Lower, Upper, Initcap, Length, Substr and Instr.
    iii. Date Functions: Sysdate, Next_Day, Add_Months, Last_Day, Months_Between, Least, Greatest, Trunc, Round
    iv. Aggregate Functions: Count, Sum, Avg, Max And Min, Group by, Having,
    v. Queries Using Conversion Functions: To_Char, To_NumberandTo_Date
    vi. Queries Using Set Operators: Union, Intersect, Minus
    vii. Queries Using Joins, Natural Join, Innerjoin, OuterJoin.
6. Queries Along with Sub Queries and Correlated Queries using Any, All, In, Exists, Notexists.
7. Creating Other Schema Objects**:** Defining Views, Creating Views, using Views to Change Data, Dropping Views, Creating Indexes andSequences.
8. Using DCL Commands: Commit andRollback.
9. Creation of Simple PL/SQL Program which includes Declaration Section, Executable Section, Select ... intoClause

10. Develop Programs that include Features of Nested If andCase.
11. Develop Programs using While Loop, For Loop, Nested Loops

Text Books:

1. SQL, PL/SQL The programming language of ORACLE by Ivan bayross, BPBpublications, 4th edition, 2009.
2. Programming Oracle triggers and Stored Procedures by Kevin Owens, PHI, 3rd Edition, 2003.

Course Outcomes:

After completing the course student will be able to:

- **CO1:** Understand the creation, alteration, and deletion of database tables and implement integrity constraints to maintain data consistency. *(L2: Understand)*

- **CO2:** Apply SQL commands to perform DML operations, retrieve data using queries, and utilize built-in functions for effective data manipulation and analysis. *(L3: Apply)*

- **CO3:** Analyse complex queries, including subqueries, joins, and set operations, to extract meaningful insights and relationships from databases. *(L4: Analyse)*

- **CO4:** Evaluate transaction control commands and schema objects such as views, indexes, and sequences to manage database operations efficiently. *(L5: Evaluate)*

- **CO5:** Design and develop PL/SQL programs using control structures like loops and conditional statements to implement advanced database functionalities. *(L3: Design)*

CO-PO Mapping table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	3	3	3	3	3	3
CO2	3	3	3	3	3	3	3	3
CO3	3	3	3	3	3	3	3	3
CO4	3	3	3	3	3	3	3	3
CO5	3	3	3	3	3	3	3	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handlenecessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Dr.V.Nagalakshmi

Verified by Dr.M.Srivenkatesh

CORE JAVA LAB

L	T	P	S	J	C
0	0	2	0	0	1

Course Description:

This hands-on laboratory course is designed to reinforce the concepts learned in Java programming lectures through practical implementation. Students will engage in a series of exercises that span basic to advanced Java features, including command-line operations, class and object manipulations, package and interface usage, multithreading, and GUI components with Swing. The lab exercises provide an opportunity for students to apply their Java programming skills to solve various problems and build functional applications.

Course Objectives

- To learn Command-Line Programming
- To enhance Object-Oriented Programming Skills
- To implement Packages and Interfaces
- To develop Multithreading and Exception Handling
- To build GUI Applications with Java Swing

Exercise Programs

1. To find the average and sum of the N numbers Using Command line argument.
2. To find the number of arguments to provide at runtime.
3. To Test the given number is prime number or not.
4. To calculate the Simple Interest and Input by the user.
5. To create a simple class to find out the Area and Perimeter of rectangle and box using super and this keyword.
6. To design a class account using the inheritance and static that show all function of bank (withdrawal ,deposit).
7. To design a class using abstract methods and classes.
8. To create a package that access the member of external class as well as same package.
9. Import the user define package and access the Member variable of classes that contained by Package.
10. To show the partial implementation of Interface.
11. To create a thread that implement the Runnable interface.
12. To accept specified number of characters as input and converts them into uppercase characters
13. To illustrate creation of threads using runnable class.(start method start each of the newly created thread. Inside the run method there is sleep() for suspend the thread for 500 milliseconds).
14. To create a class MyThread in this class a constructor, call the base class constructor, using super and starts the thread. The run method of the class starts after this. It can be observed that both main thread and created child thread are executed concurrently to get the reference to the current thread by calling currentThread() method.

15. Write a program for example of try and catch block. In this check whether the given array size is negative or not.
16. Write a program for example of multiple catch statements occurring in a program.
17. Write a program to describe usage of throws clause.
18. To create a new array list, add some colors (string) and print out the collection
19. To iterate through all elements in a linked list
20. To append the specified element to the end of a hash set.
21. Design a Simple calculator using Swing components.
22. Design a User login form using Swing components
23. Demonstrate an icon-based JList, JScrollPane , JTextArea
24. Demonstrate an icon-based JTable, JMenu, JTabbedPane,

Text Books:

1. Java The complete reference by Herbert Schildt, McGraw Hill Education Pvt. Ltd, 9th edition.
2. Understanding Object-Oriented Programming with Java by T. Budd, Pearson Education.

Course Outcomes:

After completing the course student will be able to:

- **CO1:** Apply basic Java programming concepts, including command-line arguments, control structures, and simple user-defined classes, to solve computational problems. *(L3: Apply)*
- **CO2:** Implement object-oriented programming principles, such as inheritance, abstraction, interfaces, and packages, to design modular and reusable Java applications. *(L3: Apply)*
- **CO3:** Analyse multithreading and concurrency features in Java to handle parallel execution and thread synchronization effectively. *(L4: Analyse)*
- **CO4:** Evaluate exception handling mechanisms, including try-catch blocks, multiple catch statements, and the throws clause, to develop robust Java programs. *(L5: Evaluate)*
- **CO5:** Design interactive GUI-based applications using Swing components, including forms, lists, tables, and menus, to create user-friendly Java applications. *(L3: Design)*

CO-PO Mapping table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	2	3	2	1	1	2
CO2	3	3	3	3	2	1	1	3
CO3	3	3	3	3	2	2	1	3
CO4	3	3	3	3	2	2	2	3
CO5	3	2	3	3	3	3	2	2

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	

The topics included in this course are designed to get acquainted with one of the skills that handlenecessary mathematical orientation, programming techniques and concept based learning.

Prepared by Mr.B.Satyanarayana

Verified by Dr.M.Srivenkatesh

PYTHON PROGRAMMING SKILLS LAB

L	T	P	S	J	C
0	0	2	0	0	1

Course Description:

This laboratory course is designed to provide hands-on experience with Python programming, emphasizing practical application of Python's versatile features across a range of programming tasks. Students will engage in exercises that demonstrate the fundamental and advanced aspects of Python, including data manipulation, control structures, object-oriented programming, file handling, and exception management. The lab sessions are structured to strengthen students' understanding of Python through active coding, debugging, and implementing real-world scenarios.

Course Objectives

- To demonstrate proficiency in Python
- To implement control structures such as loops and conditional statements
- To manipulate strings and lists to perform operations
- To construct and manipulate complex data structures
- To develop object-oriented programs utilizing concepts

1. Write a program to demonstrate the use of different data types in Python.
2. Write a program to perform arithmetic, logical, and bitwise operations.
3. Write a program to calculate the factorial of a number using recursion.
4. Write a program to print the Fibonacci series up to a given number using a while loop.
5. Write a program to iterate over a list using a for loop and range function.
6. Write a program to create and manipulate strings (concatenation, slicing, and indexing).
7. Write a program to count the number of vowels in a string.
8. Write a program to demonstrate string functions and methods (e.g., upper(), lower(), replace()).
9. Write a program to create a list, append elements, and access elements.
10. Write a program to demonstrate list slicing and traversing.
11. Write a program to perform matrix operations using nested lists.
12. Write a program to create a dictionary, add key-value pairs, and access values.
13. Write a program to demonstrate dictionary methods (e.g., keys(), values(), items()).

14. Write a program to create and manipulate tuples.
15. Write a program to validate an email address using regular expressions.
16. Write a program to find all occurrences of a pattern in a given string using the re module.
17. Write a program to define a class, create an object, and access attributes and methods.
18. Write a program to demonstrate single inheritance in Python.
19. Write a program to demonstrate multiple inheritance and the use of the super() function.
20. Write a program to read from a text file and display its contents.
21. Write a program to write data to a text file and append data to it.
22. Write a program to demonstrate advanced file operations using the os module.
23. Write a program to handle exceptions using try-except blocks.
24. Write a program to raise and handle user-defined exceptions.
25. Write a program to handle multiple exceptions with custom responses.

Text Book:

1. Python Programming by Anurag Gupta , G.P.BIswas, McGraw Hill, 2019.

Course Outcomes

- **CO1:** Understand and implement fundamental Python concepts, including data types, operators, control structures, and functions, to solve basic computational problems. *(L2: Understand)*
- **CO2:** Apply Python data structures, including strings, lists, dictionaries, and tuples, to manipulate and organize data efficiently. *(L3: Apply)*
- **CO3:** Analyse regular expressions and advanced string operations to validate and process complex textual data. *(L4: Analyse)*
- **CO4:** Evaluate object-oriented programming concepts, including class creation, inheritance, and the use of the super() function, to design reusable Python applications. *(L5: Evaluate)*
- **CO5:** Develop robust Python programs by performing advanced file operations and implementing exception handling mechanisms. *(L3: Develop)*

CO-PO Mapping table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	2	3	1	0	0	2
CO2	3	3	3	3	2	0	0	3
CO3	3	3	3	3	2	0	0	3
CO4	3	3	3	3	2	0	0	3
CO5	3	3	3	3	2	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handle necessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Mr.G.Ramash Naidu

Verified by Dr.M.Srivenkatesh

SERVER –SIDE PROGRAMMING WITH JAVA

L	T	P	S	J	C
3	0	0	0	0	3

Course Description:

This course offers a comprehensive exploration of server-side programming using Java technologies, focusing on developing robust and scalable web applications. It delves into various Java-based frameworks and technologies, including Servlets, JavaServer Pages (JSP), Java Database Connectivity (JDBC), Hibernate, and Spring. Through a series of practical, hands-on labs and projects, students will learn to build, deploy, and manage web applications effectively, integrating both relational database management systems and ORM technologies.

Course Objectives:

- To understand and apply Java Servlet technology.
- To develop Dynamic Web Applications
- To implement Database Interactions
- To utilize ORM Techniques
- To leverage Spring Framework

UNIT-I

Introduction to server-side programming: Role of Java in server-side development, Differences between application server and web servers.
Setting up the Java development environment: IDE setup, and server configurations Servlet lifecycle, Writing and deploying a simple servlet, servlet-to-servlet communication, session management , cookie management , URL Rewriting. 9 Hours

UNIT-II

Introduction to JSP: Basics of JSP, JSP syntax and directives, JSP implicit objects, Use of scriptlets, expression language, and JSTL (JavaServer Pages Standard Tag Library). Error handling in JSP , Cookies, HTTP session tracking, creating filters, and listeners. 9 Hours

UNIT-III

Introduction to JDBC: JDBC architecture, register driver , connecting to databases.
Performing CRUD operation: Using JDBC to create, read, update, and delete records.
Connection pooling and data source: Optimizing database connections.
Integrating JDBC with servlets and JSP: Building dynamic web applications using databases. 9 Hours

UNIT-IV

B**asics of ORM and Hibernate Setup**: Introduction to Hibernate, Concept of ORM, configuring Hibernate, session management.
Hibernate Operations: CRUD operations, querying with HQL, introduction to criteria API.
Associations and Caching: Mapping relationships, utilizing first-level and second-level caches. 9 Hours

UNIT-V

Spring Core: Dependency injection, bean lifecycle, application context configuration.

Spring MVC and Spring Boot: Setting up Spring MVC, understanding controllers, Spring Boot basics for rapid development.
Spring Data JPA: Integrating Hibernate with Spring, repository abstraction layer, transaction management. 9 Hours

Text Books :

1. Head First Servlets and JSP: Passing the Sun Certified Web Component Developer Exam Bryan Basham, Kathy Sierra, and Bert Bates, O'Reilly Media, 2nd Edition (2008)
2. Full Stack Java Development with Spring MVC, Hibernate, jQuery, and Bootstrap Mayur Ramgir,Kindle Edition,

Course Outcomes:
After completing the course student will be able to:

- **CO1:** Understand the fundamentals of server-side programming, including servlet lifecycle, session management, and servlet-to-servlet communication, to build foundational web applications. *(L2: Understand)*
- **CO2:** Apply Java Server Pages (JSP) features, including directives, implicit objects, expression language, and JSTL, to create dynamic and interactive web pages with error handling and session tracking. *(L3: Apply)*
- **CO3:** Implement JDBC to connect with databases, perform CRUD operations, and integrate it with servlets and JSP to build data-driven web applications. *(L3: Apply)*
- **CO4:** Evaluate Hibernate ORM features, including CRUD operations, HQL, and caching mechanisms, to optimize database interactions in web applications. *(L5: Evaluate)*
- **CO5:** Design and implement enterprise-level web applications using Spring Core, Spring MVC, and Spring Boot frameworks, integrating Hibernate and Spring Data JPA for transaction management and repository abstraction. *(L6: Design)*

CO-PO Mapping table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	3	3	2	1	0	2
CO2	3	3	3	3	2	1	0	2
CO3	3	3	3	3	2	1	0	3
CO4	3	3	3	3	2	1	0	3
CO5	3	3	3	3	2	1	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	

The topics included in this course are designed to get acquainted with one of the skills that handle necessary mathematical orientation, programming techniques and concept based learning.

Prepared by Mr.B.Satyanarayana

Verified by Dr.M.Srivenkatesh

SERVER-SIDE PROGRAMMING WITH PTHON

L	T	P	S	J	C
3	1	0	0	0	4

Course Description:

This course is designed to equip students with the skills necessary for full stack web development using Python, focusing on two of Python's most popular frameworks: Flask and Django. The course covers a broad spectrum of topics from basic Python programming for web applications to advanced features in Flask and Django, including RESTful API development, database integration, security implementations, and deployment strategies. Students will engage in hands-on projects that emphasize practical applications, enabling them to build and deploy dynamic web applications.

Course Objectives:

1. To master Python for Web Development
2. To develop Web Applications Using Flask
3. To implement Database Integration with SQLAlchemy
4. To explore Django's Capabilities
5. To deploy Django Applications

UNIT -I

Introduction to Python for Web Development

Python Basics: Quick overview of Python syntax, data types, control structures, and functions.

Setting up the Development Environment: Installing Python, virtual environments, and introduction to pip.

Introduction to Web Programming: Understanding client-server architecture, HTTP protocol basics. 9 Hours

UNIT-II

Flask Framework

Introduction to Flask: Setting up a Flask application, basic routing, and template rendering.

Flask Deep Dive: Handling forms, cookies, and sessions. Introduction to Flask blueprints for larger applications.

Database Integration: Connecting a Flask application to a database using SQLAlchemy ORM; performing CRUD operations. 9 Hours

UNIT -III

Advanced Flask Features

RESTful API Development: Building RESTful services with Flask-RESTful.

Middleware and Extensions: Utilizing Flask extensions like Flask-Mail for email handling, Flask-WTF for form validations.

Security Features: Implementing authentication (Flask-Login), and authorization, securing APIs.

Django Framework

UNIT –IV Hours

Introduction to Django: Setting up a Django project, understanding the MVT pattern.

Models and Databases: Deep dive into Django ORM, model relationships, migrations.

Handling HTTP Requests: Creating views, class-based views, form handling, and admin interface.

9 Hours

UNIT –V

Building and Deploying Django Applications

Advanced Django Features: Middleware, custom template tags and filters, signals and slots.

Integrating Frontend Technologies: Using static and media files, integrating with JavaScript and AJAX for dynamic content.

Deployment: Deployment strategies for Django applications, using services like Heroku, understanding Docker for deployment. 9 Hours

Text Books :

1. Flask Web Development: Developing Web Applications with Python by Miguel Grinberg Python Web Programming with Flask by Fabrizio Romano, O'Reilly Media,2nd Edition ,2018

2. Django for Beginners: Build websites with Python and Django by William S. Vincent, 3rd Edition (2020)

3. Two Scoops of Django 3.x: Best Practices for the Django Web Framework by Audrey Roy Greenfeld and Daniel Roy Greenfeld , Two Scoops Press,1st Edition ,2020

4. Django 3 By Example by Antonio Melé, Packt Publishing, 3rd Edition ,2020

Course Outcomes:

After completing the course student will be able to:

- **CO1:** Understand Python basics, web programming concepts, and the client-server architecture to set up a development environment for web applications. *(L2: Understand)*
- **CO2:** Apply Flask framework features, including routing, form handling, and database integration using SQL Alchemy ORM, to build web applications. *(L3: Apply)*
- **CO3:** Analyse advanced Flask functionalities, such as RESTful API development, middleware usage, and security features, to implement scalable and secure web services. *(L4: Analyse)*
- **CO4:** Evaluate Django framework concepts, including MVT patterns, ORM features, and class-based views, to manage complex web application workflows. *(L5: Evaluate)*
- **CO5:** Create and deploy full-stack Django applications, integrating frontend technologies, custom features, and deployment strategies like Docker and Heroku. *(L3: Design)*

CO-PO Mapping table

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1	3	3	2	3	2	0	0	3
CO2	3	3	3	3	2	0	0	3
CO3	3	3	3	3	2	0	0	2
CO4	3	3	3	3	2	0	0	3
CO5	3	3	3	3	2	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handlenecessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Dr.M.Srivenkatesh,
Mr.Shahid Shaik

Verified by Dr.M.Srivenkatesh

DESIGN AND ANALYSIS OF ALGORITHMS

L	T	P	S	J	C
3	0	0	0	0	3

Course Description:

This course is designed to provide students with a thorough understanding of algorithm design and analysis. It covers a wide range of algorithmic strategies including divide and conquer, greedy methods, dynamic programming, as well as basic traversal and advanced search techniques. The course emphasizes both the theoretical underpinnings of algorithms and their practical applications, using classical problems like the knapsack problem, job sequencing, and graph coloring to illustrate concepts. Students will also learn about NP-hard and NP-complete problems, equipping them with the tools to tackle complex computational problems.

Course Objectives:

- To understand Algorithmic Foundations
- To lean about Divide and Conquer Techniques
- To apply Greedy Methods
- To explore Dynamic Programming
- To implement Advanced Search Techniques

UNIT – I

Introduction To Algorithms: Algorithm Specification, Performance Analysis, Introduction To Randomized Algorithms. Divide And Conquer: The General Method, Binary Search, Finding Maximum & Minimum, Quick Sort, Selection, Strassen's Matrix Multiplication. 9 Hours

UNIT – II

Greedy Method: General Method, Knapsack Problem, Tree Vertex Splitting, Job Sequencing With Deadlines, Minimum Cost Spanning Trees, Single Source Shortest Paths. 9 Hours

UNIT – III

Dynamic Programming: The General Method, Multistage Graphs, All Pairs Shortest Paths, Optimal Binary Search Trees, String Editing, Reliability Design, the Traveling Sales Person Problem. 9 Hours

UNIT – IV

Basic Traversal And Search Techniques: Techniques For Graphs, Connected Components And Spanning Trees, Bi-Connected Components and DFs. **Back Tracking**: General Method, Eight Queens Problem, Sum Of Subsets, Graph Coloring, Hamiltonian cycles. 9 Hours

UNIT – V

Branch and Bound: The method, 0/1 knapsack problem, traveling salesperson problem algebraic problems: the general method, evaluation and interpolation. np hard and np complete problems: basic concepts. 9 Hours

Text Book:

1. Fundamentals of Computer Algorithms By Ellis Horowitz, SartajSahni, Sanguthevar

Rajasekaran, University Press, 2nd Edition, 2008.

Reference Books:

1. Fundamentals of AlgorithmicsBy G. Brassard AndP.Bratley, Phi, 2011.
2. Introduction to Algorithms By T.H. Cormen, C.E. Leiserson, R.L.Rivest, 3rd Edition, PHI,2010.
3. Introduction to Design Analysis of Algorithms By AnanyLevitin, 2nd Edition, Pearson Publications, 2009.

Course Outcomes:

After completing the course student will be able to:

- **CO1:** Understand algorithm specification, performance analysis, and the divide-and-conquer paradigm to solve problems like binary search, quick sort, and matrix multiplication. *(L2: Understand)*

- **CO2:** Apply the greedy method to solve optimization problems such as knapsack, job sequencing, minimum spanning trees, and shortest path problems. *(L3: Apply)*

- **CO3:** Analyse dynamic programming techniques for solving complex problems, including multistage graphs, all-pairs shortest paths, optimal binary search trees, and the traveling salesperson problem. *(L4: Analyse)*

- **CO4:** Evaluate graph traversal techniques, backtracking methods, and their applications to problems like graph coloring, Hamiltonian cycles, and the eight queens problem. *(L5: Evaluate)*

- **CO5:** Develop solutions for complex computational problems using branch-and-bound methods and assess NP-hard and NP-complete problems in the context of algorithm design. *(L3: Develop)*

CO-PO Mapping table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	1	-	-	2	-	3
CO2	3	3	2	-	-	2	-	3
CO3	3	3	3	-	-	2	1	3
CO4	3	3	3	-	-	2	-	3
CO5	3	3	3	-	-	1	-	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handlenecessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Dr.U.Santosh Kumar

Verified by Dr.M.Srivenkatesh

ARTIFICAL INTELLIGENCE AND MACHINE LEARNING

L	T	P	S	J	C
3	0	0	0	0	3

Course Description:

This course provides a comprehensive introduction to the field of Artificial Intelligence (AI) and its application in Machine Learning (ML). It covers a broad spectrum of topics, ranging from basic AI principles and search algorithms to advanced machine learning techniques and neural networks. Students will explore both the theoretical underpinnings and practical implementations of AI and ML, gaining insights into how these technologies can be leveraged to solve real-world problems.

Course Objectives

- To understand the Fundamentals of Artificial Intelligence:
- To explore Logic and Reasoning in AI
- To Learn Core Concepts of Machine Learning
- To develop Skills in Modelling and Evaluating AI Systems
- To gain Proficiency in Supervised and Unsupervised Learning Techniques

UNIT-I

Introduction to artificial intelligence: Introduction, foundations of AI, applications
Search Algorithms:
Uninformed –Dijkstra Algorithm, Depth Limited, Iterative deepening Search
Informed Search –Best First, A* and AO* Algorithms 9 Hours

UNIT-II

What is logic in AI, What is reasoning in AI, Types of reasoning in AI, Limitations of logic and reasoning in AI.

Introduction: What is machine learning, Types of Machine Learning- Supervised Machine Learning, Unsupervised Machine Leaning, Reinforcement Learning, Problems not solved with Machine Learning, Applications of Machine Learning

Preparing to Model: Machine Learning Activities, Basis types of Data in Machine Learning, Exploring the structure of the data, Data Quality and Remediation,
Data Preprocessing 9 Hours

UNIT-III

Basics of Future Engineering: Introduction ,Future Transformation ,Future Subset Selection
Supervised Learning1: Introduction, Examples of Supervised Learning, Classification Model, Classification Steps, Common Classification Algorithms KNN, Decision Tree, Random forest model, Support vector Machines. 9 Hours

UNIT IV

Supervised Learning2: Introduction, Examples of Regression, Common Regression Problems includes simple, Multi linear regression, Assumptions in Regression Analysis, Main Problems in Regression Analysis, Polynomial Regression, Logistic Regression

Modelling & Evaluation: Introduction, Selecting a model, Training a model,Model Representation and Interpretability, Evaluating Model Performance,Improving model performance. 9 Hours

UNIT V

Unsupervised Learning: Introduction, Unsupervised vs Supervised Learning, Application of Unsupervised Learning, Clustering, Clustering as a machinelearning task, Different types of clustering techniques, Partitioning methods,K-Means , K-Medoids: a representative object-based technique, Hierarchical Clustering.

9 Hours

Basic Neural Networks: Neural Network, Understanding Biological Neuron, Exploring Artificial Neuron, Types of activation function, Early implementation of ANN, Architecture of Neural Networks, Learning process in Artificial Neural Networks, Back Propagation, Deep Learning. 9 Hours

Text Books:

1. Artificial Intelligence A Modern Approach, Stuart Russell, Peter Norvig, Person

Edition, Fourth Edition.

2. Machine Learning by Subramanian, Chandra Mouli, Amit Kumar Das , Saikant Dutt, Pearson Publications, I edition, 2018.

3. Machine Learning by Tom Mitchell, McGraw Hill, 2007

Course Outcomes:

After completing the course student will be able to:

- **CO1:** Understand the concepts of logic and reasoning in AI, machine learning fundamentals, and data preprocessing techniques to prepare datasets for modeling. *(L2: Understand)*
- **CO2:** Apply feature engineering techniques and supervised learning algorithms, including KNN, decision trees, random forests, and SVM, to solve classification problems. (*L3: Apply)*
- **CO3:** Analyse regression techniques, including simple and multiple linear regression, polynomial regression, and logistic regression, to model and evaluate supervised learning problems. *(L4: Analyse)*
- **CO4:** Evaluate unsupervised learning methods, such as clustering, and explore partitioning and hierarchical clustering techniques to group data and uncover patterns. *(L5: Evaluate)*
- **CO5:** Develop and implement neural network models by understanding their architecture, activation functions, and learning processes, and apply backpropagation and deep learning concepts. *(L3: Develop)*

CO-PO Mapping table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	3	3	2	0	0	3
CO2	3	3	3	3	2	0	0	3
CO3	3	3	3	3	2	0	0	3
CO4	3	3	3	3	2	0	0	3
CO5	3	3	3	3	2	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handlenecessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Dr.M.Srivenkatesh,
Dr.U.Santosh Kumar

Verified by Dr.M.Srivenkatesh

BUNINESS INTELLIGENCE

L	T	P	S	J	C
3	0	0	0	0	3

Course Description:

This course provides an in-depth exploration of 99999 (BI), encompassing its development, methodologies, technologies, and applications in various industry sectors. The curriculum covers foundational BI concepts, data management, and the use of agent-based systems, advanced time series analysis, and the strategic implementation of BI in marketing models. Students will gain practical insights into how BI tools and technologies are applied to enhance decision-making processes, optimize performance, and drive organizational strategy.

Course Objectives:

- To understand the Core Concepts of BI
- To master Data Management for BI
- To explore BI Technologies and Tools
- To implement and Manage BI Projects
- To apply Advanced Techniques and Technologies

UNIT-I

Introduction to Business Intelligence

Overview: Definition and Scope of Business Intelligence**,** Historical Evolution of BI**,** Key Components of BI Systems**,** Role of BI in Organizational Strategy

Data Management for BI

Data Warehousing Concepts: Architecture, OLAP, OLTP, Data Mining: Methods and Techniques, Data Quality, Data Integration, and Data Governance

BI Technologies

Introduction to BI Software and Tools, Overview of Reporting, Dashboards, and Visualization Techniques, Basic Analytical Methods Used in BI

Implementation Challenges

BI Project Management: Planning and Execution, Common Pitfalls in BI Implementation and How to Avoid Them. 9 Hours

UNIT-II

Agent Systems in Business Intelligence

Agent-Based Modeling for BI: Introduction to Agent-Based Models in BI**,** Characteristics and Capabilities of Intelligent Agents**,** Types of Agents: Reactive, Proactive, and Hybrid

Agent Communication: Languages and Protocols for Agent Communication**,** Ontologies in Agent Systems: Role and Implementation**,** Inter-Agent Communication and Cooperation

Agent Systems Architecture: Architecture of Multi-Agent Systems (MAS)**,**Design and

Deployment of Agent Systems**,** Case Studies: Use of Agent Systems in BI Applications

Applications of Agent Systems in BI: Real-time Data Collection and Analysis**,** Automated

Decision-Making Processes**,** Enhancing User Experience and Interface with Intelligent Agents.

9 Hours

UNIT- III

Agent Technologies and Their Applications in Business Intelligence

Advanced Agent Technologies: Learning Agents and Adaptive Systems**,** Mobile Agents and Their Applications in BI**,** Security Aspects of Agent Technologies

Integrating AI with Agent Technologies: AI-Driven Agents for Predictive Analytics**,** Natural Language Processing for Enhanced Data Interaction**,**Machine Learning Models Deployed via Agents

Implementation and Management: Deployment Strategies for Agent-Based Systems Performance Evaluation and Scaling of Agent Systems, Maintaining and Updating Agent-Based Systems

Industry-Specific Applications
Financial Services: Risk Assessment and Fraud Detection, Healthcare: Patient Data Analysis and Management, Retail: Customer Behaviour Analysis and Personalization Strategies. 9 Hours

UNIT- IV

Time Series Data in Business Intelligence

Introduction to Time Series: Definition, Components of Time Series
Time Series Analysis: Moving Averages, Decomposition, Exponential Smoothing
Advanced Time Series Models: Autoregressive Models, Moving Average Models, ARMA and ARIMA Model, Forecasting with Time Series
Business Intelligence Applications: Marketing Models
Overview, Relational Marketing, Key Metrics and Strategies: Lifetime Value (LTV), Customer Acquisition and Retention, Cross-Selling and Up-Selling

Analytical Techniques: Market Basket Analysis, Web Mining
Case Studies. 9 Hours

UNIT- V

Business intelligence applications: Marketing models -Relational marketing, Motivations and objectives, An environment for relational marketing analysis, Lifetime value, The effect of latency in predictive models, Acquisition, Retention, Cross-selling and up-selling, Market basket analysis, Web mining, Business case studies, Retention in telecommunications, Acquisition in the automotive industry, Cross-selling in the retail industry. 9 Hours

Text Books :

1. Decision Support Systems for Business Intelligence by Vicki L. Sauter, Wiley Publisher, 2nd edition, 2010.
2. Business Intelligence by Rajiv Sabherwal, Irma Becerra-Fernandez, Wiley, 2010.
3. Business Intelligence: A Managerial Perspective on Analytics" by Ramesh Sharda, Dursun Delen, and Efraim Turban
4. Time Series Analysis and Its Applications: With R Examples" by Robert H. Shumway and David S. Stoffer

Course Outcomes:

After completing the course student will be able to:

- **CO1:** Understand the fundamental concepts of Business Intelligence (BI), including data management, BI technologies, and implementation challenges, to align BI strategies with organizational goals. (**L2: Understand**)
- **CO2:** Apply agent-based systems in BI to enhance real-time data collection, automated decision-making, and user experience through agent communication and multi-agent system architectures. (**L3: Apply**)
- **CO3:** Analyze advanced agent technologies and their integration with AI to improve predictive analytics, natural language processing, and machine learning applications in industry-specific BI scenarios. (**L4: Analyze**)
- **CO4:** Evaluate time series analysis techniques, such as ARIMA models and exponential smoothing, to forecast business trends and implement relational marketing strategies in BI. (**L5: Evaluate**)
- **CO5:** Design BI applications for industry-specific use cases, such as retention in telecommunications, acquisition in the automotive industry, and cross-selling in retail, to optimize marketing strategies. (**L6: Create**)

CO-PO Mapping table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	3	2	2	0	0	3
CO2	3	3	3	2	2	0	0	3
CO3	3	3	3	3	2	0	0	3
CO4	3	3	3	3	2	0	0	3
CO5	3	3	3	3	2	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handlenecessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Dr.M.Srivenkatesh
Mr.B.Srinivasa Rao

Verified by Dr.M.Srivenkatesh

AGILE SOFTWARE DEVELOPMENT

L	T	P	S	J	C
3	0	0	0	0	3

Course Description: Agile software development practices enable customer centric software development with collaborative teamwork centred around people. This course elaborates agile development principles and techniques covering the entire software development process from problem conception through development, testing and deployment to equip the learner with practical software development methodology

Course objectives:

- To understand the agile concept and its importance in software development.
- To acquire complete knowledge on Extreme programming.
- To know complete modelling of agile processes on the XP environment.
- To acquire knowledge on Scrum.
- To familiar with Feature driven development

UNIT -I

Introduction: The Agile manifesto, Agile methods, XP: Extreme Programming, DSDM, SCRUM,feature- Driven Development, Test Driven Development, modelling misconceptions, agile modelling,tools of misconceptions, updating agile models. 9 Hours

UNIT -II

Extreme Programming: Introduction, core XP values, the twelve XP practices, about extreme programming, planning XP projects, test first coding, making pair programming work. 9 Hours

UNIT -III

Agile Modelling and XP: Introduction, the fit, common practices, modelling specific practices, XP,objections to agile modelling, agile modelling and planning XP projects, XP implementation phase.

9 Hours

UNIT –IV

Scrum: Scrum Framework, Agile Principles, Sprints, Requirements and User Stories, Product backlogs, Estimation and Velocity, Roles, Planning, Multi-level Planning, Release Planning, Sprint planning. 9 Hours

UNIT -V

Feature-Driven Development: Introduction, incremental software development, Regaining Control,motivation behind FDD, planning an iterative project, architecture centric, FDD and XP. Test Driven Development: Unit Tests, Integration Tests, End-to-End Tests, Customer Tests. **Release Management:** Version Control, Continuous Integration. 9 Hours

Text Books:

John Hunt, Agile Software Construction,1st Edition, Springer,2005

Craig Larman, Agile and Iterative Development: A Manager's Guide, Addison-Wesley, Pearson Education – 2004.

Pearson, Robert C. Martin, Juli, James Shore, Chromatic 2013, The Art of Agile Development, O'Reilly Media.

Elisabeth Hendrickson, Agile Testing, Quality Tree Software Inc 2008.

Reference Books:

Andrew Stellman, Jenifer Greene, Headfirst Agile, O'Reilly, 2017

Peggy Gregory, Casper Lassenius, Xiaofeng Wang Philippe Kruchten (Eds.), Agile Processes in Software Engineering and Extreme Programming, 22nd International Conference on Agile Software Development, XP 2021 Virtual Event, June 14–18, 2021, Proceedings, Springer

Peggy Gregory, Philippe Kruchten (Eds.), Agile Processes in Software Engineering and Extreme Programming – Workshops XP 2021 Workshops Virtual Event, June 14–18, 2021 Revised Selected Papers, 2021

Ian Somerville, Software Engineering, 10th edition, Pearson, 2016

Course Outcomes:

After completing the course student will be able to:

CO1: Understand the Agile manifesto, methodologies, and practices such as Extreme Programming (XP), SCRUM, and Test-Driven Development (TDD) to foster agile software development. *(L2: Understand)*

CO2: Apply core Extreme Programming (XP) principles and practices, including pair programming and test-first coding, to plan and execute agile projects effectively. *(L3: Apply)*

CO3: Analyse the integration of Agile Modelling with XP, including common and modelling-specific practices, to address objections and improve project planning and implementation. *(L4: Analyse)*

CO4: Evaluate the SCRUM framework, including roles, sprints, and backlog management, to plan and execute agile projects efficiently. *(L5: Evaluate)*

CO5: Design iterative and incremental software development projects using Feature-Driven Development (FDD) and implement Test-Driven Development (TDD) with proper release management techniques. *(L3: Design)*

CO-PO Mapping table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	2	3	3	3	3	0	0	3
CO2	2	3	3	3	3	0	0	3
CO3	2	3	3	3	3	0	0	3
CO4	2	3	3	3	3	0	0	3
CO5	2	3	3	3	3	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handlenecessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Dr.T.Uma Devi Verified by Dr.M.Srivenkatesh

CLOUD COMPUTING: CONCEPTS AND TECHNOLOGIES

L	T	P	S	J	C
3	0	0	0	0	3

Course Description:

This course provides a comprehensive overview of cloud computing, a transformative computing paradigm that has reshaped how organizations deploy, manage, and scale their IT resources. It delves into various forms of computing such as grid, cluster, and distributed computing, and focuses extensively on cloud computing technologies. Students will explore the evolution, architecture, and service models of cloud computing, including Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS). The course also addresses key issues such as virtualization technologies, service management, cloud security, and the economic impacts of cloud adoption.

Course Objectives:

- Understand the Fundamentals of Cloud Computing
- Explore Service Models and Deployment Models
- Master Virtualization Technology
- Examine Cloud Service Management and Economics
- Study Cloud Security and Compliance

UNIT -I

Introduction , Overview of Computing Paradigm - Recent trends in Computing - Grid Computing, Cluster Computing, Distributed Computing, Utility Computing, Cloud Computing - Evolution of cloud computing - Cloud Computing (NIST Model) – Characteristics

Pros and Cons of Cloud Computing, Cloud computing vs. Cluster computing vs. Grid computing – Role of Open Standards - Cloud Computing Architecture - Cloud computing stack - Service Models (XaaS) - Deployment Models. 9 Hours

UNIT -II

Infrastructure as aService(IaaS) Infrastructure as a Service(IaaS) – Introduction- IaaS definition, virtualization, Different approaches to virtualization, Hypervisors, Machine Image, Virtual Machine(VM) - Resource Virtualization – Server, Storage, Network, Virtual Machine(resource) provisioning and manageability, storage as a service, Data storage in cloud computing (storage as a service) - Examples - Amazon EC2 - Renting, EC2 Compute Unit, Platform and Storage,pricing, customers – Eucalyptus. 9 Hours

UNIT -III

Platform asa Service(PaaS) Platform as a Service(PaaS) - Introduction - What is PaaS, Service Oriented Architecture (SOA) - Cloud Platform and Management – Computation, Storage – Examples – Google App Engine, Microsoft Azure, SalesForce.com, Force.complatform-

Software as a Service(PaaS)-IntroductiontoSaaS-Webservices-Web2.0- WebOS-Case Study onSaaS. 9 Hours

UNIT –IV

Service Management in Cloud Computing (9hrs) Service Management in Cloud Computing – Service Level Agreements (SLAs) - Billing & Accounting - Comparing

Scaling Hardware: Traditional vs. Cloud-Economics of scaling: Benefitting enormously-Managing Data-Looking at Data, Scalability& Cloud Services, Database& Data Stores in Cloud, Large Scale Data Processing. 9 Hours

UNIT –V

Cloud Security Cloud Security - Infrastructure Security - Network level security, Host level security, Application level security – Data security and Storage Data privacy and security Issues,

Jurisdictional issues raised by Data location-Identity &Access Management-Access Control-Trust, Reputation,Risk-Authentication in cloud computing, Client access in cloud, Cloud contracting Model,
Commercial and business considerations. 9 Hours

Text Books:

1. Rajkumar Buyya, James Broberg, Andrzej M. Goscinski, Cloud Computing: Principles and Paradigms, Wiley, 2011.

2. RonaldL.Krutz, Russell DeanVines, Cloud Security: A Comprehensive Guide to Secure Cloud Computing, WileyIndia,2010.

Reference Books:

1. Rajkumar Buyya, Christian Vecchiola, S. Thamarai Selvi, Mastering Cloud Computing, Mc Graw Hill Education,2013.

2. NikosAntonopoulos, LeeGillam, Cloud Computing: Principles, Systems and Applications, Springer, 2012. 3. Barrie Sosinsky,Cloud Computing Bible, Wiley-India,2010.

Course Outcomes:

After completing the course student will be able to:

CO1: Understand the evolution, architecture, and characteristics of cloud computing, including its comparison with other computing paradigms, to identify its benefits and challenges. *(L2: Understand)*

1. **CO2:** Apply concepts of Infrastructure as a Service (IaaS), including virtualization, resource provisioning, and data storage, to explore and utilize cloud-based infrastructures like Amazon EC2 and Eucalyptus. *(L3: Apply)*

CO3: Analyse the features of Platform as a Service (PaaS) and Software as a Service (SaaS), including their role in enabling cloud-based applications, to understand their use cases and management strategies. *(L4: Analyse)*

CO4: Evaluate service management practices in cloud computing, such as SLAs, billing, and scalability, to manage and optimize cloud services for large-scale data processing. *(L5: Evaluate)*

CO5: Design secure cloud computing environments by implementing security measures at the infrastructure, network, and application levels, addressing data privacy, identity management, and access control issues. *(L3: Design)*

CO-PO Mapping table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	3	3	2	0	0	3
CO2	3	3	3	3	2	0	0	3
CO3	3	3	3	3	2	0	0	3
CO4	3	3	3	3	3	0	0	3
CO5	3	3	3	3	3	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handlenecessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Dr.G.Babu Rao Verified by Dr.M.Srivenkatesh

COMPUTER NETWORKS

L	T	P	S	J	C
3	0	0	0	0	3

Course Description:

This course provides a comprehensive introduction to the principles and practical aspects of data communications, network architectures, protocols, and standards, with a specific focus on the Internet and related technologies. It covers the foundational theory behind data communication systems and network protocols, as well as the implementation and operational aspects of network layers, transmission methods, and advanced networking concepts. Students will explore both the theoretical underpinnings and the practical applications of building, maintaining, and securing networks.

Course Objectives

- To understand Network Fundamentals and Protocols
- To explore Network Media and Switching Technologies
- To master Data Link Layer Operations and Ethernet Technologies
- To understand Advanced Networking Concepts
- To apply Knowledge to Transport and Application Layers

UNIT - I

Introduction: Data Communications, Networks, The Internet, Protocols and Standards.
Network Models: Layered Tasks, The OSI Model, Layers in the OSI Model, TCP/IP Protocol Suite, Addressing.
Digital Transmission: Transmission Modes. 9 Hours

UNIT - II

Transmission Media: Guided Media, Unguided Media-Wireless Transmission.
Switching: Circuit Switched Networks, Datagram Networks, Virtual, Circuit Networks, Structure of a Switch.
Data Link Layer: Error Detection and Correction- Introduction, Block Coding, Linear Block Codes, Cyclic Codes, Checksum. 9 Hours

UNIT - III

Data Link Control: Framing, Flow and Error Control, Protocols, Noiseless Channels, Noisy Channels, HDLC, Point-to-Point Protocol.
Wired LANs: Ethernet-IEEE Standards, Standard Ethernet, Changes in the Standard, Fast Ethernet, Gigabit Ethernet.
Connecting LANs, Backbone Networks and Virtual LANs: Connecting Devices, Backbone Networks, Virtual LANs. 9 Hours

UNIT - IV

Network Layer: Logical Addressing - IPv4 Addresses, IPv6 Addresses.
Internet Protocol: IPv4, IPv6.
Network Layer: Address Mapping, ICMP, IGMP, ICMPv6.
Delivery, Forwarding and Routing: Delivery, Forwarding, Unicast Routing Protocols, Multicast Routing Protocols.

UNIT - V

Transport Layer: Process to Process Delivery- UDP, TCP and SCTP.
Application Layer: Domain Name System-Namespace, Domain Name Space, Distribution of Name Space, DNS in the INTERNET, RESOLUTION, DNS Messages, Types of Records, Registrars, Dynamic Domain Name System(DDNS), Encapsulation. 9 Hours

Text Books: 1. Data Communications and Networking by Behrouz A Forouzan, TMH, 4th edition, 2007.

Reference Books:
1. Data and Computer Communications by William Stallings, Pearson Publications, 9th edition, 2011.
2. Computer Networks by Andrew S. Tanenbaum, Prentice Hall, 5 th edition, 2012.

Course Outcomes:

After completing the course student will be able to:

CO1: Understand the fundamentals of data communication, networking models, and digital transmission to explain the layered architecture and key standards in networking. *(L2: Understand)*

CO2: Apply knowledge of transmission media and switching techniques to evaluate error detection and correction methods for reliable data communication. *(L3: Apply)*

CO3: Analyze data link control protocols, Ethernet standards, and techniques for connecting LANs to design efficient and robust local area networks. *(L4: Analyse)*

CO4: Evaluate logical addressing schemes, Internet protocols (IPv4 and IPv6), and routing techniques to ensure efficient delivery and forwarding of data packets. *(L5: Evaluate)*

CO5: Develop and implement transport and application layer protocols, including DNS and DDNS, to enable seamless communication and resolution in distributed systems. *(L3: Develop)*

CO-PO Mapping table

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1	3	3	3	3	2	0	0	2
CO2	3	3	3	3	2	0	0	2
CO3	3	3	3	3	2	0	0	2
CO4	3	3	3	3	2	0	0	3
CO5	3	3	3	3	2	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handle necessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Mr.M.Suresh Kumar

Verified by Dr.M.Srivenkatesh

DATA MINING: CONCEPTS AND TECHNIQUES

L	T	P	S	J	C
3	0	0	0	0	3

Course Description:

This course offers a comprehensive introduction to Data Mining, a crucial component of modern data analysis used to discover patterns and relationships in large datasets. It covers a broad range of topics from the basics of data mining, understanding and preparing data, to advanced techniques in pattern recognition, classification, clustering, and outlier detection. Students will learn about the various methods and technologies used in data mining, including how data warehousing and Online Analytical Processing (OLAP) contribute to mining processes.

Course Objectives:

- To understand the Foundations of Data Mining
- To develop Skills in Data Analysis and Pre-processing
- To learn to Utilize Data Warehousing and OLAP for Data Mining
- To acquire Techniques for Mining Patterns and Building Models
- To analyse Cluster Analysis and Outlier Detection

UNIT – I

Introduction: What is Data Mining? Kind of data on which mining is done, Kinds of patterns can bemined, and Technologies used, Kinds of Applications targeted, Major Issues of Data mining.

Getting to Know Your Data: Data Objects and Attribute, Measuring Data Similarity and Dissimilarity. 9 Hours

UNIT - II

Data Pre-processing: An Overview, Data Integration, Data Reduction, Data Transformation and Data Discretization.

Data Warehouse and OLAP Technology for Data Mining: Data Warehouse basic concepts, Data warehouse modeling. 9 Hours

UNIT – III

Mining Frequent patterns: Associations and Correlations: Basic Concepts, Frequent Item set MiningMethods, Interesting Patterns and Pattern evaluation methods. 9 Hours

UNIT - IV

Classification: Basic Concepts, Decision Tree Induction, Bayesian Classification, Rule based Classification, Model Evaluation & Selection, Techniques to improve classification Accuracy. 9 Hours

UNIT - V

Cluster Analysis: Requirements, Overview of various clustering methods , Densitybased Methods, Grid-based Methods, and Evaluation of Clustering.

Outlier Detection: Types of Outliers & Challenges in outlier detection, Outlier detection methods, Statistical Approaches, Proximity based approaches, Cluster & Classification

based approach. 9 Hours

Text Books:

1. Data Mining Concepts and Techniques by Jiawei Han, Michelin Kamber, and Jian Pei,Elsevier Publications,3rd Edition, 2013.

Reference Books:

1. Introduction to Data Mining by Pang- Ning Tan, Michael Steinbach and Vipin Kumar,Pearson Education, Low Price Edition.
2. Insight to Data Mining Theory and Practice by K.P.Soman, ShyamDiwakar and V. Ajay, Prentice Hall of India, 2006.

Course Outcomes:

After completing the course student will be able to:

- **CO1:** Understand the fundamental concepts of data mining, including data types, patterns, and similarity measures, to explore and analyze datasets effectively. *(L2: Understand)*

- **CO2:** Apply data preprocessing techniques, including data integration, reduction, transformation, and discretization, and utilize data warehousing concepts for data organization and preparation. *(L3: Apply)*

- **CO3:** Analyse frequent pattern mining techniques, such as association and correlation methods, to discover interesting patterns in datasets. *(L4: Analyse)*

- CO**4:** Evaluate classification techniques, including decision trees, Bayesian methods, and rule-based systems, to build accurate and efficient predictive models. (**L5: Evaluate**)

- **CO5:** Develop clustering models and implement outlier detection methods using statistical, proximity-based, and classification-based approaches to uncover hidden structures and anomalies in data. (**L3: Develop**)

CO-PO Mapping table

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1	3	3	3	3	2	0	0	3
CO2	3	2	3	3	2	0	0	3
CO3	3	3	3	3	2	0	0	3
CO4	3	3	3	3	2	0	0	3
CO5	3	3	3	3	3	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handlenecessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Dr.M.Seshashayee

Verified by Dr.M.Srivenkatesh

INTRODUTION TO PARELLEL COMPUTING

L	T	P	S	J	C
3	0	0	0	0	3

Course Description:

This course offers a comprehensive exploration into the concepts, techniques, and technologies of parallel programming. It is designed to equip students with the knowledge and skills required to develop efficient parallel algorithms and applications using various parallel programming platforms. The curriculum covers a broad spectrum of topics, including the architectural considerations of parallel systems, principles of parallel algorithm design, performance analysis, the message-passing programming paradigm, and specific parallel algorithms for matrix operations and sorting.

Course Objectives:

- To understand Parallel Computing Platforms
- To learn Parallel Algorithm Design
- To analyse Parallel Program Performance
- To Utilize the Message-Passing Paradigm
- To implement Parallel Algorithms and Applications

UNIT -I

Introduction : Introduction–Motivation–Scope-Parallel Programming Platforms: Implicit Parallelism–Limitations of Memory System Performance - Dichotomy of Parallel Computing Platforms – Communication cost in Parallel Machines – Routing Mechanism for Interconnection Networks.

9 Hours

UNIT - II

Principles of Parallel Algorithm , Decomposition Techniques – Characteristics of Tasks and Interactions – Mapping Techniques for Load Balancing – Methods for containing Interaction overheads – Parallel Algorithm Models – Basic communication Operations: One-to-all Broadcast
and All-to-one Reduction – Scatter and Gather – Improving the Speed of some communication Operations.

9 Hours

UNIT - III

Analytical Modelling of Parallel programs ,Sources of Overhead in Parallel Programs – Performance metrics for parallel systems – effect of granularity and data mapping on performance – scalability of parallel systems – Minimum analysis of parallel Programs – other Scalability Metrics.

9 Hours

UNIT - IV

Programming using Message Passing Paradigm, Principles of Message-Passing Programming – The Building Blocks: Send and Receive Operations – MPI: The Message Passing Interface – Topologies and Embedding -Overlapping Communication with Computation - Collective Communication and Computation Operations - Groups and Communicators. 9 Hours

UNIT - V

Parallel Algorithms and Applications, Dense Matrix Algorithms: Matrix-Vector Multiplication-Matrix- Matrix Multiplication-Solving a System of Linear Equations. Sorting: Issues in Sorting on Parallel Computers - Sorting Networks - Bubble Sort and its Variants – Quick sort - Bucket and Sample Sort - Other Sorting Algorithms. 9 Hours

Text Book(s): 1. Ananth Grama, Anshul Gupta, George Karypis, Vipin Kumar, Introduction to Parallel Computing, , Pearson, Second Edition,2003.

Reference Book(s):
1. Peter S. Pacheco, An introduction to Parallel Programming, Morgan Kaufmann, First Edition,2011. 2. Fayez Gebali, Algorithms and Parallel Computing, Wiley series,2011.

Course Outcomes:

After completing the course student will be able to:

- **CO1:** Understand the fundamental concepts of parallel programming platforms, including communication costs and interconnection networks, to identify the challenges and opportunities in parallel computing. (**L2: Understand**)
- **CO2:** Apply principles of parallel algorithm design, decomposition techniques, and load-balancing strategies to optimize parallel computation tasks. (**L3: Apply**)
- **CO3:** Analyse performance metrics, scalability factors, and sources of overhead in parallel programs to evaluate the efficiency of parallel computing systems. (**L4: Analyse**)
- **CO4:** Evaluate message-passing programming principles and MPI operations, including collective communication and topologies, to design efficient parallel programs. (**L5: Evaluate**)
- **CO5:** Design parallel algorithms for dense matrix computations, sorting techniques, and solving linear equations to solve computational problems on parallel systems. (**L3: Design**)

CO-PO Mapping table

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1	3	3	3	3	2	0	0	3
CO2	3	3	3	2	2	0	0	3
CO3	3	3	3	2	2	0	0	3
CO4	3	3	3	2	2	0	0	3
CO5	3	3	3	2	2	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handlenecessary mathematical orientation, programming techniques and concept based learning.	

CRYPTOGRAPHY AND NETWORK SECURITY

L	T	P	S	J	C
3	0	0	0	0	3

Course Description:

This comprehensive course in Network and Computer Security provides a detailed examination of the fundamental principles, techniques, and technologies used to secure computer systems and networks. It delves into the architectures, protocols, and standards that ensure the confidentiality, integrity, and availability of information systems. Covering a broad spectrum from symmetric and asymmetric cryptography to securing network communications and managing security threats, this course equips students with the knowledge and skills necessary to design, implement, and manage secure computer systems.

Course Objectives:

- To understand Fundamental Concepts of Computer Security
- To analyze Cryptographic Techniques and Protocols
- To **e**xplore Security at Different Network Layers
- To **i**mplement Security in Wireless Networks and Electronic Communications
- To identify and Mitigate Threats from Malicious Software

UNIT- I

Introduction: Computer Security Concepts, The OSI Security Architecture, Security Attacks, Security Services, Security Mechanisms, A Model for Network Security, Standards. Symmetric Encryption and Message Confidentiality: Symmetric Encryption Principles, Symmetric Block Encryption Algorithms, DES, AES, Stream. 9 Hours

UNIT – II

Public-Key Cryptography and Message Authentication: Approaches to Message Authentication, Secure Hash Functions, Message Authentication Codes, Public Key Cryptography Principle, PublicKey Cryptography Algorithms, RSA, Diffie Hellman key Exchange, Digital Signatures. Key Distribution and User Authentication: Symmetric Key Distribution Using Symmetric Encryption, Kerberos, Key Distribution Using Asymmetric Encryption. 9 Hours

UNIT - III

Transport-Level Security: Web Security Issues, Secure Sockets Layer (SSL), Transport Layer Security (TLS), HTTPS, Secure Shell (SSH).

Wireless Network Security: Wireless Security, Mobile Device Security, IEEE 802.11 Wireless LAN Overview, IEEE 802.11i Wireless LAN Security.Understand various security protocols and mobile security issues.(L2). 9 Hours

UNIT –IV

Electronic Mail Security: Pretty Good Privacy (PGP), S/MIME.IP Security: IP Security Overview, IP Security Policy, Encapsulating Security Payload, Combining Security Associations, Internet Key Exchange and Cryptographic Suites. 9 Hours

UNIT - V

Malicious Software: Types of Malicious Software, Propagation, Infected Content, Viruses Propagation, Vulnerability Exploit, Worms, Propagation, Social Engineering, SPAM, Trojans, Payload – System Corruption, Payload, Attack Agent, Zombie, Bots, Payload, Information Theft, Keyloggers, Phishing, Spyware, Payload, Stealthing, Backdoors, Rootkits Distributed Denial of Service Attacks. 9 Hours

Text Books:

1. Network Security Essentials, Applications and Standards, 5th edition, William Stallings, Pearson Education, 2013.

Reference Books:

1. Cryptography & Network Security, Behrouz A. Forouzan, Tata McGraw-Hill, New Delhi, 2007.
2. Network Security: Private Communication in a Public World, Kaufman, Pearson Education Asia,New Delhi, 2002.
3. Cryptography and Network Security: AtulKahate, Tata McGraw-Hill Publishing Company Limited, New Delhi, 2003.

Course Outcomes:

After completing the course student will be able to:

CO1: Understand the fundamental concepts of computer security, including the OSI security architecture, symmetric encryption, and algorithms like DES and AES, to explain secure communication principles. *(L2: Understand)*

CO2: Apply principles of public-key cryptography, message authentication, and key distribution techniques, including RSA and Kerberos, to secure communications and verify authenticity. *(L3: Apply)*

CO3: Analyse transport-level security mechanisms, such as SSL, TLS, and SSH, as well as wireless security standards like IEEE 802.11i, to address mobile and wireless network vulnerabilities. *(L4: Analyse)*

CO4: Evaluate email security protocols, including PGP and S/MIME, and IP security measures, such as IPsec and Internet Key Exchange, to protect sensitive information. *(L5: Evaluate)*

CO5: Develop strategies to identify, mitigate, and defend against malicious software attacks, including viruses, worms, Trojans, and DDoS attacks, to enhance system security. *(L3: Develop)*

CO-PO Mapping table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	3	2	2	0	0	3
CO2	3	3	3	3	2	0	0	3
CO3	3	3	3	3	2	0	0	3
CO4	3	3	3	3	2	0	0	3
CO5	3	3	3	2	2	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handle necessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Dr.V.Nagalakshmi

Verified by Dr.M.Srivenkatesh

DIGITAL FORENSICS

L	T	P	S	J	C
3	0	0	0	0	3

Course Description:

This course offers an in-depth exploration into the realm of wireless and mobile technologies, focusing on the security challenges and vulnerabilities inherent in these systems, as well as the methodologies employed in digital forensics to address criminal activities involving these technologies. Students will gain insights into various wireless platforms from personal area networks to wide area networks, and delve into mobile system security frameworks, mobile forensics, and the broader field of digital forensics.

Course Objectives:

- To understand Wireless and Mobile Security Challenges
- To explore Security Frameworks for Mobile Systems
- To conduct Mobile Forensics Investigations
- To learn Principles of Digital Forensics
- To Analyse and Apply Digital Forensic Techniques.

UNIT-I

Introduction to Wireless Technologies: Overview of wireless technologies and security: Personal Area Networks, Wireless Local Area Networks, Metropolitan Area Networks, Wide Area Networks. Wireless threats, vulnerabilities and security: Wireless LANs, War Driving, War Chalking, War Flying, Common Wi-Fi security recommendations, PDA Security, Cell Phones and Security, Wireless DoS attacks, GPS Jamming, Identity theft. 9 Hours

UNIT-II

Security Framework for Mobile Systems :CIA triad in mobile Phones-Voice, SMS and Identification data interception in GSM: Introduction, practical setup and tools, implementation-Software and Hardware Mobile phone tricks: Netmonitor, GSM network service codes, mobile phone codes, catalog tricks and AT command set- SMS security issues. 9 Hours

UNIT-III

Mobile Phone Forensics : Crime and mobile phones, evidences, forensic procedures, files present in SIM card, device data, external memory dump, evidences in memory card, operators systems- Android forensics: Procedures for handling an android device, imaging android USB mass storage devices, logical and physical techniques. 9 Hours

UNIT -IV

Introduction to Digital Forensics : Digital forensics: Introduction – Evidential potential of digital devices: closed vs. open systems, evaluating digital evidence potential- Device handling: seizure issues, device identification, networked devices and contamination. 9 Hours

UNIT-V

Analysis of Digital Forensic Techniques: Digital forensics examination principles: Previewing, imaging, continuity, hashing and evidence locations- Seven element security model- developmental model of digital systems audit and logs- Evidence interpretation: Data content and context. 9 Hours

Text Books :
1. Iosif I. Androulidakis, "Mobile phone security and forensics: A practical approach", Springer publications, 2012.
2. Andrew Hoog, "Android Forensics: Investigation, Analysis and Mobile Security for Google Android", Elsevier publications, 2011.
3. Angus M. Marshall, "Digital forensics: Digital evidence in criminal investigation", John – Wiley and Sons, 2008
4. Gregory Kipper, "Wireless Crime and Forensic Investigation", Auerbach Publications.

Course Outcomes:

After completing the course student will be able to:

- **CO1:** Understand the fundamentals of wireless technologies, threats, vulnerabilities, and security recommendations to safeguard wireless communication systems. *(L2: Understand)*
- **CO2:** Apply the principles of mobile system security, including the CIA triad, GSM interception, and SMS security, to identify and mitigate mobile device vulnerabilities. *(L3: Apply)*
- **CO3:** Analyse mobile phone forensic procedures, including SIM card and memory data extraction, to collect and process digital evidence for forensic investigations. *(L4: Analyse)*
- **CO4:** Evaluate digital forensic procedures, including device seizure, identification, and contamination prevention, to ensure proper handling and assessment of digital evidence. *(L5: Evaluate)*
- **CO5:** Create a structured approach for analyzing digital forensic techniques, including imaging, hashing, and evidence interpretation, to enhance the reliability and accuracy of forensic investigations. *(L6: Develop)*

CO-PO Mapping table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	3	2	2	0	0	3
CO2	3	3	3	3	2	0	0	3
CO3	3	3	3	2	2	0	0	3
CO4	3	3	3	2	2	0	0	3
CO5	3	3	3	2	2	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	

The topics included in this course are designed to get acquainted with one of the skills that handle necessary mathematical orientation, programming techniques and concept based learning.

Prepared by Dr.V.Nagalakshmi
Dr.M.Srivenkatesh

Verified by Dr.M.Srivenkatesh

DIGITAL MARKETING

L	T	P	S	J	C
3	0	0	0	0	3

Course Description:

This course provides a comprehensive overview of digital marketing, exploring the various components, strategies, and tools essential for success in the digital landscape. Students will learn about the differences between digital and traditional marketing, the benefits of digital marketing, and current trends that shape the industry. The curriculum covers key digital marketing channels and techniques, including SEO (Search Engine Optimization), SEM (Search Engine Marketing), social media marketing, content marketing, email marketing, and mobile marketing.

Course Objectives:

- To understand the Fundamentals of Digital Marketing
- To develop Digital Marketing Strategies
- To master SEO and SEM Techniques
- To leverage Digital Marketing Channels
- To analyse and Optimize Digital Marketing Performance

UNIT-I

Understanding Digital Marketing: Components of Digital Marketing, Need and Scope of Digital Marketing, Benefits of Digital Marketing, Digital Marketing Platforms and Strategies, Comparison of traditional marketing and Digital marketing, Digital Marketing Trends.

Digital Marketing Plan: Elements of a Digital Marketing Plan, Marketing Plan, Executive Summary, Mission, Situational Analysis, Opportunities and Issues, Goals and Objectives, Marketing Strategy, Action Plan, Budget, Writing the Marketing Plan and Implementing the Plan. Keyword analysis. 9 Hours

UNIT-II

Marketing in the Digital Era: Keyword Planner in Google Search Console, Analysis of keyword dataset, Sample case study on Keyword dataset, Importance of Audience Segmentation, segments usage in Digital Media, Organizational Characteristics, Purchasing Characteristics, Acquisition and Retention of new customers.

Channels of Digital Marketing: Website Planning and creation tools, Search Engine Optimization, Online Advertising, PPC advertising, Google AdWords, Email Marketing, Blog Marketing, Social Media Marketing, Audio, Video and Interactive Marketing, Mobile Marketing, Migrating from Traditional Channels to Digital Channels. 9 Hours

UNIT-III

Search Engine Optimization : Introduction to On page SEO, SEO factors and tags that influence website performance, Title tag, Description tag, Header tag, Alt tag, URL structure, Robots.txt file, Sitemap, Canonicalization, Bread Crumb Structure, 301 Redirection, Custom 404 Error, Web Content Optimization, Text to Html Ratio, Website Loading Speed factors

Off page Optimization : Introduction to Off page SEO, Checking Domain Authority of website, Factors influence the high authority sites, Directory Submissions, Social Bookmarking, Forum Submission, Business Profile Creation, Article Submission, Creating backlinks, Internal, External links and Footer links, Competitor back link and data analysis. 9 Hours

UNIT- IV

Online Advertising and analysis using Web masters: Google Web Masters & Audit tools, PPC Advertising & Google AdWords, Google Ad Metrics and Performance, Payment Methods of Online Advertising – CPM (Cost-per-Thousand) and CPC (Cost-per-click), Display Ads - choosing a Display Ad Format, Landing Page and its importance, Sample Case study on online advertisement.

Google MyBusiness & Monitoring, Google Algorithms, E-Mail Marketing, SMS Marketing, Lead Generation for Business, Affiliate Marketing, Making Money through Google Adsense

9 Hours

UNIT-V

Social Media Marketing and other marketing: Influence of Social Media, Social Media Marketing, Google Analytics, Analyzing Digital Media Performance, Analyzing Website Performance, Analyzing Advertising Performance, Networking with Facebook, LinkedIn, Blogging as a social medium, Microblogging with Twitter, Social Sharing with YouTube, Social Media for Customer Reach, Acquisition and Retention. 9 Hours

Text Books:

1. B2B Digital Marketing by Michael Miller, 1e, Pearson, 2014.
2. Digital marketing by Vandana Ahuja,Oxford University Press 2015
3. Social Media Marketing by Michael R Solomon, Tracy Tuten, Pearson, 1e, 2015.
4. E-Marketing by Judy Strauss & Raymond Frost, Pearson, 2016
5. Online marketing – A customer led approach by Richard Gay, Alan Charles worth, Rita Esen,

Course Outcomes:

After completing the course student will be able to:

- **CO1:** Understand the components, strategies, and trends of digital marketing, including the creation and implementation of a comprehensive digital marketing plan. (**L2: Understand**)
- **CO2:** Apply keyword analysis, audience segmentation, and various digital marketing channels like SEO, PPC advertising, and social media to attract and retain customers. (**L3: Apply**)
- **CO3:** Analyze on-page and off-page SEO techniques, such as optimizing website performance, backlink creation, and competitor analysis, to improve search engine rankings. (**L4: Analyze**)
- **CO4:** Evaluate online advertising strategies, including Google Ad Metrics, payment methods, and landing page optimization, to design and implement effective campaigns. (**L5: Evaluate**)
- **CO5:** Develop social media marketing strategies and analyze digital media performance using tools

like Google Analytics, blogging, and social sharing platforms to enhance customer engagement and retention. *(L3: Develop)*

CO-PO Mapping table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	2	3	3	2	2	0	0	3
CO2	2	3	3	2	2	0	0	3
CO3	2	3	3	2	2	0	0	3
CO4	2	3	3	2	2	0	0	3
CO5	2	3	3	2	2	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handle necessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Dr.G,Ramash Naidu Verified by Dr.M.Srivenkatesh

E-COMMERCE STRATEGIES AND TECHNOLOGIES

L	T	P	S	J	C
3	0	0	0	0	3

Course Description:

This course provides an in-depth exploration of electronic commerce (e-commerce) and its profound impact on business practices and society. It covers the fundamental concepts, technologies, and business models that underpin e-commerce. Students will learn about the scope, benefits, and limitations of e-commerce, various business models, and the role of the Internet in transforming commerce. The course also delves into e-marketing, e-payment systems, e-finance, legal frameworks related to online business, and security measures necessary to protect online transactions.

Course Objectives:

- To understand the Foundations of E-Commerce.
- To explore E-Commerce Business Models: Master
- To e-Marketing and Online Strategies
- To navigate E-Payment Systems and E-Finance
- To address Legal and Security Issues in E-Commerce

UNIT -I

E – Commerce: Meaning, definition, features, functions of E-Commerce, Scope, Benefits and limitations of E-Commerce – The Internet and India – E-commerce opportunities and challenges for Industries.

Business Models for E-commerce: The Birh of Portals – E-Business Models – Business-to Consumer (B2C) – Business-to-Business (B2B) – Consumer-to Consumer (C2C) – Consumer to-Business (C2B) – Brokerage Model – Value Chain Model – Advertising Model. 9 Hours

UNIT –II

E-marketing – Traditional Marketing Vs.E-Marketing – Impact of E-commerce on markets – Marketing issues in E-Marketing – Online Marketing – E-advertising – Internet Marketing Trends – E-Branding – Marketing Strategies. 9 Hours

E-payment Systems: Digital payment Requirements – Digital Token-based E-payment systems – Benefits to Buyers – Benefits to Sellers – Credit card as E-payment system – Mobile payments – smart card cash payment system – Micropayment system – E- Cash.

UNIT – III

E-Finance: Areas of Financing, E-Banking - Traditional Banking Vs. E-Banking – Operations in E-Banking – E-Trading – Stock Market trading – Importance and advantages of E-Trading.
Legal Framework for E-Commerce: E-Commerce Legal Framework – Rights and Obligations in the World of E-commerce – Copyrights – Defamation – Privacy – Contracts – Taxation – Signing a contract Electronically – Domain name and Registration. 9 Hours

UNIT –IV

E-Security: Security for E-commerce – Security Design – Analysing risk – E-Banks and Security – safety of E-Commerce – Online Shopping with confidence – Firewalls for system Integrity – Virus Protection and Protection from intruders. 9 Hours

UNIT –V

Mobile Commerce: Challenges of E-commerce – Global Mobile E-Commerce – Secure Mobile Commerce – Secured Payments through Mobile – First Mobile Commerce Service. 9 Hours

TEXT BOOK: Joseph P. T., E - Commerce – An Indian Perspective
REFERENCE: Jaiswal S., E-Commerce
Mohammad Mahmoudi Maymand, E-Commerce
Murthy C.S.V., E-Commerce - Concepts, Models and Strategies

Course Outcomes:

After completing the course student will be able to:

- **CO1:** Understand the features, functions, and business models of E-commerce, including B2C, B2B, C2C, and C2B, to identify opportunities and challenges in the digital market. (**L2: Understand**)
- **CO2:** Apply E-marketing strategies, including online advertising and branding, and evaluate digital payment systems like credit cards, mobile payments, and e-cash to enhance customer engagement and transaction efficiency. (**L3: Apply**)
- **CO3:** Analyse the impact of E-banking, E-trading, and legal frameworks on E-commerce operations, including rights, obligations, and electronic contracts, to ensure compliance and efficiency. (**L4: Analyse**)
- **CO4:** Evaluate E-commerce security mechanisms, including firewalls, virus protection, and risk management, to protect systems from cyber threats and ensure secure transactions. (**L5: Evaluate**)
- **CO5:** Develop secure mobile commerce solutions by addressing global challenges and integrating secure payment systems for effective mobile-based E-commerce services. (**L3: Develop**)

CO-PO Mapping table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	3	2	2	0	0	3
CO2	2	3	3	3	2	0	0	3
CO3	2	3	3	3	2	0	0	3
CO4	2	3	3	2	2	0	0	3
CO5	3	3	3	3	3	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handle necessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Dr.G,Ramash Naidu

Verified by Dr.M.Srivenkatesh

SOCIAL NETWORK ANALYSIS

L	T	P	S	J	C
3	0	0	0	0	3

Course Description:

This course offers a comprehensive exploration of social networks and social media from a data analysis perspective, covering the structural properties, dynamics, and impacts of networks in various contexts. Students will learn about the theory and applications of social network analysis (SNA), including the use of graphs to represent and analyze social structures. The curriculum integrates a blend of theoretical foundations with practical exercises involving network measures, centrality types, network growth models, link analysis algorithms, and community detection methods.

Course Objectives:

- To understand the Fundamentals of Social Networks
- To analyse Network Structures
- To evaluate Centrality and Influence
- To explore Network Models and Community Detection
- To apply Theories to Real-World Data

UNIT -I

Introduction: Defining Social Networks, Social Media; Social Network vs Social Media, Social Network Analysis-Introduction, Applications, Small World Phenomenon-Six degrees of separation- Milgram Experiment, Use case of Milgram experiment- Small World phenomenon in the current era of Facebook.

Graph Essentials: Graph Basics, Types of Graphs, Special Graphs in Social Networks. Introduction to Networks: Definition, Types of networks, Node vs Link-centric view, Other types of network views- local, temporal, and generalized, Three levels of Social Network Analysis. 9 Hours

UNIT -II

Network Measures: Network basics-Degree, Degree distribution, Paths, Clustering coefficients, connected components.

Node centrality Types- Degree, Closeness, Betweenness, Eigen, Katz, PageRank, Group; Hub and Authority; Assortativity, Similarity, Degeneracy, Transitivity and Reciprocity, Social Balance Theory, Similarity in Networks. 9 Hours

UNIT -III

Network Growth Models- Properties of Real-World Networks, Random Network Model, Ring lattice Network model, Watts-Strogatz, Preferential Attachment Model, Small World Model, Price's Model, Aging in Preferential Attachment. 9 Hours

UNIT –IV

Link Analysis- Strong and Weak Ties, Link Analysis Algorithm, PageRank-Matrix Manipulation and Convergence, DivRank, SimRank, PathSIM

Community Structure in Networks- Types of Communities, Community Detection Models, Community Detection- Disjoint, Overlapping, Local. 9 Hours

Unit-V:

Information Diffusion in Social Media- Herd Behaviour, Epidemics- SI, SIS, SIR, SIRS models for information diffusion, Cascade models- Probabilistic, Independent; Cascade Prediction.

Anomaly Detection in Networks- Outliers vs Network-based anomalies, Anomaly detection in static vs dynamic networks. Brief Introduction to Collusion in Online Social Networks. 9 Hours

Textbooks:

1. Tanmoy Chakraborty, Social Network Analysis, Wiley India Pvt Ltd., 2021, ISBN-978-93-5424-783-5.
2. Reza Zafarani, Mohammad Ali Abbasi, Huan Liu, Social Media Mining – An Introduction, Cambridge University Press, 2014.

Reference Books:

1. Hansen, Derek, Ben Sheiderman, Marc Smith., Analyzing Social Media Networks with NodeXL: Insights from a Connected World, Morgan Kaufmann, 2011.
2. Krishna Raj P.M, Ankit Mohan, K.G. Srinivasa, Practical Social Network Analysis with Python, Springer, Computer Communications and Networks, 2018, ISBN 978-3-319-96745-5.
3. Iribarren, José Luis, and Esteban Moro. "Information diffusion epidemics in social networks." *arXiv preprint arXiv:0706.0641* (2007).
4. David Easley, Jon Kleinberg, Networks, Crowds, and Markets: Reasoning About a Highly Connected World, Cambridge University Press, 2010.
5. Sergey Edunov, Smriti Bhagat, Moira Burke, Carlos Diuk, Ismail Onur Filiz, Three and a half degrees of separation, Meta, 2016, https://research.facebook.com/blog/2016/2/three-and-a-half-degrees-of-separation/

Course Outcomes:

After completing the course student will be able to:

CO1: Understand the components, strategies, and trends of digital marketing, including the creation and implementation of a comprehensive digital marketing plan. *(L2: Understand)*

CO2: Apply keyword analysis, audience segmentation, and various digital marketing channels like SEO, PPC advertising, and social media to attract and retain customers. *(L3: Apply)*

CO3: Analyse on-page and off-page SEO techniques, such as optimizing website performance, backlink creation, and competitor analysis, to improve search engine rankings. *(L4: Analyse)*

CO4: Evaluate online advertising strategies, including Google Ad Metrics, payment methods, and landing page optimization, to design and implement effective campaigns. *(L5: Evaluate)*

CO5: Develop social media marketing strategies and analyse digital media performance using tools like Google Analytics, blogging, and social sharing platforms to enhance customer engagement and retention. *(L6: Develop)*

CO-PO Mapping table

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1	3	3	3	3	3	3	3	3
CO2	2	2	2	2	2	2	2	2
CO3	1	1	1	1	1	1	1	1
CO4	-	-	-	-	-	-	-	-
CO5	-	-	-	-	-	-	-	-

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handle necessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Dr.U.Santosh Kumar Verified by Dr.M.Srivenkatesh

COMPILER DESIGN

L	T	P	S	J	C
3	0	0	0	0	3

Course Description:

This course delves into the fundamentals of compiler design, introducing the essential components and processes involved in translating source code written in a high-level programming language into executable machine code. Students will explore the architectural structure of compilers, including phases such as lexical analysis, syntax analysis, syntax-directed translation, intermediate code generation, code optimization, and code generation. The course incorporates practical tools and methods used in compiler construction, such as LEX/FLEX for lexical analysis and YACC for parser generation.

Course Objectives:

- To understand Compiler Structure and Functionality
- To master Lexical Analysis
- To develop Skills in Syntax Analysis
- To implement Syntax-Directed Translation and Intermediate Code Generation
- To optimize and Generate Target Code

UNIT-I

Introduction & Lexical Analysis

Introduction: The Structure of Compiler, The Science of Building a Compiler in Bootstrapping and Cross compiler.
Lexical Analysis: The role of the Lexical analyzer, Input Buffering, Specification of Tokens, Recognition of Tokens, The Lexical Analyzer Generator (LEX/FLEX). 9 Hours

UNIT-II

Syntax Analysis (Part-I)

Syntax Analysis (Part-I): Introduction, Context-Free Grammars, Top-Down parsing: Brute force Parsing, Recursive Descent Parsing, Non-recursive Predictive Parsing, Error Recovery in Predictive Parsing, Bottom- Up parsing - Shift Reduce Parsing, Operator-Precedence Parsing, Error Recovery in Operator Precedence Parsing. 9 Hours

UNIT-III

Syntax Analysis (Part-II)
Introduction to LR Parsing: Simple LR Parser, More Powerful LRParsers (CLR&LALR), Using Ambiguous grammars, Error Recovery in LR Parsers, Parser Generator (YACC). 9 Hours

UNIT-IV

Syntax Directed Translation & Intermediate Code

Generation
Syntax Directed Translation: Syntax Directed Definitions, Types of attributes, Evaluation Orders for SDD's, Applications of Syntax-Directed Translation.
Intermediate Code Generation: Three Address codes, Types & Declarations, Translation of Arithmetic Expressions, Back patching for Boolean Expressions, and Flow of Control. 9 Hours

UNIT-V

Code Optimization & Code Generation

Code Optimization: The Principal Sources of Optimization, Basic blocks and Flow Graphs, Optimization of Basic Blocks, Introduction to Data-Flow Analysis: Live Variable Analysis
Code Generation: Issues in designing a code Generator, The Target Language, A Simple Code Generator, Peephole Optimization, Register allocation, and Assignment. 9 Hours

TextBooks:

1. Alfred.V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey. D. Ullman, Compilers Principles, Techniques and Tools, 2/e, Pearson Education, 2008.

References:

1. Alfred.V. Aho, J.D.Ullman, Principles of compiler design, Narosa Publications, 2002

2. John R. Levine, Tony Mason, Doug Brown, Lex &yacc, O'reilly ,2/e, 1992

3. Keith Cooper, Linda Torczon, Engineering a compiler, Morgan Kaufmann, 2/e, 2011

4. Samuel, M. P., Philip, P. K., 1997, "Powder metallurgy tool electrodes for electric discharge machining", International journal of machine tools and manufacture, 37, 1625–33.

Course Outcomes:
After completing the course student will be able to:

CO1: Understand the features, functions, and business models of E-commerce, including B2C, B2B, C2C, and C2B, to identify opportunities and challenges in the digital market. *(L2: Understand)*
CO2: Apply E-marketing strategies, including online advertising and branding, and evaluate digital payment systems like credit cards, mobile payments, and e-cash to enhance customer engagement and transaction efficiency. *(L3: Apply)*
CO3: Analyse the impact of E-banking, E-trading, and legal frameworks on E-commerce operations, including rights, obligations, and electronic contracts, to ensure compliance and efficiency. *(L4: Analyse)*
CO4: Evaluate E-commerce security mechanisms, including firewalls, virus protection, and risk management, to protect systems from cyber threats and ensure secure transactions. *(L5: Evaluate)*
CO5: Develop secure mobile commerce solutions by addressing global challenges and integrating secure payment systems for effective mobile-based E-commerce services. *(L3: Develop)*

CO-PO Mapping table

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1	3	3	3	3	2	0	0	3
CO2	3	3	3	3	2	0	0	3
CO3	3	3	3	3	2	0	0	3
CO4	3	3	3	3	2	0	0	3
CO5	3	3	3	3	2	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handle necessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Dr.M.Srivenkatesh Verified by Dr.M.Srivenkatesh

MACHINE LEARNING LAB

L	T	P	S	J	C
0	0	2	0	0	1

Course Description:

This laboratory course provides hands-on experience in machine learning using Python, focusing on practical applications of data manipulation, exploratory data analysis, data visualization, and predictive modelling. Throughout the course, students will work with popular Python libraries such as Pandas for data handling and analysis, Matplotlib and Seaborn for data visualization, and Scikit-learn for implementing machine learning algorithms. The lab exercises are designed to reinforce the theoretical concepts covered in machine learning courses by applying them to real-world datasets and scenarios.

Course Objectives:

- To master Data Manipulation and Analysis Using Pandas
- To conduct Exploratory Data Analysis (EDA)
- To develop Skills in Data Visualization
- To implement Machine Learning Models
- To evaluate Model Performance

Lab Exercises

1. Creating a Data Frame in Pandas from csv files.
2. Importing Data with Pandas – adding columns to the data frame.
3. Handling Missing Data- drop, fill, aggregate functions.
4. Indexing Data Frames with Pandas, Indexing Using Labels in Pandas.
5. Exploratory Data Analysis with Pandas- for both one dimensional and two dimensional data (series or data frames) - describe, group data, ANOVA, correlation and correlation methods, rank.
6. Calculating Mean, Trimmed Mean, Weighted Mean, Median,
7. Plotting using pandas- Exploratory analysis based on the plots.
8. Data Visualization with different charts in python.
9. A weather prediction model that predicts if there'll be rain or not in a particular day with decision tree regression concept.
10. A Python script to create a confusion matrix on a predicted model.
11. Write a program demonstrates how to use K-Nearest Neighbors (KNN), Decision Trees, Random Forest, and Support Vector Machines (SVM) for classification
12. Write a program demonstrates linear regression, multiple linear regression, and polynomial regression using synthetic data.
13. This program demonstrates K-Means and hierarchical clustering using synthetic data.

Text Books:

1. Machine Learning by Subramanian, Chandra Mouli, Amit Kumar Das , Saikant Dutt, Pearson Publications, I edition, 2018.

2. Machine Learning by Tom Mitchell, McGraw Hill,

Course Outcomes:

After completing the course student will be able to:

- **CO1:** Understand the fundamental concepts of social networks, social media, and graph essentials, including network views and levels of social network analysis, to explore real-world social interactions.

- **CO2:** Apply network measures such as centrality, clustering coefficients, assortativity, and social balance theory to analyse the structure and dynamics of social networks. (**L3: Apply**)

- **CO3:** Analyse network growth models, including random networks, preferential attachment, and small-world models, to study the properties of real-world social networks. (**L4: Analyse**)

- **CO4:** Evaluate link analysis algorithms like PageRank, SimRank, and community detection models to understand community structures and relationships within networks. (**L5: Evaluate**)

- **CO5:** Develop models for information diffusion and anomaly detection in social networks to predict cascading behaviors, identify outliers, and detect collusion in online platforms. (**L3: Develop**)

CO-PO Mapping table

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1	3	3	2	3	2	0	0	3
CO2	3	3	2	3	2	0	0	3
CO3	3	3	2	3	2	0	0	3
CO4	3	3	3	3	2	0	0	3
CO5	3	3	3	3	2	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handle necessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Dr.M.Srivenkatesh
Mrs Shahid Shaik

Verified by Dr.M.Srivenkatesh

SERVER-SIDE PROGRAMMING WITH JAVA LAB

L	T	P	S	J	C
0	0	2	0	0	1

Course Description:

This laboratory course is designed for students to apply Java technologies in full stack development, covering both front-end and back-end components. Through a series of hands-on lab exercises, students will gain practical experience in using Java Servlets, JavaServer Pages (JSP), Java Persistence API (JPA), Hibernate, and Spring frameworks. The course focuses on building dynamic web applications, implementing server-side programming, managing sessions, interacting with databases, and developing secure APIs.

Course Objectives:

- To understand and apply Java Servlet technology.
- To develop Dynamic Web Applications
- To implement Database Interactions
- To utilize ORM Techniques
- To leverage Spring Framework

Lab Exercise Programs

1. Write A JAVA Servlet Program To Implement And Demonstrate Get() And Post Methods (Using HTTP Servlet Class).
2. Write A JAVA Servlet Program To Implement Request dispatcher Object (Use Include() And Forward() Methods).
3. Write A JAVA Servlet Program To Implement Sessions (Using HTTP Session Interface).
4. Write A JSP Program Which Uses Jsp:Include & Jsp:Forward Action To Display A Web Page.
5. Write A JSP Program To Implement All The Attributes Of Page Directive Tag.
6. write a JSP program to upload image file into server location?
7. Write a JSP which does the following job: Insert the details of the 3 or 4 users who register with the web site by using registration form. Authenticate the user when he/she submits the login form using the user name and password from the database
8. Create tables in the database which contain the details of items (books in our case likeBook name, Price, Quantity, Amount)) of each category. Modify your catalogue page in such a way that you should connect to the database and extract data from the tables and display them in the catalogue page using JDBC.
9. Write a java JSP program to connect to that database and execute various SQL queries.
10. Develop a JPA Application to demonstrate use of ORM associations
11. Develop a Hibernate application to store and retrieve employee details in Database
12. Create a simple application that uses Spring for dependency injection.
13. Create a simple REST API using Spring Boot.
14. Demonstrate transaction management with Spring boot

Text Books :

1. Head First Servlets and JSP: Passing the Sun Certified Web Component Developer Exam Bryan Basham, Kathy Sierra, and Bert Bates, O'Reilly Media, 2nd Edition (2008)
2. Full Stack Java Development with Spring MVC, Hibernate, jQuery, and Bootstrap Mayur Ramgir,Kindle Edition,

Course Outcomes:

After completing the course student will be able to:

- **CO1:** Understand the core concepts of Java Servlets and JSP, including the use of HTTP methods, request dispatchers, and session management, to develop dynamic web applications. (**L2: Understand**)

- **CO2:** Apply JSP directives, file upload techniques, and database connectivity to create interactive and data-driven web pages. (**L3: Apply**)

- **CO3:** Develop JDBC and ORM frameworks like JPA and Hibernate to implement efficient data storage and retrieval systems. (**L6: Develop**)

- **CO4:** Apply dependency injection mechanisms in Spring and REST API development using Spring Boot to design modular and scalable applications. (**L3: Apply**)

- **CO5:** Develop robust enterprise-level applications by demonstrating transaction management in Spring Boot for ensuring data consistency and reliability. (**L3: Develop**)

CO-PO Mapping table

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1	3	3	3	3	2	1	0	3
CO2	3	3	3	3	2	1	0	3
CO3	3	3	3	3	2	1	0	3
CO4	3	3	3	3	2	1	0	3
CO5	3	3	3	3	2	1	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	

The topics included in this course are designed to get acquainted with one of the skills that handle necessary mathematical orientation, programming techniques and concept based learning.

Prepared by Mr.B.Satyanarayana

Verified by Dr.M.Srivenkatesh

SERVER-SIDE PROGRAMMING WITH PYTHON LAB

L	T	P	S	J	C
0	0	2	0	0	1

Course Description:

This laboratory course is designed to provide hands-on experience in full stack web development using Python, with a particular focus on Flask and Django frameworks. Students will engage in building functional web applications that showcase their ability to integrate both front-end and back-end technologies effectively. The course covers a broad range of activities from setting up simple web servers to developing complex RESTful APIs and deploying web applications to production environments like Heroku.

Course Objectives:

- To develop Web Applications with Flask
- To implement Database Integration
- To construct RESTful APIs
- To master Django's MVT Architecture
- To deploy Applications to Production

Lab Exercise Programs

1. Write a Python script that makes an HTTP GET request to a specified URL and processes the JSON response to extract and print specific information.
2. Write a program for developing a simple Flask application with message.
3. Write a Flask application with basic routing that renders different HTML templates for different URLs.
4. Develop a Flask application that includes a form for user input. The application should process the input and display the result on a new page. Implement sessions to maintain user state.
5. Develop a larger Flask application using blueprints. Create at least two blueprints representing different sections of the application.
6. Develop a simple Flask application that connects to a SQLite database using SQLAlchemy.
7. Create a model for a simple entity (e.g., `User`) and perform CRUD operations through the Flask application.

8. Build a RESTful API using Flask-RESTful that supports CRUD operations on a resource (e.g., `Product`). Implement endpoints for creating, reading, updating, and deleting resources.

9. Create a simple Django application that renders a "Hello, World!" message using the MVT pattern.

10 .Create Django models for a simple entity (e.g., `Book`) with fields such as title, author, and publication date. Perform migrations and use the Django ORM to perform CRUD operations on the models.

11.Create views and class-based views to handle HTTP GET and POST requests in a Django application. Develop a form to collect user data and handle form submissions.

12 .Demonstrate Django signals to perform specific actions when certain events occur (e.g., sending an email when a new user registers).

13. Develop and deploy a Django application on Heroku environment .

Text Books :

1. Flask Web Development: Developing Web Applications with Python by Miguel Grinberg Python Web Programming with Flask by Fabrizio Romano, O'Reilly Media,2nd Edition ,2018

2. Django for Beginners: Build websites with Python and Django by William S. Vincent, 3rd Edition (2020)

3. Two Scoops of Django 3.x: Best Practices for the Django Web Framework by Audrey Roy Greenfeld and Daniel Roy Greenfeld , Two Scoops Press,1st Edition ,2020

4. Django 3 By Example by Antonio Melé, Packt Publishing, 3rd Edition ,2020

Course Outcomes:

After completing the course student will be able to:

- **CO1:** Understand the fundamentals of building web applications using Flask and Django, including routing, templates, and the MVT pattern, to create dynamic web pages. *(L2: Understand)*
- **CO2:** Apply form handling, session management, and CRUD operations using Flask and Django to develop user-interactive and database-driven web applications. *(L3: Apply)*
- **CO3:** Analyse the use of advanced Flask and Django features, including blueprints, RESTful APIs, and Django ORM, to structure scalable and maintainable web applications. *(L4: Analyse)*
- **CO4:** Evaluate the integration of event-driven features like Django signals and deploy web applications on cloud platforms like Heroku to enhance functionality and accessibility. *(L5: Evaluate)*
- **CO5:** Develop robust and efficient web solutions by combining Flask and Django frameworks with databases, RESTful APIs, and deployment strategies for real-world applications. *(L3: Develop)*

CO-PO Mapping table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	3	3	2	0	0	3
CO2	3	3	3	3	2	0	0	3
CO3	3	3	3	3	2	0	0	3
CO4	3	3	3	3	2	0	0	3
CO5	3	3	3	3	2	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handle necessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Dr.MSrivenkatesh
Mr. Shahid Shaik

Verified by Dr.M.Srivenkatesh

BUSINESS INTELLIGENCE LAB USING POWER BI

L	T	P	S	J	C
0	0	2	0	0	1

Course Description:

This laboratory course is designed to provide hands-on experience with Power BI, a leading business intelligence tool, to create dynamic visualizations and business intelligence reports. Students will learn to import data from various sources, perform data transformation and loading (ETL) processes, and develop interactive visualizations and dashboards that make data-driven insights accessible and actionable.

Course Objectives:

- Master Data Importation and Transformation
- Develop Skills in ETL Processes
- Create Interactive Visual Reports
- Design and Implement Dynamic Dashboards
- Apply DAX for Advanced Data Modelling

Lab Exercise Programs

1. Import the data from different data sources (xls ,OLE DB, CSV other formats) and load in the target system
2. Perform the Extraction Transformation and Loading (ETL) process to construct the database in the Power BI
3. Create an interactive chart showing Units in Stock by Product and Total Sales by Year
4. Create a stacked Area Chart using a sample diabetic patients data set to show increase in no of patients by year and month
5. Create an Interactive report using a sample data set to show Filled Map for Geographical Region
6. Create an Interactive report using a sample data set to show Bar Chart for Age Distribution
7. Create an Interactive report using a sample data set to show Pie Chart for Gender Classification
8. Create an Interactive report using a sample data set to show Tree Map for Job Classification
9. Create a dynamic report using a sample data set to show reflections in other charts.
10. Create a dynamic dashboard for diabetic patients increase with respect to age , gender and geographical area
11. Create a dynamic dashboard for a sample dataset to visualize the increases of customers based on age, c
12. Company wise and geographical area

Text Books :

1. Mastering Microsoft Power BI: Expert techniques for effective data analytics and business intelligence, Brett Powell,: Packt Publishing, 2nd Edition, 2020
2. Applied Microsoft Power BI (2nd Edition): Bring your data to life!,Authors: Teo Lachev Publisher: Prologika Press, 2nd Edition,2021
3. Analyzing Data with Power BI and Power Pivot for Excel", Alberto Ferrari, Marco Russo Microsoft Press, 1st Edition ,2017

Course Outcomes:

After completing the course student will be able to:

- **CO1:** Understand the process of importing and integrating data from various sources, including Excel, OLE DB, and CSV, into Power BI for analysis and reporting. *(L2: Understand)*
- **CO2:** Apply the ETL (Extract, Transform, Load) process in Power BI to construct and transform databases for effective data visualization. *(L3: Apply)*
- **CO3:** Analyse data trends using interactive visualizations such as area charts, bar charts, pie charts, and tree maps to provide insights into patterns like stock levels, sales, and demographic distributions. *(L4: Analyse)*
- **CO4:** Evaluate dynamic reporting techniques, including reflections and interactivity across charts and maps, to enhance user engagement and decision-making. *(L5: Evaluate)*
- **CO5:** Develop dynamic dashboards to visualize complex relationships between variables, such as patient demographics and sales trends, and effectively communicate key business insights. *(L3: Develop)*

CO-PO Mapping table

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1	3	3	3	3	2	0	0	3
CO2	3	3	3	3	2	0	0	3
CO3	3	3	3	3	2	0	0	3
CO4	3	3	3	3	2	0	0	3
CO5	3	3	3	3	2	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handle necessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Mr. Shahid Shaik

Verified by Dr.M.Srivenkatesh

AGILE SOFTARE DEVELOPMENT LAB

L	T	P	S	J	C
0	0	2	0	0	1

Course Description:

This laboratory course is designed to provide practical, hands-on experience with various Agile software development methodologies, including Scrum, Extreme Programming (XP), Feature-Driven Development (FDD), and Test-Driven Development (TDD). Students will engage in a series of exercises that simulate real-world software development projects to understand and apply the principles and practices of Agile development.

Course Objectives:

- Familiarize with Agile Tools and Environments
- Understand and Compare Agile Methodologies
- Apply XP and TDD Practices
- Simulate Scrum Processes
- Integrate and Implement Agile Modelling and FDD

Lab Exercise Programs

Lab 1: Agile Practices and Tools

Objective: Familiarize students with Agile tools.

Exercise: Set up and use Jira or Trello to create a project. Define epics, stories, and tasks using an example project of your choice. Explore features like backlogs, sprints, and boards.

Deliverable: A screenshot of the project setup and a brief report on the tool's features.

Lab 2: Introduction to Agile Methods

Program :Agile Methodology Comparison

Objective: Compare and contrast different Agile methodologies.

Exercise: Research and create presentations or a comparative analysis document on various Agile methodologies mentioned in the unit (XP, SCRUM, DSDM, FDD, TDD). Highlight key practices, benefits, and challenges of each.

Deliverable: Presentation or document detailing the findings.

Lab 3: Extreme Programming (XP)

Objective: Understand and apply XP practices.
Exercise: Conduct a pair programming session. Implement a simple application (e.g., a calculator or a todo list) where one student writes the test and the other writes the code to pass the test.
Deliverable: Code repository (e.g., GitHub) with commit history showing pair programming practice.

Lab 4: Extreme Programming (XP)

Program :XP Simulation
Objective: Simulate an XP environment to understand its dynamics.
Exercise: Organize a coding dojo session where students practice XP's 12 practices.
Deliverable: Code repository and a reflection on the experience and effectiveness of XP practices.

Lab 5: Test-Driven Development (TDD)

Objective: Practice TDD by developing features starting with tests.
Exercise: Develop a small software feature using TDD. Start with writing tests for the requirements
and then write the code to pass the tests.
Deliverable: Code repository with test cases and the final code that passes all tests.

Lab 6: Agile Modeling and XP

Program: Agile Modeling Workshop
Objective: Apply Agile modeling techniques in a real project.
Exercise: Create a simple system design using Agile modeling techniques. Integrate these models into an existing XP project to demonstrate the fit and benefits of Agile modeling.

- **Deliverable:** A set of Agile models (like user story maps, UML diagrams) and a brief report on how these models improved the project planning and implementation phases.

Lab 7: Scrum Simulation

Objective: Simulate a Scrum sprint to understand Scrum processes.
Exercise: Perform a two-week sprint simulation where students perform roles such as Scrum Master, Product Owner, and Team Members. Use daily standups, sprint planning, sprint review, and retrospective.
Deliverable: Sprint report including burndown chart, user stories completed, and retrospective findings.

Lab 8: Scrum Simulation

Program: Scrum Role Play
Objective: Experience the roles within a Scrum team.
Exercise: Conduct a role-playing session where students take on the roles of Product Owner, Scrum Master, and Development Team members. Tackle a mock project from product backlog creation to sprint review.
Deliverable: Role play script, product backlog, sprint backlog, and a sprint review report.

Lab 9: Feature-Driven Development (FDD) and Incremental Software Development

Objective: Implement a feature using FDD principles.
Exercise: Plan and develop a feature using the FDD methodology. Focus on building by feature and track progress in a visual way.
Deliverable: Documentation of the planning and development process, including diagrams and

final code.

Lab 10: Feature-Driven Development and Test-Driven Development
Program: FDD and TDD Integration Project
Objective: Integrate FDD planning with TDD execution.
Exercise: Plan a small project using FDD, breaking down the project into features. Implement one of these features using TDD practices, focusing on writing tests first and then coding to fulfill the tests.
Deliverable: Document outlining the FDD plan, tests written, code developed, and how TDD influenced feature implementation.

Text Books:

1. John Hunt, Agile Software Construction,1st Edition, Springer,2005

2. Craig Larman, Agile and Iterative Development: A Manager's Guide, Addison-Wesley, Pearson Education – 2004.

3. Pearson, Robert C. Martin, Juli, James Shore, Chromatic 2013, The Art of Agile Development, O'Reilly Media.

4. Elisabeth Hendrickson, Agile Testing, Quality Tree Software Inc 2008.

Course Outcomes:
After completing the course student will be able to:

- **CO1:** Understand the fundamental principles of Agile methodologies and tools such as Jira or Trello to create and manage Agile projects. *(L2: Understand)*

- **CO2:** Apply Extreme Programming (XP) practices such as pair programming, test-first coding, and Agile modeling to develop and manage software projects effectively. *(L3: Apply)*

- **CO3:** Analyse the dynamics of Agile practices like Scrum and Feature-Driven Development (FDD) by simulating sprints and implementing feature-based development. *(L4: Analyse)*

- **CO4:** Evaluate the integration of Agile methodologies like FDD and Test-Driven Development (TDD) to enhance software quality and project planning. *(L5: Evaluate)*

- **CO5:** Develop comprehensive Agile project deliverables by integrating Agile tools, techniques, and methodologies such as TDD, FDD, and Scrum to solve real-world software development problems. *(L3: Develop)*

CO-PO Mapping table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	3	3	3	0	0	3
CO2	3	3	3	3	3	0	0	3
CO3	3	3	3	3	3	0	0	3
CO4	3	3	3	3	3	0	0	3
CO5	3	3	3	3	3	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handle necessary mathematical orientation, programming techniques and concept based learning.	

CLOUD COMPUTING LAB

L	T	P	S	J	C
0	0	2	0	0	1

Course Description:

This laboratory course is designed to provide students with practical, hands-on experience in cloud computing across multiple platforms including Microsoft Azure, Amazon Web Services (AWS), OpenAI, and Google Cloud Platform. Through a series of guided exercises, students will learn to configure and deploy virtual machines, manage cloud storage, develop serverless applications, and implement AI-powered applications using cloud APIs.

Course Objectives:

- To gain Proficiency with Major Cloud Platforms
- To master Virtual Machine Deployment and Management
- To deploy and Manage Web and Serverless Applications
- To implement Cloud Storage Solutions
- To develop AI Applications with Cloud APIs

Microsoft Azure Lab Exercises

1. **Creating a Virtual Machine:**
    - Objective: Create and configure a virtual machine in Azure.
    - Steps:
        - Create a new virtual machine.
        - Configure the operating system and network settings.
        - Connect to the virtual machine via Remote Desktop.
        - Install and configure necessary software.
2. **Azure App Services:**
    - Objective: Deploy a web application using Azure App Services.
    - Steps:
        - Create a new web app in Azure.
        - Deploy a sample application using Azure App Service.
        - Configure application settings and scaling options.
        - Monitor the app's performance and usage.
3. **Azure Storage:**
    - Objective: Utilize Azure Storage for data storage.
    - Steps:
        - Create a storage account.
        - Use Blob Storage to upload and manage files.
        - Configure access control and generate SAS tokens.
        - Implement a sample application to interact with Azure Storage.

AWS Lab Exercises

1. **Creating an EC2 Instance:**
    - Objective: Launch and configure an EC2 instance.
    - Steps:
        - Launch an EC2 instance using the AWS Management Console.
        - Configure security groups, key pairs, and network settings.
        - Connect to the instance using SSH.
        - Install and configure necessary software.
2. **Deploying a Serverless Application with AWS Lambda:**
    - Objective: Create a serverless application using AWS Lambda.
    - Steps:
        - Create a Lambda function.
        - Configure triggers and permissions.
        - Deploy and test the Lambda function.
        - Integrate Lambda with other AWS services like S3 and DynamoDB.
3. **AWS S3 Bucket Management:**
    - Objective: Manage data using AWS S3.
    - Steps:
        - Create an S3 bucket.
        - Upload, download, and manage objects in the bucket.
        - Configure bucket policies and permissions.
        - Implement versioning and lifecycle policies.

OpenAI Tools Lab Exercises

1. **Chatbot Development using OpenAI GPT:**
    - Objective: Create a chatbot using OpenAI GPT API.
    - Steps:
        - Set up the OpenAI API environment.
        - Create a basic chatbot application.
        - Implement conversation flows and context handling.
        - Test and deploy the chatbot.
2. **Text Generation with OpenAI GPT:**
    - Objective: Generate text using OpenAI GPT.
    - Steps:
        - Set up the OpenAI API environment.
        - Create a script to generate text based on prompts.
        - Experiment with different prompt designs and parameters.
        - Analyze the generated text and refine the prompts.
3. **Sentiment Analysis using OpenAI:**
    - Objective: Perform sentiment analysis using OpenAI GPT.
    - Steps:
        - Set up the OpenAI API environment.
        - Create a script to analyze sentiment in given text inputs.
        - Test the script with various text samples.
        - Integrate sentiment analysis into a larger application.

Google Cloud Lab Exercises

1. **Creating a Compute Engine Instance:**
    - Objective: Create and configure a Google Cloud Compute Engine instance.
    - Steps:
        - Create a new Compute Engine instance.
        - Configure the machine type, disk, and network settings.
        - Connect to the instance using SSH.
        - Install and configure necessary software.
2. **Google Cloud Storage:**
    - Objective: Utilize Google Cloud Storage for data storage.
    - Steps:
        - Create a Cloud Storage bucket.
        - Upload and manage objects in the bucket.
        - Configure access control and permissions.
        - Implement a sample application to interact with Cloud Storage.
3. **Deploying Applications using Google App Engine:**
    - Objective: Deploy a web application using Google App Engine.
    - Steps:
        - Create a new App Engine project.
        - Deploy a sample web application.
        - Configure application settings and scaling options.
        - Monitor the app's performance and usage.

Course Outcomes:

After completing the course student will be able to:

- **CO1:** Understand the fundamental concepts of cloud computing platforms such as Microsoft Azure, AWS, Google Cloud, and OpenAI tools to deploy and manage applications and services. (L2: Understand)
- **CO2:** Apply cloud-based services for creating and configuring virtual machines, deploying applications, and managing storage solutions. *(L3: Apply)*
- **CO3:** Analyse the integration of serverless architectures, APIs, and advanced cloud tools to optimize application deployment and functionality. *(L4: Analyse)*
- **CO4:** Evaluate the performance and scalability of cloud applications through monitoring tools, configuration settings, and resource management. *(L5: Evaluate)*
- **CO5:** Develop interactive and scalable cloud-based solutions, such as chatbots, sentiment analysis applications, and dynamic storage systems, by leveraging cloud-native tools and APIs. *(L3: Develop)*

CO-PO Mapping table

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1	3	2	3	3	2	0	0	3
CO2	3	2	3	3	2	0	0	3
CO3	3	2	3	3	2	0	0	3
CO4	3	2	3	3	2	0	0	3
CO5	3	2	3	3	2	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handle necessary mathematical orientation, programming techniques and concept-based learning.	

Prepared by Dr.M.Srivenkatesh

Verified by Dr.M.Srivenkatesh

INTERNSHIP /INDUSTRIAL TRAINING AND SEMINAR

L	T	P	S	J	C
0	0	0	0	0	2

Course Duration: 4 Weeks

Course Description: This introductory internship program is designed to bridge the gap between academic theories and real-world industry practices. Students will spend a four weeks at an industry site, gaining first-hand experience under the supervision of industry experts. This exposure prepares them for future engagements such as more advanced internships or full-time positions.

Course Educational Objectives:

- Gain practical exposure to industry standards and teamwork.
- Identify essential skills required for effective participation in real-time industry projects.
- Develop an understanding of the broader implications of engineering solutions, including their global, economic, environmental, and societal impacts.
- Acquire the foundational technical knowledge necessary for advancing to more complex internships.
- Enhance problem-solving skills by addressing real-world challenges, encouraging innovative and critical thinking approaches.

Mode of Evaluation:

- Internship Report: Students must submit a detailed report documenting their experiences and learnings.
- Presentation: A formal presentation reviewing the internship experience.
- Project Review: Evaluation based on the contributions and insights gained during the onsite project work.

Course Outcomes:

Upon successful completion of this internship, students will be able to:

CO1: Understand industry workflows, standards, and team dynamics through direct exposure to professional environments. *(L2: Understand)*
CO2: Apply technical knowledge and skills to real-world challenges encountered in industry projects. *(L3: Apply)*
CO3: Analyse the technical and non-technical aspects of industry projects, including economic, environmental, and societal impacts of engineering solutions. *(L4: Analyse)*
CO4: Evaluate personal and professional growth by reviewing learnings from industry engagement and assessing how these skills align with career goals. *(L5: Evaluate)*
CO5: Develop a detailed report and deliver a professional presentation summarizing internship experiences, contributions, and insights. *(L3: Develop)*

CO-PO Mapping table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	2	2	2	2	3	0	0	3
CO2	2	3	2	2	3	0	0	3
CO3	2	2	3	2	2	0	0	3
CO4	2	3	2	2	2	0	0	3
CO5	3	3	3	2	2	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handle necessary mathematical orientation, programming techniques and concept based learning.	

INTRODUCCTION TO BIG DATA AND ANALYTICS

L	T	P	S	J	C
3	0	0	0	0	3

Course Description:

This comprehensive course provides an in-depth exploration of Big Data technologies, focusing on the tools and techniques essential for managing, processing, and analyzing vast amounts of data. Students will learn about the evolution of data management and the role of Big Data in various business contexts such as social networking, retail, and insurance to prevent fraudulent activities.The course covers distributed and parallel computing fundamentals with a strong emphasis on Hadoop and its ecosystem, including Hadoop YARN, HBase, Hive, and Pig. Additionally, the course introduces cloud computing solutions for Big Data and in-memory computing technology. Advanced topics include the use of Apache Spark for performing complex data analysis with Resilient Distributed Datasets (RDDs), Data Frames, and the Spark Dataset API.

Course Objectives:

- To understand the Fundamentals of Big Data
- To master Big Data Processing with Hadoop
- To explore Advanced Big Data Tools
- To implement Big Data Solutions Using Spark
- To apply Big Data Technologies in Real-World Scenarios

UNIT – I

Getting an overview of Big Data: Big Data definition, History of Data Management, Structuring Big Data, Elements of Big-data, Big Data Analytics.
Exploring use of Big Data in Business Context: Use of Big Data in Social Networking, Use of Big Data in preventing Fraudulent Activities in Insurance Sector & in Retail Industry. 9 Hours

UNIT – II

Introducing Technologies for Handling Big Data: Distributed and parallel computing for Big Data, Introducing Hadoop, Cloud computing and Big Data, In-memory Computing Technology for Big Data.
Understanding Hadoop Ecosystem: Hadoop Ecosystem, Hadoop Distributed File System, MapReduce, Hadoop YARN, Introducing HBase, Combing HBase and HDFS, Hive, Pig and Pig Latin, Sqoop, ZooKeeper, Flume, Oozie.
Understanding MapReduce Fundamentals and HBase: The MapReduce Framework, Techniques to Optimize Map Reduce Jobs, Uses of Map Reduce, Role of HBase in Big Data Processing.
9 Hours

UNIT – III

Understanding Big Data Technology Foundations: Exploring the Big Data Stack, Virtualization and Big Data, Virtualization approaches.

Understanding Hadoop YARN Architecture: Background of YARN, Advantages of YARN, YARN Architecture, Working of YARN, YARN Schedulers, YARN Configurations, YARN commands. 9 Hours

UNIT – IV

Exploring Hive: Introducing Hive, Getting Started with Hive, Hive Services, Data Types, Built- in Functions, Hive-DDL, Data Manipulation, Data Retrieval Queries, Using Joins.

Analyzing Data with Pig: Introducing Pig, Running Pig, Getting started with Pig Latin, working with operators in Pig, Debugging Pig, Working with Functions in pig, Error Handling in Pig. 9 Hours

UNIT – V

Spark: Introduction, Spark Jobs and API, Spark 2.0 Architecture, Resilient Distributed Datasets: Internal Working, Creating RDDs, Transformations, Actions. DataFrames: Python to RDD Communications, speeding up PySpark with DataFrames, Creating Data Frames and Simple DataFrame Queries, Interoperating with RDDs, Querying with DataFrame API, Querying with SQL, DataFrame Scenario, Spark Dataset API. 9 Hours

Text book:

1. Big Data Black Book by Dt Editorial Services, Dreamtech Publications, 2016. (Unit- I-IV)
2. Learning PySpark by Tomasz Drabas, Denny Lee, Packt publishing, 2017. (Unit – V)

Course Outcomes:

After completing the course student will be able to:

CO1: Understand the fundamental concepts of Big Data, its history, and its applications in various industries, such as social networking and fraud prevention. *(L2: Understand)*

CO2: Apply knowledge of distributed computing and Hadoop ecosystem tools, including HDFS, MapReduce, and related technologies, for handling and processing Big Data. *(L3: Apply)*

CO3: Analyse the architecture and functionalities of YARN and virtualization approaches in Big Data processing. *(L4: Analyse)*

CO4: Evaluate the capabilities of Hive and Pig for data querying and manipulation in Big Data environments. *(L5: Evaluate)*

CO5: Develop efficient data processing pipelines using Spark, DataFrames, and RDDs to perform advanced analytics and querying. *(L3: Develop)*

CO-PO Mapping table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	3	3	2	0	0	3
CO2	3	3	3	3	2	0	0	3
CO3	3	3	3	3	2	0	0	3
CO4	3	3	3	3	2	0	0	3
CO5	3	3	3	3	2	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handle necessary mathematical orientation, programming techniques and concept based learning.	

CYBER SECURITY: CONCEPTS AND TECHNOLOGIES

L	T	P	S	J	C
3	0	0	0	0	3

Course Description:

This course offers an in-depth exploration of cybercrime, the various methodologies employed by cybercriminals, and the countermeasures that can be implemented to enhance information security. Students will learn about the origins and classifications of cybercrime, the evolution of cybersecurity measures, and the legal frameworks surrounding cybercrime, particularly in the Indian context with references to global perspectives. Throughout the course, students will be exposed to various cyber offences including social engineering, cyberstalking, and the threats associated with botnets and cloud computing. The proliferation of mobile and wireless devices brings additional security challenges, which will be examined in detail. The course also covers practical tools and methods used in committing cyber crimes such as phishing, password cracking, and various types of malware, as well as defensive techniques to protect against such threats.

Course Objectives:

- To understand the Scope and Impact of Cybercrime
- To analyze Cyber Offenses and Attack Strategies
- To learn Security Measures for Mobile and Wireless Devices
- To examine Tools and Techniques Used in Cybercrime
- To explore Cyber Forensics and Legal Countermeasures

UNIT-I

Introduction to cyber-crime: Definition & Origin of the Word, Cyber Crime and Information Security, Who are Cyber Criminals, Classification of cyber Crimes, Cyber Crime – Legal and Indian perspective, Cyber Crime and Indian ITA 2000, Global Perspective on cyber-crimes.

Cyber offences: Introduction, How criminals plan the attacks, Social engineering, Cyber talking, Cyber Café and Cyber Crimes.

Botnets: The fuel for Cyber Crimes, Attack Vector, Cloud Computing. 9 Hours

UNIT-II

Cybercrime-Mobile and Wireless Devices: Proliferation of mobile and wireless devices, Trends in mobility, Credit card Frauds in Mobile and Wireless computing Era, Security Challenges posted by Mobile Devices, Registry settings for mobile devices, Authentication Service Security, Attacks on mobile/cell phones

Mobile Devices: Security Implications for Organizations, Devices – Related Security Issues, Organizational Security Policies & Measures in mobile computing era, Laptops. 9 Hours

UNIT-III

Tools and Methods used in Cyber Crime: Proxy servers and Anonymigers, Phishing, Password Cracking, Key Loggers, and Spywares, Virus and Worms, Trojan Horses & Backdoors Steganography, DOS & DDOS Attacks, SQL Injection, Buffer Overflow, Attacks on wireless networks. 9 Hours

UNIT-IV

Cyber Crimes and Cyber Security Legal Perspectives: Cybercrime and Legal Landscape around the World, Cyber Laws Indian Context, Indian IT Act challenges to Indian Law & Cyber Crime Scenario in India Consequences of Not Addressing the Weakness in IT Act, Digital Signature & Indian IT Act, Cyber Crime & Punishment. 9 Hours

UNIT-V

Understanding Cyber Forensics: Historical Background, Digital Forensics Science, Need For Computer Forensics, Cyber Forensics & Digital Evidence, Forensic Analysis of Email, Digital Forensics Life Cycle, Chain of Custody Concept, Network Forensics, Computer Forensics & Steganography, Relevance of OSI 7 Layer Model to Computer Forensics, Forensics & Social Networking Sites, Challenges in Computer Forensics, Special Tools & Techniques, Forensics Auditing, Anti Forensics. 9 Hours

Text Book:

- Understanding Cyber Crimes, Computer Forensics & Legal Perspective by Sunit Belapure, Nina Godbole ,Wiely India , 2011

Reference Book:

- Cyber Security Essentials by Charles J.Brooks, Christopher Grow, Philip Craig , MC Grawhill, 2018

Course Outcomes:

After completing the course student will be able to:

- **CO1:** Understand the fundamental concepts of cybercrime, its origins, classifications, and legal perspectives, with an emphasis on Indian and global scenarios. (**L2: Understand**)
- **CO2:** Analyse the security challenges and organizational implications of cybercrime on mobile and wireless devices, including attacks and mitigation measures. (**L4: Analyse**)
- **CO3:** Evaluate tools and methods such as phishing, password cracking, and DDoS attacks, and understand their impact on cybersecurity. (**L5: Evaluate**)
- **CO4:** Examine the legal frameworks addressing cybercrime in the Indian and global context, including the Indian IT Act and its implications. (**L4: Analyse**)
- **CO5:** Apply digital forensic techniques to analyze cybercrime scenarios, including email forensics, network forensics, and forensic auditing, while addressing the challenges in cyber forensics. (**L3: Apply**)

CO-PO Mapping table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	3	2	2	0	0	3
CO2	3	3	3	2	2	0	0	3
CO3	3	3	3	3	2	0	0	3
CO4	3	3	3	2	2	0	0	3
CO5	3	3	3	2	2	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handle necessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Dr.V.Nagalakshmi

Verified by Dr.M.Srivenkatesh

BLOCKCHAIN TECHNOLOGIES: CONCEPTS AND APPLICATIONS

L	T	P	S	J	C
3	0	0	0	0	3

Course Description:

This course provides a comprehensive introduction to blockchain technology, exploring its development, underlying principles, and a variety of applications. Students will begin by learning the basics of blockchain as a public ledger system, tracing its evolution from simple cryptocurrency implementations like Bitcoin to more sophisticated applications in Blockchain 2.0, including smart contracts and permissioned blockchains.The curriculum delves into the cryptographic underpinnings essential for securing transactions within a blockchain network, including hash functions, digital signatures, and public and private key cryptography. It also covers the various mechanisms used to achieve distributed consensus, which is vital for maintaining the integrity and consistency of blockchain data across disparate nodes.

Course Objectives:

- To understand the Fundamentals of Blockchain
- To explore Cryptographic Techniques Used in Blockchain
- To examine Distributed Consensus Mechanisms
- To develop Skills in Designing and Deploying Smart Contracts
- To apply Blockchain in Real-World Scenarios

UNIT – I

Introduction: What is Blockchain (BC), public ledgers, BC as public ledgers; BC history - Bitcoin and Cryptocurrency, BC 2.0, Smart contracts; BC architecture – Blocks in BC, transactions and distributed consensus; BC conceptualization - The Chain and the Longest Chain, Cryptocurrency to Blockchain 2.0, Permissioned Model of Blockchain. 9 Hours

UNIT -II

Cryptographic Primitives: Cryptographic Hash Function, Properties of a hash function, Hash pointer and Merkle tree, Digital Signature, Public Key Cryptography, Private Key Cryptography, A basic cryptocurrency. 9 Hours

UNIT - III

Distributed consensus : Distributed consensus in open environments, Consensus in a Bitcoin network; Bitcoin Consensus - Proof of Work (PoW) – basic introduction, HashcashPoW, Beyond Consensus in Bitcoin - BitcoinPoW, Attacks on PoW and the monopoly problem, Proof of Stake, Proof of Burn and Proof of Elapsed Time; Consensus in Bitcoin (The Miners) - The life of a Bitcoin Miner, Mining Difficulty, Mining Pool. 9 Hours

UNIT - IV

Smart contracts - I : Smart contracts, Solidity, REMIX IDE, EthereumBlockchain, Ethereum Virtual Machine.

Smart contracts–II: Decentralized applications (Dapps), Truffle development, Design improvements, Application models and standards. 9 Hours

UNIT - V

Use Cases: Blockchain for Voting, Government Use-cases – Public distribution system, Blockchain for Tax Payments, Blockchain for Managing Land Registry Records, Other Block Chain Frame Works: IBM Hyperledge fabric

Research Aspets in Block Chain: Consensus protocols, Identity management, Strong and weak synchronization, avoiding forks, Mining improvements. 9 Hours

Text Books:

1. Drescher, Daniel. "Blockchain basics",A Non-Technical Introduction in 25 Steps Apress, 2017.
2. Mougayar, William. "The business blockchain: promise, practice, and application of the next Internet technology", John Wiley & Sons, 2016.
3. Dannen, Chris. "Introducing Ethereum and Solidity", Berkeley: Apress, 2017.
4. Prusty, Narayan. "Building Blockchain Projects", Packt Publishing Ltd, 2017.
5. Pilkington, Marc. "Blockchain technology: principles and applications" Research handbook on digital transformations, 2016.
6. Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller, Steven Goldfeder. Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction, Princeton University Press, 2016.
7. Swan, Melanie, "Blockchain: Blueprint for a new economy", O'Reilly Media, Inc., 2015.
8. Antonopoulos, Andreas M. "Mastering Bitcoin: unlocking digital crypto currencies", O'Reilly Media, Inc., 2014.

Course Outcomes:

After completing the course student will be able to:

- **CO1:** Understand the foundational concepts of blockchain technology, including its history, architecture, and the evolution from cryptocurrency to Blockchain 2.0. (**L2: Understand**)
- **CO2:** Apply cryptographic primitives such as hash functions, Merkle trees, and digital signatures to secure blockchain operations. (**L3: Apply**)
- **CO3:** Analyse distributed consensus mechanisms like Proof of Work (PoW), Proof of Stake (PoS), and their role in achieving consensus in blockchain networks. (**L4: Analyze**)
- **CO4:** Develop smart contracts and decentralized applications (Dapps) using Solidity, Ethereum Blockchain, and other tools for blockchain-based solutions. (**L3: Develop**)
- **CO5:** Evaluate real-world blockchain use cases, frameworks like IBM Hyperledger Fabric, and explore research areas in consensus protocols, identity management, and mining improvements. (**L5: Evaluate**)

CO-PO Mapping table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	3	0	2	1	0	3
CO2	3	3	3	2	2	1	0	3
CO3	3	3	3	2	2	1	0	3
CO4	3	3	3	2	3	1	0	3
CO5	3	3	3	2	3	1	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handle necessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Dr.V.Nagalakshmi

Verified by Dr.M.Srivenkatesh

DEEP LEARNING AND GENERATIVE AI

L	T	P	S	J	C
3	0	0	0	0	3

Course Description:

This comprehensive course offers an in-depth exploration of deep learning technologies and their applications, particularly focusing on neural networks, convolutional neural networks (CNNs), and generative models such as Generative Adversarial Networks (GANs) and ChatGPT. Beginning with a historical overview of deep learning, the course covers foundational concepts like perceptrons and multi-layer perceptrons, advances through sophisticated architectures like autoencoders and residual networks, and culminates in cutting-edge applications in generative AI.Students will learn about different types of neural network training techniques, including gradient descent variations and regularization methods to combat overfitting. The course also delves into both supervised and unsupervised deep learning models, providing a broad spectrum of learning strategies from traditional CNNs used in image recognition to recent trends in unsupervised learning and recurrent neural networks.

Course Objectives:

- To understand Foundations of Deep Learning
- To master CNN Architectures and Applications
- To explore Unsupervised Learning Techniques
- To develop Skills in Generative AI
- To address Ethical and Practical Challenges in AI Deployment

UNIT -I

INTRODUCTION–History of Deep Learning, Introduction to Neural Network, Perceptrons, Perceptron Learning Algorithm. Multilayer Perceptrons (MLPs), Representation of MLPs, Sigmoid Neurons, Gradient Descent. Feed Forward Neural Networks, Backpropagation. Gradient Descent (GD), Momentum Based GD, Stochastic GD;

REGULARIZATION Bias Variance Tradeoff, L2regularization, Earlystopping, Dataset augmentation, Parameter sharing and tying; Supervised Deep Learning-Convolutional Neural Networks, Building blocks of CNN. 12 Hours

UNIT -II

Transfer Learning, LeNet, AlexNet, ZFNet, VGGNet, GoogLeNet, ResNetModels, Visualizing Convolutional Neural Networks. 12 Hours

UNIT -III

Unsupervised Learning with Deep Network, Autoencoders, Variational Autoencoder, Regularization in autoencoders, Denoising autoencoders, Sparse autoencoders; Recent Trends in Deep Learning Architectures,

Residual Network, Skip Connection Network, Fully Connected CNN, Recurrent Neural Networks(RNN), Long Shot Term Memory (LSTM) . 12 Hours

UNIT -IV

Networks, Generative Adversarial Networks(GAN); Classical Supervised Tasks with Deep Learning, Image Denoising, Semantic Segmentation, Object Detection. 12 Hours

UNIT -V

Introduction to Generative AI Models, Explainable AI, Prompt Engineering, ChatGPT, Fine-tuning ChatGPT, Ethical Considerations in Generative AI Models & ChatGPT. The Future of Generative AI, Deploying and Scaling ChatGPT, Security and Privacy Considerations, Monitoring and Debugging ChatGPT, Maintaining ChatGPT. 12 Hours

Text Books

1. Ian Good Fellow, Yoshua Benjio, Aaron, Deep Learning Courville, The MIT Press.
2. Christopher Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

Course Outcomes:

After completing the course student will be able to:

Course Outcomes

- **CO1:** Understand the fundamental concepts of neural networks, gradient descent, backpropagation, and regularization techniques in supervised deep learning. *(L2: Understand)*
- **CO2:** Analyse and compare deep learning architectures such as LeNet, AlexNet, ResNet, and VGGNet for transfer learning and visualization. *(L4: Analyse)*
- **CO3:** Apply unsupervised learning techniques, including autoencoders and recent deep learning architectures like RNNs and LSTMs, for data representation and sequence modelling. *(L3: Apply)*
- **CO4:** Develop deep learning models for classical tasks like image denoising, semantic segmentation, and object detection using GANs and other deep learning techniques. *(L3: Develop)*
- **CO5:** Evaluate generative AI models, including ChatGPT, and address ethical considerations, deployment strategies, and security/privacy aspects in generative AI solutions. *(L5: Evaluate)*

CO-PO Mapping table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	3	3	2	0	0	3
CO2	3	3	3	3	2	0	0	3
CO3	3	3	3	3	2	0	0	3
CO4	3	3	3	3	2	0	0	3
CO5	3	3	3	3	2	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handle necessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Dr.M.Srivenkatesh

Verified by Dr.M.Srivenkatesh

FUNDAMENTLS OF NATURAL LANGUAGE PROCESSING

L	T	P	S	J	C
3	0	0	0	0	3

Course Description:

This course offers an in-depth exploration of Natural Language Processing (NLP), a crucial field at the intersection of computer science, artificial intelligence, and linguistics. Students will learn about the fundamental techniques and algorithms used to process and analyze human language, preparing them to tackle complex challenges in NLP.The curriculum begins with an introduction to NLP, discussing its applications and the difficulties posed by natural language ambiguity. It progresses to language modeling, covering both traditional N-gram models and advanced neural network approaches. The course also includes practical exercises in parts-of-speech tagging, syntactic parsing, and semantics, integrating both theoretical concepts and hands-on learning.

Course Objectives:

- To understand the Fundamentals of NLP
- To develop Skills in Language Modelling
- To master Parts-of-Speech Tagging
- To implement Syntactic Parsing Techniques
- To explore Semantic Analysis

UNIT-I

Introduction to NLP NLP – introduction and applications, NLP phases, Difficulty of NLP including ambiguity; Spelling error and Noisy Channel Model; Concepts of Parts-of-speech and Formal Grammar of English.

9 Hours

UNIT -II

Language Modelling: N-gram and Neural Language Models Language Modelling with N-gram, Simple N-gram models, Smoothing (basic techniques), Evaluating language models; Neural Network basics, Training; Neural Language Model, Case study: application of neural language model in NLP system development .

9 Hours

UNIT-III

Parts-of-speech Tagging Parts-of-speech Tagging: basic concepts; Tagset; Early approaches: Rule based and TBL; POS tagging using HMM, Introduction to POS Tagging using Neural Model. 9 Hours

UNIT-IV

Parsing Basic concepts: top down and bottom up parsing, treebank; Syntactic parsing: CKY parsing; Statistical Parsing basics: Probabilistic Context Free Grammar (PCFG); Probabilistic CKY Parsing of PCFGs. 9 Hours

UNIT-V

Semantics Vector Semantics; Words and Vector; Measuring Similarity; Semantics with dense vectors; SVD and Latent Semantic Analysis; Embeddings from prediction: Skip-gram and CBOW; Concept of Word Sense; Introduction to WordNet .

9 Hours

Text books:

1. Jurafsky Dan and Martin James H. "Speech and Language Processing" ,3rd Edition, 2018.

Reference books:

1. Jurafsky D. and Martin J. H., "Speech and language processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition", 2nd Edition, Upper Saddle River, NJ: Prentice-Hall, 2008.
2. Goldberg Yoav "A Primer on Neural Network Models for Natural Language Processing".

Course Outcomes

After completing the course student will be able to:

- **CO1:** Understand the foundational concepts and phases of Natural Language Processing, including the challenges such as ambiguity and spelling errors. (**L2: Understand**)
- **CO2:** Apply N-gram models and neural language models for language modeling tasks, including smoothing techniques and system evaluation. (**L3: Apply**)
- **CO3:** Analyse and implement different approaches for Parts-of-Speech (POS) tagging, including rule-based, HMM, and neural model techniques. (**L4: Analyze**)
- **CO4:** Evaluate and apply parsing techniques such as CKY parsing and Probabilistic Context-Free Grammar (PCFG) for syntactic analysis of sentences. (**L5: Evaluate**)
- **CO5:** Apply and utilize vector semantics, embeddings, and tools like WordNet for semantic analysis and word sense disambiguation in NLP applications. (**L3: Apply**)

CO-PO Mapping table

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1	3	3	3	3	2	0	0	3
CO2	3	3	3	3	2	0	0	3
CO3	3	3	3	3	2	0	0	3
CO4	3	3	3	3	2	0	0	3
CO5	3	3	3	3	2	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handle necessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Dr.K.Vanitha

Verified by Dr.M.Srivenkatesh

FUNDAMENTLS OF DIGITAL IMAGE PROCESSING

L	T	P	S	J	C
3	0	0	0	0	3

Course Description:

This course introduces the fundamentals of digital image processing, focusing on the manipulation and analysis of digital images through the use of computer algorithms. Students will explore the origins and applications of image processing in various fields such as medical imaging, remote sensing, and multimedia applications. The course covers key processes in digital image processing, including image enhancement, restoration, compression, watermarking, and segmentation. Starting with basic concepts, the course will guide students through the fundamental steps involved in digital image processing and the components of an image processing system. Techniques for enhancing image quality through spatial filtering, histogram processing, and the use of various filters will be thoroughly discussed. Additionally, the course delves into more complex topics such as image restoration and reconstruction, addressing common problems such as noise and image degradation.

Course Objectives:

- To understand the Basics of Digital Image Processing
- To master Image Enhancement Techniques
- To develop Skills in Image Restoration
- To implement Image Compression and Watermarking
- To explore Image Segmentation Methods

UNIT I

Introduction: What is Digital Image Processing ,The Origins of Digital Image Processing , Examples of Fields that Use Digital Image Processing,Fundamental Steps in Digital Image Processing Components of an Image Processing System. 9 Hours

UNIT II

Image Enhancement Background ,Some Basic Intensity Transformation Functions,Histogram Processing ,Fundamentals of Spatial Filtering Smoothing (Lowpass) Spatial Filters ,Sharpening (Highpass) Spatial Filters, Highpass , Bandreject , and Bandpass Filters from Lowpass Filters, Combining Spatial Enhancement Methods. 9 Hours

UNIT III

Image Restoration and Reconstruction: A Model of the Image Degradation/Restorationprocess , Noise Models, Restoration in the Presence of Noise Only—Spatial Filtering , Periodic Noise Reduction Using Frequency Domain Filtering, Linear, Position-Invariant Degradations, Estimating the Degradation Function, Inverse Filtering, Minimum Mean Square Error (Wiener) Filtering, Constrained Least Squares Filtering, Geometric Mean Filter, Image Reconstruction from Projections.

9 Hours

UNIT IV

Image Compression and WaterMarking: Fundamentals, Huffman Coding, Golomb Coding, Arithmetic Coding, LZW Coding, Run-length Coding, Symbol-based Coding, Bit-plane Coding, Block Transform Coding, Predictive Coding, Wavelet Coding, Digital Image Watermarking.

9 Hours

UNIT V

Image Segmentation: Fundamentals, Point, Line, and Edge Detection, Thresholding, Segmentation by Region Growing and by Region Splitting and Merging, Region Segmentation Using Clustering and Super pixels, Region Segmentation Using Graph Cuts, Segmentation Using Morphological Watersheds, The Use of Motion in Segmentation. 9 Hours

Text Book

1. Rafael C. Gonzalez, Richard E. Woods, Digital Image processing, 4/e, Pearson Education Limited 2018.

References: 1. B. Chanda, D. Dutta Majumder, Digital Image Processing and Analysis, PHI, New Delhi, 2006.

2. A.K. Jain, Fundamentals of Digital Image Processing, PHI, New Delhi, 2006.

Course Outcomes

After completing the course student will be able to:

CO1: Understand the fundamental concepts, steps, and components involved in digital image processing. *(L2: Understand)*

CO2: Apply techniques for image enhancement using intensity transformations, spatial filtering, and frequency-based methods. *(L3: Apply)*

CO3: Analyse noise models and restoration techniques, including filtering methods and degradation modelling, to reconstruct degraded images. *(L4: Analyse)*

CO4: Evaluate and implement image compression methods and digital watermarking techniques for efficient storage and secure transmission. *(L5: Evaluate)*

CO5: Develop image segmentation techniques, including edge detection, region-based methods, and morphological approaches, for object identification in images. *(L3: Develop)*

CO-PO Mapping table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	3	3	2	0	0	3
CO2	3	3	3	3	2	0	0	3
CO3	3	3	3	3	2	0	0	3
CO4	3	3	3	3	2	0	0	3
CO5	3	3	3	3	2	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handle necessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Dr.K.Vanitha

Verified by Dr.M.Srivenkatesh

INTRODUCTION TO BIOINFORMATICS

L	T	P	S	J	C
3	0	0	0	0	3

Course Description:

This course offers a comprehensive introduction to the field of bioinformatics, focusing on the application of computational techniques to analyse biological data. Students will explore the intersection of biology, computer science, and information technology through the study of bioinformatics tools and methods used for managing, analysing, and predicting the structure and function of biological molecules. The curriculum starts with a historical overview of bioinformatics, detailing its evolution and foundational concepts, including the central dogma of molecular biology. The course covers a wide range of topics, from understanding different types of biological databases and data retrieval methods to advanced topics in genome analysis, gene mapping, and sequence alignment. Students will also delve into phylogenetic, learning about methods to analyse evolutionary relationships.

Course Objectives:

- To understand the Fundamentals of Bioinformatics
- To learn to Utilize Biological Databases
- To master Techniques in Genome Analysis and Sequence Alignment
- To explore Phylogenetic Analysis
- To apply Bioinformatics Tools to Protein Structure and Function Prediction

UNIT -I

Bioinformatics an Introduction: historical overview and definition, bioinformatics applications,
major databases in bioinformatics, data management and analysis, molecular biology, and bioinformatics, the central dogma of molecular biology.
Introduction to Biological Databases: Databases, Types of Databases, Biological Databases considerations
Information Search and Data Retrieval: introduction, Electronic Libraries, Tools for Web Search, Data Retrieval Tools, Data Mining of Biological Databases. 9 Hours

UNIT –II

Genome Analysis and Gene Mapping: introduction, Genome Analysis, Genome Mapping, Sequence Assembly problem, Genetic Mapping and Linkage Analysis, Human Genome Project.
Alignment of Pairs of Sequences: introduction, Biological motivation of alignment problems, methods of sequence alignments, using scoring matrices, measuring sequence detection efficiency
Alignment of Multiple Sequences: introduction, methods of multiple sequence alignment, evaluating multiple alignments, applications of multiple alignments. 9 Hours

UNIT-III

Introduction to Phylogenetics: introduction, taxonomic relationships from molecular properties, terminology, tree topologies, gene trees, tools for tree visualization.

Methods of Phylogenetic Analysis: introduction, Distance-based methods, Character-based methods, other methods, Tree evaluation, problems in phylogenetic analysis, Automated tools in phylogenetic analysis. 9 Hours

UNIT-IV

Tools of Similarity Search and Sequence Alignment: introduction, working with Fasta, working with Blast, Filtering and Gapped Blast, Fasta and Blast algorithms comparison, other programs.

Gene Identification and Prediction: introduction, basis of gene prediction, pattern recognition, gene prediction methods, other gene prediction tools. 9 Hours

UNIT -V

Protein Classification And Structure Visualization: introduction, overview of the protein structure, protein structure visualization, structure-based protein classification, protein structure databases, protein structure visualization databases and tools, protein structure alignment, domain architecture databases, tools for plotting protein-ligand interaction, protein classification approaches.

Protein structure prediction: introduction, protein identification and characterization, primary structure analysis and prediction, secondary structure analysis and prediction, motifs, profiles, patterns and fingerprint search, methods of sequence bases protein prediction, methods of 2-D structure prediction, protein function prediction. 9 Hours

Textbook:

1. Bioinformatics : Methods and Applications, 4th Edition, S.C. Rastogi, PHI Learning, 2013.

References:

1. Bioinformatics: A practical guide to the analysis of Genes and Proteins, Andreas D. Baxevanis, B.F. Francis Ouellette, 3rd edition, Wiley,2005.
2. Introduction to Bioinformatics, Teresa Attwood, 2007.
3. Bioinformatics for Beginners: Supratim Choudhuri, Elsevier,

Course Outcomes

After completing the course student will be able to:

- **CO1:** Understand the foundational concepts of bioinformatics, including databases, data management, and molecular biology's role in bioinformatics. (**L2: Understand**)

- **CO2:** Apply sequence alignment techniques and genome analysis methods, including gene mapping and multiple sequence alignment, to solve biological problems. (**L3: Apply**)

- **CO3:** Analyse phylogenetic relationships using various phylogenetic analysis methods, tools, and tree evaluation techniques. (**L4: Analyse**)

- **CO4:** Evaluate tools and algorithms such as Fasta and Blast for similarity search and sequence alignment, and assess methods for gene prediction. (**L5: Evaluate**)

- **CO5:** Develop protein structure prediction and visualization approaches, leveraging databases and computational tools for functional and structural analysis. (**L3: Develop**)

CO-PO Mapping table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	3	3	2	0	0	3
CO2	3	3	3	3	2	0	0	3
CO3	3	3	3	3	2	0	0	3
CO4	3	3	3	3	2	0	0	3
CO5	3	3	3	3	2	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handle necessary mathematical orientation, programming techniques and concept based learning.	

INTRODUCTION TO NOSQL DATABASE

L	T	P	S	J	C
3	0	0	0	0	3

Course Description:

This course provides a comprehensive introduction to NoSQL databases, exploring their history, types, and specific technologies including MongoDB, Cassandra, HBase, Riak, and Neo4j. Students will learn about the differences between NoSQL databases and traditional relational database systems (RDBMS), understanding the scenarios in which NoSQL databases excel due to their schema flexibility, scalability, and performance characteristics. The curriculum covers the fundamental concepts of NoSQL databases, including key-value stores, document databases, column-family stores, and graph databases. Each type will be examined in detail with regard to its architecture, data model, consistency, transactions, availability, scaling, and suitable use cases. Practical applications such as event logging, content management systems, e-commerce, and real-time analytics will be explored.

Course Objectives:

- To understand the Evolution and Types of NoSQL Databases
- To compare and Contrast NoSQL and RDBMS
- To master Specific NoSQL Technologies
- To develop Applications Using NoSQL Databases
- To explore Advanced NoSQL Features

UNIT-I

Overview and History of NoSQL Databases. Definition of the Four Types of NoSQL Database, The Value of Relational Databases, Getting at Persistent Data, Concurrency, Integration, Impedance Mismatch, Application and Integration Databases, Attack of the Clusters, The Emergence of NoSQL, Key Points. 9 Hours

UNIT-II

Comparison of relational databases to new NoSQL stores, MongoDB, Cassandra, HBASE, Neo4j use and deployment, Application, RDBMS approach, Challenges NoSQL approach, Key-Value and Document Data Models, Column-Family Stores, Aggregate-Oriented Databases. Replication and sharding, MapReduce on databases. Distribution Models, Single Server, Sharding, Master-Slave Replication, Peer-to-Peer Replication, Combining Sharding and Replication. 9 Hours

UNIT-III

NoSQL Key/Value databases using MongoDB, Document Databases, Document oriented Database Features, Consistency, Transactions, Availability, Query Features, Scaling, Suitable Use Cases, Event Logging, Content Management Systems, Blogging Platforms, Web Analytics or Real-Time Analytics, E-Commerce Applications, Complex Transactions Spanning Different Operations, Queries against Varying Aggregate Structure **9** Hours

UNIT-IV

Column- oriented NoSQL databases using Apache HBASE, Column-oriented NoSQL databases using Apache Cassandra, Architecture of HBASE, Column-Family Data Store Features, Consistency, Transactions, Availability, Query Features, Scaling, Suitable Use Cases, Event Logging, Content Management Systems, Blogging Platforms, Counters, Expiring Usage. 9 Hours

UNIT-V

NoSQL Key/Value databases using Riak, Key-Value Databases,Key-Value Store, Key- Value Store Features, Consistency, Transactions, Query Features, Structure of Data, Scaling, Suitable Use Cases, Storing Session Information, User Profiles, Preferences, Shopping

Cart Data, Relationships among Data, Multi operation Transactions, Query by Data, Operations by Sets. Graph NoSQL databases using Neo4,NoSQL database development tools and programming languages, Graph Databases, Graph Database. Features, Consistency, Transactions, Availability, Query Features, Scaling, Suitable Use Cases. 9 Hours

TEXT BOOKS: 1. Sadalage, P. & Fowler, NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence, Wiley Publications,1st Edition ,2019.

WEB REFERENCES:

1. https://www.ibm.com/cloud/learn/nosql-databases
2. https://www.coursera.org/lecture/nosql-databases/introduction-to-nosql-VdRNp
3. https://www.geeksforgeeks.org/introduction-to-nosql/
4. https://www.javatpoint.com/nosql-databa

Course Outcomes

After completing the course student will be able to:

- **CO1:** Explain the history, emergence, and key concepts of NoSQL databases, including their types and applications in data management. *(L2: Understand)*
- **CO2:** Compare relational databases with NoSQL databases and analyze different data models such as key-value, document, and column-family stores, including their replication and sharding mechanisms. *(L4: Analyze)*
- **CO3:** Apply NoSQL techniques using MongoDB to design and implement solutions for scenarios like event logging, content management, and e-commerce applications.*(L3: Apply)*

- **CO4:** Evaluate column-oriented NoSQL databases such as Apache HBASE and Cassandra for their architecture, features, and suitable use cases in data-intensive applications.*(L5: Evaluate)*
- **CO5:** Develop solutions using graph and key-value NoSQL databases, such as Neo4j and Riak, for applications requiring complex data relationships and multi-operation transactions. *(L3: Develop)*

CO-PO Mapping table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	3	3	2	0	0	3
CO2	3	3	3	3	2	0	0	3
CO3	3	3	3	3	2	0	0	3
CO4	3	3	3	3	2	0	0	3
CO5	3	3	3	3	2	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handle necessary mathematical orientation, programming techniques and concept-based learning.	

Prepared by Dr.M.Srivenkatesh Verified by Dr.M.Srivenkatesh

SOFTWARE PROCESS MANAGEMENT

L	T	P	S	J	C
3	0	0	0	0	3

Course Description

This course provides an in-depth understanding of software process management, covering the lifecycle of software development processes, project management, and process improvement methodologies. Students will learn about different software process models, the principles of agile and lean methodologies, and advanced topics in process management. Emphasis is placed on practical tools and techniques for managing software projects, improving process quality, and ensuring successful software delivery. Through a combination of theoretical knowledge and hands-on lab exercises, students will be equipped with the skills necessary to manage software processes effectively in a variety of organizational contexts.

Course Objectives

- To understand Software Process Models and Standards
- To develop Project Management Skills
- To implement Agile and Lean Methodologies
- To apply Process Improvement Techniques
- To utilize Process Management Tools

UNIT-I

Introduction to Software Process Management

Software Process Fundamentals: Definition and importance, Software process models (Waterfall, Incremental, Spiral, V-Model, Agile)

Software Development Life Cycle (SDLC):Phases of SDLC,Role of process management in SDLC

Process Management Standards and Models: ISO/IEC 12207,Capability Maturity Model Integration (CMMI),ISO 9001

Process Improvement: Goals and benefits, Process assessment and improvement methodologies (Six Sigma, Lean, TQM)

Process Metrics and Measurement: Types of metrics (product, process, project), Measurement techniques, Key Process Indicators (KPIs)

UNIT-II

Software Project Management

Project Planning and Estimation: Project scope and objectives, Work Breakdown Structure (WBS), Estimation techniques (COCOMO, Function Point Analysis)

Resource Management: Resource planning and allocation, Team organization and roles

Managing cross-functional teams

Risk Management: Risk identification, analysis, and mitigation, Risk management strategies

Contingency planning

Scheduling and Tracking: Gantt charts and PERT diagrams, Milestones and deliverables

Earned Value Management (EVM)

Quality Management: Quality assurance and control, Software testing strategies, Defect management and tracking

UNIT -III

Agile and Lean Methodologies

Agile Principles and Practices: Agile Manifesto, Scrum framework (roles, artifacts, ceremonies),Extreme Programming (XP)

Lean Software Development: Lean principles, Value stream mapping, Waste elimination techniques

Scaling Agile: Scaled Agile Framework (SAFe),Large Scale Scrum (LeSS),Disciplined Agile Delivery (DAD)

Agile Metrics and Tools: Velocity, burn-down, and burn-up charts,Tools for agile project management (JIRA, Trello)

Continuous Integration and Delivery: I/CD pipeline, DevOps practices, Tools for CI/CD (Jenkins, GitLab CI/CD)

UNIT-IV

Process Improvement and Maturity Models

Process Maturity Models: CMMI Levels and Process Areas, ISO/IEC 15504 (SPICE)

Process Improvement Initiatives: Initiating process improvement programs, Defining process improvement goals and metrics, Benchmarking and best practices

Process Reengineering: Principles of business process reengineering, Process redesign techniques, Case studies and examples

Change Management: Managing resistance to change, Strategies for successful change implementation, Communication and training

Software Process Tools:Tools for process modeling and simulation (e.g., Bizagi, ARIS)

Process automation tools,Case studies of tool implementation

UNIT -V

Advanced Topics and Emerging Trends

Global Software Development: Challenges and strategies for distributed teams Communication and collaboration tools

Process Management in Emerging Technologies

Software processes for AI/ML projects, Agile in the context of DevOps and Cloud Computing

Managing processes in IoT and Big Data projects

Legal and Ethical Considerations: Intellectual property rights, Data privacy and security regulations, Ethical issues in software development

Future Trends in Software Process Management: Emerging frameworks and methodologies

The impact of AI and automation on process management, Trends in software quality assurance and testing

Case Studies and Industry Best Practices

Analysis of successful process management implementations, Lessons learned from industry leaders, Continuous learning and improvement in process management

Text Books:

1. Software Engineering: A Practitioner's Approach by Roger S. Pressman

McGraw-Hill Education, 8th Edition, 2014

2. Software Process Improvement and Management: Approaches and Tools for Practical Development" by Robin B. Hunter, IGI Global,1st Edition, 2011

3. Agile Project Management with Scrum" by Ken Schwaber, Microsoft Press, 1st Edition, 2004.

4. Managing the Software Process" by Watts S. Humphrey, Addison-Wesley Professional ,1st Edition, 1989

5. Software Project Management in Practice" by Pankaj Jalote,Addison-Wesley Professional 1st Edition, 2002

Reference Text Books

1. Software Metrics: A Rigorous and Practical Approach,by Norman Fenton and James Bieman, CRC Press,3rd Edition, 2014

2. Succeeding with Agile: Software Development Using Scrum by Mike Cohn
Addison-Wesley Professional,1st Edition, 2009

Course Outcomes

After completing the course student will be able to:

- **CO1:** Explain the fundamentals of software process management, various software process models, and their relevance to SDLC, including standards like ISO and CMMI. *(L2: Understand)*
- **CO2:** Apply project management techniques, including estimation, scheduling, resource allocation, risk management, and quality assurance to software projects.*(L3: Apply)*
- **CO3:** Analyse the principles of Agile and Lean methodologies, scaling frameworks, and CI/CD practices for improving software development processes.*(L4: Analyse)*
- **CO4:** Evaluate process improvement initiatives, maturity models like CMMI, and reengineering techniques for enhancing software process efficiency and effectiveness.*(L5: Evaluate)*
- **CO5:** Develop strategies for managing global software development, addressing legal and ethical considerations, and incorporating emerging trends and technologies into software process management. *(L3: Develop)*

- **Co-PO Mapping table**

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	3	1	1	0	0	3
CO2	3	3	3	2	3	0	0	3
CO3	3	3	3	3	3	0	0	3
CO4	2	3	3	2	3	0	0	3
CO5	3	3	2	3	2	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handle necessary mathematical orientation, programming techniques and concept based learning.	

SOFTWARE TESTING AND QUALITY ASSURANCE

L	T	P	S	J	C
3	0	0	0	0	3

Course Description

This course provides an in-depth understanding of software testing and quality assurance (QA) practices essential for ensuring the reliability and performance of software applications. Students will learn various testing techniques, tools, and methodologies to identify and address defects throughout the software development life cycle (SDLC). The course covers both functional and non-functional testing, test management, quality assurance standards, and advanced topics like Agile testing, DevOps, and AI in testing. Through hands-on lab exercises and real-world examples, students will develop practical skills to implement effective testing strategies and contribute to software quality improvement initiatives.

Course Objectives

- To understand Fundamental Testing Concepts
- To perform Functional and Non-Functional Testing
- To manage and Control Testing Processes
- To implement Quality Assurance Standards and Practices
- To explore Advanced Testing Techniques and Trends

UNIT-I

Introduction to Software Testing

Fundamentals of Testing: Definition, importance, and objectives of testing.

Software Development Life Cycle (SDLC) and Testing: Role of testing in SDLC, V-Model,

and Agile Model.

Types of Testing: Manual testing vs. automated testing, static testing vs. dynamic testing.

Levels of Testing: Unit testing, integration testing, system testing, acceptance testing.

Testing Techniques: Black-box testing, white-box testing, and grey-box testing.

Test Planning and Documentation: Test plan, test cases, test scripts, and test data. 9 Hours

UNIT-II

Functional and Non-Functional Testing

Functional Testing: Overview, techniques, and tools.

Non-Functional Testing: Performance testing, load testing, stress testing, usability testing, security testing.

Regression Testing: Purpose, techniques, and tools.

Compatibility Testing: Testing across different browsers, devices, and operating systems.

Testing for Specific Applications: Web application testing, mobile application testing, and database testing.

Automation Testing Tools: Selenium, QTP, Test Complete, etc. 9 Hours

UNIT-III

Test Management and Control

Test Management: Test organization, test team roles and responsibilities.

Test Metrics and Measurements: Test coverage, defect density, defect detection rate.

Test Reporting: Test summary report, defect report, metrics report.

Defect Management: Defect life cycle, defect tracking tools (JIRA, Bugzilla).

Configuration Management: Version control, release management.

Risk Management: Identifying and managing risks in testing. 9 Hours

UNIT-IV

Quality Assurance and Standards

Quality Assurance (QA): QA activities, difference between QA and QC.

Quality Models and Standards: ISO 9001, CMMI, Six Sigma.

Software Quality Attributes: Reliability, maintainability, portability, usability

Quality Metrics: Product metrics, process metrics, project metrics.

Reviews and Inspections: Code reviews, design reviews, walkthroughs, inspections.

Software Process Improvement: Introduction to SPI, TQM, PDCA cycle. 9 Hours

UNIT-V

Advanced Topics in Testing and QA

Agile Testing: Agile testing principles, test-driven development (TDD), behavior-driven development (BDD).

Continuous Integration and Continuous Testing: Concepts, tools (Jenkins, Bamboo), best practices.

DevOps and Testing: Role of testing in DevOps, test automation in CI/CD pipeline.

AI and Machine Learning in Testing: Applications, tools, and techniques.

Test Environment and Data Management: Setting up test environments, managing test data.

Future Trends in Testing and QA: Emerging trends, challenges, and opportunities. 9 Hours

Textbooks

1. Foundations of Software Testing: ISTQB Certification, by Dorothy Graham, Rex Black, Erik van Veenendaal Cengage Learning, 4th Edition, 2021
2. Software Testing: Principles and Practices by Srinivasan Desikan, Gopalaswamy Ramesh,Publisher: Pearson Education,1st Edition, 2006
3. Software Testing and Quality Assurance: Theory and Practice by Kshirasagar Naik, Priyadarshi Tripathy, Wiley,1st Edition, 2008

Recommended Text Books

1. Agile Testing: A Practical Guide for Testers and Agile Teams by Lisa Crispin, Janet Gregory, Addison-Wesley Professional,1st Edition, 2009

2 ."Continuous Testing for DevOps Professionals" by Eran Kinsbruner,Publisher: Packt Publishing,Edition: 1st Edition, 2018

Course Outcomes

After completing the course student will be able to:

- **CO1:** Explain the fundamentals of software testing, testing techniques, and their integration into various software development life cycle (SDLC) models.*(L2: Understand)*

- **CO2:** Differentiate between functional and non-functional testing approaches, and apply appropriate tools and techniques for various testing scenarios, including automation.*(L4: Analyse)*

- **CO3:** Manage and control the testing process by applying test metrics, defect management practices, and risk management techniques using tools like JIRA or Bugzilla. *(L3: Apply)*

- **CO4:** Evaluate software quality attributes, quality assurance standards, and improvement processes such as ISO 9001, CMMI, and Six Sigma to ensure reliability and maintainability. *(L5: Evaluate)*

- **CO5:** Develop testing strategies using Agile, CI/CD pipelines, and emerging technologies such as AI and machine learning to address future challenges in testing and quality assurance. *(L3: Develop)*

CO-PO Mapping table

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1	3	3	2	2	1	0	0	3
CO2	2	3	3	3	2	0	0	3
CO3	2	2	2	3	3	0	0	2
CO4	2	3	3	3	2	0	0	3
CO5	3	3	3	3	3	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handle necessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Dr.M.Srivenkatesh Verified by Dr.M.Srivenkatesh

UNIVERSAL HUMAN VALUES 2: UNDERSTANDING HARMONY (BHUT-401)

L	T	P	S	J	C
3	0	0	0	0	3

Course Description

This course also discusses their role in their family. It, very briefly, touches issues related to their role in the society and the nature, which needs to be discussed at length in one more semester for which the foundation course named as "H-102 Universal Human Values 2: "Understanding Harmony" is composed of practical work, which may include societal work or involvement of Faculty and Students

1. OBJECTIVE:

The objective of the course is four fold:

1. Development of a holistic perspective based on self-exploration about themselves (human being), family, society and nature/existence.
2. Understanding (or developing clarity) of the harmony in the human being, family, society and nature/existence
3. Strengthening of self-reflection.
4. Development of commitment and courage to act.

2. COURSE TOPICS:

The course has 28 lectures and 14 practice sessions in 5 modules:

Module 1: Course Introduction - Need, Basic Guidelines, Content and Process for Value Education

1. Purpose and motivation for the course, recapitulation from Universal Human Values-I

2. Self-Exploration–what is it? - Its content and process; 'Natural Acceptance' and experiential Validation- as the process for self-exploration

3. Continuous Happiness and Prosperity- A look at basic Human Aspirations 4. Right understanding, Relationship and Physical Facility- the basic requirements for fulfilment of aspirations of every human being with their correct priority

4. Understanding Happiness and Prosperity correctly- A critical appraisal of the current scenario

5. Method to fulfil the above human aspirations: understanding and living in harmony at various levels.

Include practice sessions to discuss natural acceptance in human being as the innate acceptance for living with responsibility (living in relationship, harmony and co-existence) rather than as arbitrariness in choice based on liking-disliking

Module 2: Understanding Harmony in the Human Being - Harmony in Myself!

1. Understanding human being as a co-existence of the sentient 'I' and the material 'Body'
2. Understanding the needs of Self ('I') and 'Body' - happiness and physical facility
3. Understanding the Body as an instrument of 'I' (I being the doer, seer and enjoyer)
4. Understanding the characteristics and activities of 'I' and harmony in 'I'
5. Understanding the harmony of I with the Body: Sanyam and Health; correct appraisal of Physical needs, meaning of Prosperity in detail
6. Programs to ensure Sanyam and Health.

Include practice sessions to discuss the role others have played in making material goods available to me. Identifying from one's own life. Differentiate between prosperity and accumulation. Discuss program for ensuring health vs dealing with disease

Module 3: Understanding Harmony in the Family and Society- Harmony in Human-Human Relationship

1. Understanding values in human-human relationship; meaning of Justice (nine universal values in relationships) and program for its fulfilment to ensure mutual happiness; Trust and Respect as the foundational values of relationship 14. Understanding the meaning of Trust; Difference between intention and competence
2. Understanding the meaning of Respect, Difference between respect and differentiation; the other salient values in relationship
3. Understanding the harmony in the society (society being an extension of family): Resolution, Prosperity, fearlessness (trust) and co-existence as comprehensive Human Goals
4. Visualizing a universal harmonious order in society- Undivided Society, Universal Order- from family to world family.

Include practice sessions to reflect on relationships in family, hostel and institute as extended family, real life examples, teacher-student relationship, goal of education etc. Gratitude as a universal value in relationships. Discuss with scenarios. Elicit examples from students' lives

Module 4: Understanding Harmony in the Nature and Existence - Whole existence as Coexistence

1. Understanding the harmony in the Nature
2. Interconnectedness and mutual fulfilment among the four orders of nature- recyclability and self regulation in nature
3. Understanding Existence as Co-existence of mutually interacting units in allpervasive space
4. Holistic perception of harmony at all levels of existence.

Include practice sessions to discuss human being as cause of imbalance in nature (film "Home" can be used), pollution, depletion of resources and role of technology etc.

Module 5: Implications of the above Holistic Understanding of Harmony on Professional Ethics

1. Natural acceptance of human values
2. Definitiveness of Ethical Human Conduct
3. Basis for Humanistic Education, Humanistic Constitution and Humanistic Universal Order
4. Competence in professional ethics: a. Ability to utilize the professional competence for augmenting universal human order b. Ability to identify the scope and characteristics of people friendly and eco-friendly production systems, c. Ability to identify and develop appropriate technologies and management patterns for above production systems.
5. Case studies of typical holistic technologies, management models and production systems
6. Strategy for transition from the present state to Universal Human Order: a. At the level of individual: as socially and ecologically responsible engineers, technologists and managers b. At the level of society: as mutually enriching institutions and organizations

7. Sum up.

Include practice Exercises and Case Studies will be taken up in Practice (tutorial) Sessions eg. To discuss the conduct as an engineer or scientist etc.

3. READINGS

Text Book

1. Human Values and Professional Ethics by R R Gaur, R Sangal, G P Bagaria, Excel Books, New Delhi, 2010

Reference Books

1. Jeevan Vidya: Ek Parichaya, A Nagaraj, Jeevan Vidya Prakashan, Amarkantak, 1999.
2. Human Values, A.N. Tripathi, New Age Intl. Publishers, New Delhi, 2004.
3. The Story of Stuff (Book).
4. The Story of My Experiments with Truth - by Mohandas Karamchand Gandhi
5. Small is Beautiful - E. F Schumacher.
6. Slow is Beautiful - Cecile Andrews
7. Economy of Permanence - J C Kumarappa
8. Bharat Mein Angreji Raj - PanditSunderlal
9. Rediscovering India - by Dharampal
10. Hind Swaraj or Indian Home Rule - by Mohandas K. Gandhi
11. India Wins Freedom - Maulana Abdul Kalam Azad
12. Vivekananda - Romain Rolland (English)
13. Gandhi - Romain Rolland (English)

4. MODE OF CONDUCT (L-T-P-C 2-1-0-3 or 2L:1T:0P 3 credits)

Lectures hours are to be used for interactive discussion, placing the proposals about the topics at hand and motivating students to reflect, explore and verify them.
Tutorial hours are to be used for practice sessions.

While analysing and discussing the topic, the faculty mentor's role is in pointing to essential elements to help in sorting them out from the surface elements. In other words, help the students explore the important or critical elements. In the discussions, particularly during practice sessions (tutorials), the mentor encourages the student to connect with one's own self and do self-observation, self-reflection and self-exploration. Scenarios may be used to initiate discussion. The student is encouraged to take up "ordinary" situations rather than" extra-ordinary" situations. Such observations and their analyses are shared and discussed with other students and faculty mentor, in a group sitting.

Tutorials (experiments or practical) are important for the course. The difference is that the laboratory is everyday life, and practical are how you behave and work in real life. Depending on the nature of topics, worksheets, home assignment and/or activity are included. The practice sessions (tutorials) would also provide support to a student in performing actions commensurate to his/her beliefs. It is intended that this would lead to development of commitment, namely behaving and working based on basic human values.

It is recommended that this content be placed before the student as it is, in the form of a basic foundation course, without including anything else or excluding any part of this content. Additional content may be offered in separate, higher courses.

This course is to be taught by faculty from every teaching department, including HSS faculty.
Teacher preparation with a minimum exposure to at least one 8-day FDP on Universal Human Values is deemed essential.

5. ASSESSMENT:

This is a compulsory credit course. The assessment is to provide a fair state of development of the student, so participation in classroom discussions, selfassessment, peer assessment etc. will be used in evaluation.
Example:
Assessment by faculty mentor: 10 marks
Self-assessment: 10 marks
Assessment by peers: 10 marks
Socially relevant project/Group Activities/Assignments: 20 marks
Semester End Examination: 50 marks
The overall pass percentage is 40%. In case the student fails, he/she must repeat the course.

6. OUTCOME OF THE COURSE:

By the end of the course, students are expected to become more aware of themselves, and their surroundings (family, society, nature); they would become more responsible in life, and in handling problems with sustainable solutions, while keeping human relationships and human nature in mind. They would have better critical ability. They would also become sensitive to their commitment towards what they have understood (human values, human relationship and human society). It is hoped that they would be able to apply what they have learnt to their own self in different day-to-day settings in real life, at least a beginning would be made in this direction.

Note: it may be followed by faculty-student or mentor-mentee programs throughout their time with the institution. The HV Workshop (5/8 days) is compulsory for faculty taking this course

BIG DATA INSIGHTS LAB

L	T	P	S	J	C
0	0	2	0	0	1

Course Description:

This laboratory course is designed to provide practical, hands-on experience with big data technologies, focusing on the tools and methodologies used to handle, process, and analyze vast amounts of data. Students will work with Hadoop, HBase, Hive, Pig, Sqoop, Mahout, and Apache Spark, which are among the most prominent technologies in the field of big data analytics. Through a series of structured lab exercises, students will explore the Hadoop Distributed File System (HDFS), implement MapReduce jobs, perform data manipulations using HBase, and carry out data importation tasks with Sqoop. The course also covers the use of Hive for data warehousing tasks, Mahout for building recommendation systems, and Pig for data flow scripting. Additionally, students will gain exposure to Apache Spark for in-memory data processing to perform complex analytics on distributed datasets.

Course Objectives:

- Master the Use of Hadoop and HDFS
- Develop Proficiency in Writing and Managing MapReduce Jobs
- Utilize HBase for NoSQL Data Management
- Perform Data Integration and Analysis Using Hive and Pig
- Explore Data Science Techniques with Mahout and Spark

Lab Exercise Programs

1. Exploring Hadoop Distributed File System (HDFS). Implementation of file system commands in HDFS.
2. Understanding Map Reduce Jobs:
   a. Writing a MapReduce Program
   b. Running a MapReduce Job.
   c. Writing and Implementing a Combiner
   d. Writing a Partitioner
3. Using the HBase Shell perform basic table functions view the results of each operation:
   a. Creating a table,
   b. Putting rows into table
   c. Scanning a table
   d. Manually flush and compact a table
4. Write an HBase program that creates a table, puts several rows, scans the table and outputs the column values.
5. Importing Data with Sqoop: import data from a relational database using Sqoop.
6. Creating a Mahout Recommender: Use of Mahout to generate recommendations for users based on the data imported using Sqoop.
7. Loading Data into Hive:
   a. Using the dump file present in the HDFS file system.
   b. Using Sqoop to import the table into Hive from MySQL.

8. Executing Hive Queries.
9. Partitioning and Bucketing Data in Hive.
10. Managing an HBase Table with Hive: Create a table in HBase and then create an External Table in Hive which points to the HBase table.
11. Reading and Writing Data using Pig using Grunt Shell and perform the following operations:
    a. DESCRIBE the table.
    b. Use DUMP to print the records you loaded.
    c. Use STORE to save the records in a file. The resulting file should be comma-delimited.
    d. Using a terminal or the Grunt shell, inspect the results.
12. Writing PigLatin Scripts for the following tasks:
    a. Write a Pig Latin program to read u.user and find the users who are female and scientists. Store the results in a file.
    b. What is the average age of the users in u.user? Hint: use the AVG function. Dump the result to the screen.
    c. Create a script using a text editor. It should load the u.user file, find the 3 most common occupations and store the results in HDFS. Execute the script using "pig -f script"
13. Write your Pig script. It should work on all the files in a directory and return a list of the 20 most common words in the directory. Hint: the TOKENIZE function will be useful
14. Understanding Spark Shell-
    a. Create a Spark Session,
    b. Inspect Spark Session using following Commands-
        i. Retrieve Spark Session Version,
        ii. Return Spark Application name, Retrieve Spark Application Id,
        iii. Check and Return Minimum Number of Partitions,
15. Create an RDD which contains key value pairs using parallelize method and Perform the following operations on RDD using pyspark:
    a. List number of partitions RDD consists,
    b. Count the instances of RDD,
    c. Count the instances of RDD by key,
    d. Count the instances of RDD by value,
    e. Calculate Sum of RDD elements,
    f. Check whether RDD is empty or not.
16. Create an RDD which contains range of 100 numbers and perform the following operations on RDD using pyspark-
    a. Find the minimum value in the RDD elements
    b. Find the maximum value in the RDD elements
    c. Find the Mean, Median & standard deviation of RDD elements
17. Word Count using RDD- Write a pyspark program for the following
    a. Create a Spark Session
    b. Read the text file using RDD API
    c. Perform transformations (Map & flat Map) on the text file to countthe number of words are there in the file.
    d. Save the outcome as a file.

Text Books:

1. Big Data Black Book by Dt Editorial Services, Dreamtech Publications,2016.
2. Hadoop The Definitive Guide by Tom White, O'reilly ,4th Edition,2016.
3. Programming Hive- Jason Rutherglen, Dean Wampler, Edward Capriolo,O'reilly

Publisher, 1st edition,2012.

Course Outcomes:

After completing the course student will be able to:

CO1: Demonstrate an understanding of Hadoop Distributed File System (HDFS), MapReduce, and their functionalities through practical implementations. *(L3: Apply)*

CO2: Utilize tools like HBase, Sqoop, and Hive for performing database operations, managing structured and semi-structured data, and integrating them with other Big Data ecosystems. *(L4: Analyze)*

CO3: Implement data transformations, aggregations, and advanced querying techniques using Hive and Pig scripting for efficient data processing and analysis.*(L5: Evaluate)*

CO4: Apply PySpark to create, manipulate, and analyze Resilient Distributed Datasets (RDDs) for large-scale data processing, including operations like word counts, statistical computations, and key-value operations. *(L3: Apply)*

CO5: Develop scalable data processing workflows using Spark, focusing on interactive data exploration, session creation, and deploying machine learning recommendations with Mahout. L3: Develop

CO-PO Mapping table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	3	3	2	0	0	3
CO2	3	3	3	3	2	0	0	3
CO3	3	3	3	3	2	0	0	3
CO4	3	3	3	3	2	0	0	3
CO5	3	3	3	3	2	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handlenecessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Mr.Shahid Shaik Verified by Dr.M.Srivenkatesh

CYBER SECURITY LAB

L	T	P	S	J	C
0	0	2	0	0	1

Course Description:

This laboratory course provides students with practical, hands-on experience in implementing key cybersecurity techniques and protocols. Through a series of exercises, students will learn to apply cryptographic algorithms, digital signatures, and secure communication protocols. The lab sessions focus on encoding and encryption methods including Base64, Triple DES, AES, and RSA, along with the application of digital signatures using SHA-256 and RSA.Students will also engage in creating a sample Certifying Authority (CA) to sign and encrypt data and manage SSL certificates for web servers. The course includes programming tasks primarily in Java, aimed at building a robust understanding of both symmetric and asymmetric encryption techniques, and the use of Java tools for certificate management and secure web communications.

Course Objectives:

- To understand and Implement Encoding and Encryption Techniques
- To develop Skills in Digital Signing and Verification
- To create and Manage Certifying Authorities
- To secure Web Communications
- To utilize Java Tools for Cybersecurity Applications

Lab Exercise Programs

1 Implement base 64 encoding and decoding of a sample text
2 Implement encryption and decryption of a sample text using Triple DES and AES 256
3. Implement encryption and decryption of a sample text using RSA 1024 and 2048 keys.
4. Implement Digital signing and verification using SHA 256 and RSA 1024/2048
5. Implement Sample Certifying authority(CA) for signing and encryption and issue server certificate with CA

6. Implement SSL enabling of web server using SSL certificate issued by CA created earlier (Step 5)

7. Implement a java program to implement base64 encoding and decoding of a sample text
8. Implement a Java program to implement Symmetric encryption and decryption of a sample text using AES algorithm.

9. Implement a Java program to implement Asymmetric encryption of sample text using the RSA algorithm
10. Implement a Java program to digitally sign and verify the sample text. Exercise
11. Create root CA using Java key tool and issue certificate with the root CA created. Exercise
12. Create SSL certificate using JAVA Key tool and SSL enable the web server using the SSL certificate issued and access the website via https

Text Books

Cryptography and Network Security: Principles and Practice" by William Stallings, 8th Edition, Pearson

Java Cryptography: Tools and Techniques" by David Hook, 1st Edition, Wiley

Java Network Programming by Elliotte Rusty Harold, 4th Edition, O'Reilly Media

Course Outcomes:

After completing the course student will be able to:

Course Outcomes for the Listed Exercises

CO1: Demonstrate proficiency in implementing base64 encoding and decoding for secure data representation using Python and Java.*(L3: Apply)*

CO2: Apply symmetric encryption algorithms like AES and Triple DES and asymmetric encryption algorithms like RSA to secure text data through encryption and decryption techniques. *(L3: Apply)*

CO3: Evaluate solutions for digital signing and verification of data using SHA-256 and RSA, ensuring message integrity and authenticity.*(L5: Evaluate)*

CO4: Develop a sample Certificate Authority (CA) to issue SSL certificates and integrate SSL/TLS protocols for secure communication over web servers.*(L3: Develop)*

CO5: Analyse Java KeyTool for creating root Certificate Authorities (CA), issuing certificates, and enabling SSL on web servers to establish secure HTTPS connections.*(L4: Analyse)*

CO-PO Mapping table

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**
CO1	3	3	3	3	2	0	0	3
CO2	3	3	3	3	2	0	0	3
CO3	3	3	3	3	2	0	0	3
CO4	3	3	3	3	2	0	0	3
CO5	3	3	3	3	2	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handle necessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Dr.V.Nagalakshmi Verified by Dr.M.Srivenkatesh

BLOCKCHAIN TECHNOLOGIES LAB

L	T	P	S	J	C
0	0	2	0	0	1

Course Description:

This laboratory course provides students with practical, hands-on experience in the fundamental and advanced concepts of blockchain technology. Through a series of programming exercises, students will explore the creation and operation of blockchain systems, including the development of smart contracts, decentralized applications (DApps), and consensus mechanisms. The course focuses on using popular blockchain development tools such as Solidity and the Truffle Suite to build and deploy applications on the Ethereum blockchain.Students will start with basic blockchain simulations to understand the core operations and data structures, such as blocks and Merkle trees. They will then progress to more complex tasks, including writing and deploying smart contracts, developing DApps, and exploring different blockchain consensus mechanisms like Proof of Work and Proof of Stake. The course aims to provide a comprehensive understanding of how blockchain can be implemented in real-world applications through coding and development.

Course Objectives:

- Understand Blockchain Fundamentals
- Develop and Simulate Blockchain Operations
- Explore Consensus Mechanisms
- Write and Deploy Smart Contracts
- Create Decentralized Applications (DApps)

Lab Exercise Programs

1. Write a program the basic operations of blockchain through a simple simulation tool.

2. Write a program to simulate different consensus mechanisms used in blockchain networks.

3. Write a program to understand the fundamental concepts of smart contracts and their role in decentralized applications.

4. Write a program to perform the basics of Solidity syntax and structure.

5. Write a program to develop and deploy smart contracts on the Ethereum blockchain.

6. Write a program to develop a decentralized application using the Truffle Suite.

7. Write a program to create Markle Tree

8. Write a program to create block in blockchain

9. Write a program to create blockchain

1 0 .Write a program to implement blockchain in Merkle Trees.

11.Write a Program to implement Mining using block chain

Text Books

1. "Mastering Blockchain: Unlocking the Power of Cryptocurrencies, Smart Contracts, and Decentralized Applications" by Imran Bashir,3rd Edition, Packt Publishing

2."Mastering Ethereum: Building Smart Contracts and DApps" by Andreas M. Antonopoulos and Gavin Wood,1st Edition, O'Reilly Media, 2018.

3."Solidity Programming Essentials: A Beginner's Guide to Build Smart Contracts for Ethereum

and Blockchain" by Ritesh Modi, 2nd Edition, Packt Publishing

4."Building Ethereum Dapps: Decentralized Applications on the Ethereum Blockchain" by Roberto Infante,1st Edition,Manning Publications

Course Outcomes:

After completing the course student will be able to:

- **CO1:** Understand and simulate the basic operations of blockchain, including block creation, linkage, and verification, through programming and simulation tools.*(L2: Understand)*

- **CO2:** Analyse and implement various consensus mechanisms, such as Proof of Work and Proof of Stake, to achieve distributed agreement in blockchain networks.*(L4:Analyze)*

- **CO3:** Develop smart contracts using Solidity and deploy them on blockchain platforms like Ethereum to enable decentralized applications. *(L3: Develop)*

- **CO4:** Implement cryptographic structures, such as Merkle Trees, and integrate them into blockchain systems to enhance data integrity and validation processes. *(L6: Develop)*

- **CO5:** Develop a basic blockchain system, including block mining and transaction validation, to understand its practical applications and scalability challenges. *(L3: Develop)*

CO-PO Mapping table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	3	3	2	1	0	3
CO2	3	3	3	3	2	1	0	3
CO3	3	3	3	3	2	1	0	3
CO4	3	3	3	3	2	1	0	3
CO5	3	3	3	3	2	1	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handle necessary mathematical orientation, programming techniques and concept-based learning.	

DEEP LEARNING TECHINQUES LAB

L	T	P	S	J	C
0	0	2	0	0	1

Course Description:

This laboratory course is designed to provide students with practical, hands-on experience in deep learning, focusing on the development and implementation of various neural network architectures. Through a series of programming exercises, students will explore the fundamentals of neural networks, including perceptrons, multilayer perceptrons (MLPs), convolutional neural networks (CNNs), recurrent neural networks (RNNs), and autoencoders.The course will cover key deep learning techniques such as gradient descent, backpropagation, regularization, and optimization algorithms. Students will work with popular deep learning frameworks to implement and train models on real-world datasets like MNIST and CIFAR-10, and they will learn to fine-tune pre-trained models for tasks like image classification and sequence prediction.

Course Objectives:

- Understand and Implement Basic Neural Network Architectures
- Master Neural Network Training Techniques
- Implement Regularization and Prevent Over fitting
- Utilize Advanced Model Tuning and Transfer Learning
- Explore Model Visualization and Interpretation Techniques

Lab Exercise Programs

1. Write a program to develop a simple perceptron model for binary classification.

2. Write a program to build and train an MLP for a multi-class classification problem using a dataset like MNIST.

3. Write a program to implement gradient descent and back propagation from scratch.

4. Write a program to compare the performance of Batch Gradient Descent, Stochastic Gradient Descent (SGD), and Momentum-based GD on a simple regression problem.

5. Write a program to implement L2 regularization in a neural network and observe its effect on overfitting using a dataset like CIFAR-10.Experiment with early stopping during training.

6. Write a program to build a simple CNN for image classification using a dataset like CIFAR-10. Also Visualize the feature maps and filters learned by the CNN.

7. Write a program to fine-tune a pre-trained model (e.g., VGGNet, ResNet) on a new dataset (e.g., flowers or animals).Compare the performance of different pre-trained models on the same task.

8. Write a program implement techniques to visualize and interpret the activations of different layers in a CNN.Use Grad-CAM or similar methods to visualize class activation maps.

9. Write a program to build and train a simple autoencoder on the MNIST dataset for dimensionality reduction.Experiment with denoising autoencoders and visualize the reconstructed images.

10. Write a program to implement an RNN for sequence prediction (e.g., predicting the next character in a text sequence).Compare the performance of a simple RNN and an LSTM on the same task

Text Books :

1. Ian Good Fellow, Yoshua Benjio, Aaron, Deep Learning Courville, The MIT Press.
2. Christopher Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

Course Outcomes:

After completing the course student will be able to:

- **CO1:** Develop a foundational understanding of neural network architectures, including perceptrons, multi-layer perceptrons (MLPs), and convolutional neural networks (CNNs), to solve binary and multi-class classification problems. *(L2: Understand)*

- **CO2:** Apply optimization techniques like gradient descent, stochastic gradient descent (SGD), and momentum-based gradient descent, and analyze their performance in training neural networks. *(L3: Apply)*

- **CO3:** Investigate techniques for regularization, overfitting prevention (e.g., L2 regularization and early stopping), and pre-trained model fine-tuning to improve neural network generalization on real-world datasets. *(L5: Evaluate)*

- **CO4:** Experiment with advanced deep learning architectures, such as CNNs for image classification and RNNs/LSTMs for sequence prediction, to understand their specific use cases and comparative performance.*(L3:Develop)*

- **CO5:** Implement techniques for visualizing and interpreting neural network behaviors, such as feature map visualization, Grad-CAM, and autoencoders, to gain insights into model decisions and enhance interpretability. .*(L3:Develop)*

CO-PO Mapping table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	3	3	3	2	0	0	3
CO2	3	3	3	3	2	0	0	3
CO3	3	3	3	3	2	0	0	3
CO4	3	3	3	3	2	0	0	3
CO5	3	3	3	3	2	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

APPROVED IN:	
BOS : 8.6.2024	**ACADEMIC COUNCIL:4.7.2024**
SDG No. & Statement:	
4. Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.	
SDG Justification:	
The topics included in this course are designed to get acquainted with one of the skills that handle necessary mathematical orientation, programming techniques and concept based learning.	

Prepared by Dr.M.Srivekatesh

Verified by Dr.M.Srivenkatesh

PROJECT

L	T	P	S	J	C
0	0	3	0	0	8

Hours per week: 4 End Examination: 50 Marks
Credits: 8 Sessionals: 150 Marks

1. Specify the broad topic of the project based on the Machine Learning and Data mining.
2. Study minimum 6 quality research papers based on the selected topic.
3. Do the SWOT analysis of selected research papers/reports.
4. Identify the research problem.
5. Propose your novelty/improvement in terms of algorithm/new feature.
6. Design the architecture for the proposed problem.
7. Design the test bed.
8. Design a set of experiments to be carried out for the proposed problem.
9. Perform the experimental analysis (in Python language only).
10. Prepare your report.

Write a short research paper based on your contribution.

1. Understand the Software Development Automation processes and work to develop a project on software development automation. Understanding Level (Level II)
2. Conduct preliminary literature Review, study different automation tools and find vulnerabilities in the studied literature/tools. Understanding Level (Level II)
3. Analyze and identify the various frameworks, APIs , libraries and tools used for project/ software implementation. Analyzing Level (Level III)
4. Design Software Development Automation software using required frameworks, APIs and libraries. Applying Level (Level IV)
5. Evaluate and validate developed project with respect to various software automation frameworks. Evaluating Level (Level V)

6 .Prepare technical detailed report detailing the problem statement, proposed methodology, software specification, design, test plan, and implementation details. Creating Level (Level VI)

Course Outcomes:

At the end of the course, the student is able to:

- CO1: Understand the principles of Machine Learning, Data Mining, and Software Development Automation. *(L2: Understand)*
- CO2: Critically analyze existing literature, frameworks, and tools to identify research gaps and vulnerabilities. *(L2: Understand, L3: Analyze)*
- CO3: Design and propose novel solutions or enhancements to existing algorithms and frameworks. *(L4: Apply)*
- CO4: Develop a project that incorporates their proposed solution using Python, alongside relevant

software automation frameworks. *(L4: Apply, L5: Evaluate)*

- CO5: Develop a detailed technical report summarizing their research, solution, and experimental findings, contributing to their professional portfolio. *(L3: Develop)*

CO-PO Mapping Table

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	3	2	1	1	1	0	0	2
CO2	3	3	2	2	1	0	0	3
CO3	2	3	3	3	2	0	0	3
CO4	2	3	3	3	2	0	0	3
CO5	2	3	2	2	3	0	0	3

Note: 1 - Low Correlation 2 - Medium Correlation 3 - High Correlation

Prepared by Dr.M.Srivenkatesh

Verified by Dr.M.Srivenkatesh