Rajagiri College of Social Sciences (Autonomous),
Kalamassery

Master of Computer Applications (2 Year)
Syllabus and Scheme
2025 Admission Onwards

BOARD OF STUDIES (COMPUTER SCIENCE)
RAJAGIRI COLLEGE OF SOCIAL SCIENCES (AUTONOMOUS)
KALAMASSERY, KOCHI, 683104
KERALA, INDIA

Table of Contents

Rajagiri College of Social Sciences (Autonomous)

Rajagiri College of Social Sciences (Autonomous) was established through the dedicated efforts and foresight of the Carmelites of Mary Immaculate (CMI) congregation. The institution embodies the spirit of "Relentlessly Towards Excellence," striving for the highest standards in education and holistic development. As the flagship institution of Rajagiri Vidyapeetham, RCSS operates from two scenic campuses in Kochi, Kerala: the Hill Campus at Kalamassery and the Valley Campus at Kakkanad.

Vision

To become a centre par excellence of learning, unique in experience, value-based in its approach, and pioneering in its efforts for enriching and fulfilling LIFE.

Mission

To facilitate comprehensive and integrated development of individuals imbued with righteousness and courage of conviction, to effectively function as social beings.

Core Values

The institution upholds the following core values:

- Service
- Mutual Respect
- Integrity
- Learning
- Excellence

These guiding principles reflect Rajagiri's commitment to fostering an environment of quality education, ethical values, and social responsibility.

Department of Computer Science

The Department of Computer Science at Rajagiri College of Social Sciences (RCSS) has been a pioneer in advanced computer education since its inception in 2001. Affiliated with Mahatma Gandhi University, Kottayam, Kerala, and approved by the All India Council for Technical Education (AICTE), New Delhi, the department has consistently achieved a "Platinum" rating in AICTE-CII surveys, reflecting its outstanding placement records and robust industry engagement.

Programs Offered:

1. Master of Computer Applications (MCA): Originally a three-year program, the MCA was restructured into a two-year course starting from the 2020 admissions, aligning with AICTE guidelines. This program offers various specializations, producing graduates who are highly qualified for the software industry.
2. M.Sc. Computer Science (Data Analytics): Launched in 2020, this two-year master's program addresses the growing demand for skilled data analysts, equipping students with essential competencies in data science.
3. BCA (Hons.): Introduced in the academic year 2024-25, this four-year undergraduate program is meticulously crafted to cater to the evolving demands of the

information technology sector, providing students with a comprehensive understanding of computer science principles and practical skills.

4. B.Sc. Computer Science (Hons.) Data Analytics: Also commencing in 2024-25, this four-year bachelor's program is designed to meet the increasing demand for skilled professionals in data analysis, offering a comprehensive curriculum that covers programming languages, data management, statistical analysis, machine learning, and data visualization.
5. Integrated MCA: Set to begin in the academic year 2024-25, this five-year integrated program aims to provide a seamless education path from undergraduate to master's level in computer applications.
6. Ph.D. in Computer Science

Vision:

To create technically competent individuals who are innovative and uphold human values.

Mission:

To develop globally recognized, competent, and innovative IT professionals committed to lifelong learning and blended with social commitment through comprehensive programs.

Rajagiri Immersive Learning Experience (RILE):

A distinctive pedagogical innovation, RILE aims to develop students into socially responsible professionals through four key dimensions:

- Conceptual Learning: Establishing a strong theoretical foundation in computer science and applications.
- Experiential Engagement: Providing hands-on experience through projects, internships, and workshops.
- Executive Modelling: Cultivating leadership and decision-making skills via case studies and simulations.
- Corporate Competency: Enhancing industry readiness through collaborations and real-world problem-solving.

Faculty Excellence:

Continuous Professional Development: Faculty members stay abreast of the latest technological trends through regular online and onsite training programs. Workshops and lectures conducted by prominent industry professionals and academicians further enrich their expertise. Participation in Faculty Development Programs and other courses is actively encouraged to enhance their knowledge base.

Inclusive Governance:

Rajagiri's governance policy fosters a creative and flexible atmosphere for learning where all individuals are respected and valued, promoting an inclusive community that supports innovation and personal growth.

Through this comprehensive approach, the MCA program at Rajagiri College equips students with the technical proficiency, ethical grounding, and adaptive skills necessary to excel in the ever-evolving field of computer applications.

Comprehensive Academic Offerings

The **Department of Computer Science** offers a diverse range of programs, including **Undergraduate (UG)**, **Postgraduate (PG)**, **Integrated**, and **Ph.D.** programs. Each program is designed to provide students with a balanced mix of theoretical knowledge, practical skills, and research exposure, preparing them to excel in diverse roles across the IT industry, data science, artificial intelligence, cybersecurity, and academic research.

This comprehensive learning approach ensures that graduates are not only technically proficient but also well-rounded professionals equipped with ethical values and social responsibility, ready to lead and innovate in the global technology sector.

Master of Computer Applications (MCA)

The **Master of Computer Applications (MCA)** program at **Rajagiri College of Social Sciences (Autonomous)** is a dynamic two-year postgraduate program designed to develop highly skilled professionals with in-depth knowledge in computer science and cutting-edge technologies. This program is carefully structured to provide a seamless transition from core computing concepts to specialized domains, aligning with industry demands.

In the **first year**, students build a strong foundation in core computer science areas through subjects like **Data Structures Using C**, **Database Management Systems with SQL/PL-SQL**, **Operating Systems with Linux**, **Java Programming**, and **Agile Development with Case Tools**. These foundational courses equip students with essential programming, algorithmic, and problem-solving skills required for advanced learning.

A distinctive feature of this program is the **specialization options offered from the third semester onwards**, allowing students to tailor their learning experience based on their career interests. Students can choose from the following cutting-edge specialization streams:

- **Artificial Intelligence and Machine Learning:** Covering advanced topics like **Deep Learning**, **Natural Language Processing**, **Big Data Analytics**, and **Genetic Algorithms** to develop intelligent systems and data-driven solutions.
- **Advanced Programming:** Focused on modern software development with courses in **Kotlin**, **.NET**, **UX Design**, **Game Development with Unity**, and **Quantum Computing** to master full-stack and enterprise-level applications.
- **Computer Security:** Designed to build expertise in securing digital systems, featuring courses in **Ethical Hacking**, **Database Security**, **Network Security**, **Blockchain Technology**, and **Cyber Security Services**.

Practical learning is integrated throughout the program with hands-on experience in **Full Stack Development**, **Android Development**, **Data Analytics using Python**, and **Cybersecurity Labs**. The curriculum is further strengthened by a mandatory **Internship** and a comprehensive **Capstone Project**, ensuring students gain real-world exposure and industry-ready skills.

By combining core computer science fundamentals with industry-relevant specializations, the MCA program empowers graduates to pursue diverse and high-demand roles in software development, data science, cybersecurity, and emerging technology sectors.

Programme Educational Objectives (PEO) of the MCA Programme

- PEO1 : Graduates of the program will be computer professionals of probity,positive attitude and scientific temper
- PEO2 : Graduates of the program will have sound theoretical knowledge and skillfor software development and implementation
- PEO3 : Graduates of the program will possess good communication, technical and innovative skills
- PEO4 : Graduates of the program will have a sense of social awareness

Programme Outcome (PO)

At the end of the Programme, a student will be able to achieve the following programme outcomes:

1. Computational Knowledge:

Apply knowledge of computing fundamentals, computing specialization, mathematics, and domain knowledge appropriate for the computing specialization to the abstraction and conceptualization of computing models from defined problems and requirements.

2. Problem Analysis:

Identify, formulate, research literature, and solve *complex* computing problems reaching substantiated conclusions using fundamental principles of mathematics, computing sciences, and relevant domain disciplines.

3. Design /Development of Solutions:

Design and evaluate solutions for *complex* computing problems, and design and evaluate systems, components, or processes that meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations.

4. Conduct Investigations of Complex Computing Problems:

Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

5. Modern Tool Usage:

Create, select, adapt and apply appropriate techniques, resources, and modern computing tools to complex computing activities, with an understanding of the limitations.

6. Professional Ethics:

Understand and commit to professional ethics and cyber regulations, responsibilities, and norms of professional computing practice.

7. Life-long Learning:

Recognize the need, and have the ability, to engage in independent learning for continual development as a computing professional.

8. Project management and finance:

Demonstrate knowledge and understanding of the computing and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

9. Communication Efficacy:

Communicate effectively with the computing community, and with society at large, about *complex* computing activities by being able to comprehend and write effective reports, design documentation, make effective presentations, and give and understand clear instructions.

10. Societal and Environmental Concern:

Understand and assess societal, environmental, health, safety, legal, and cultural issues within local and global contexts, and the consequential responsibilities relevant to professional computing practice.

11. Individual and Team Work:

Function effectively as an individual and as a member or leader in diverse teams and in multidisciplinary environments.

12. Innovation and Entrepreneurship

Identify a timely opportunity and using innovation to pursue that opportunity to create value and wealth for the betterment of the individual and society at large.

Program Specific Objectives (PSO)

PSO1: Data Analytics:

Acquire knowledge of Data pre-processing and Data quality, Modelling and design of data warehouses, Algorithms for data mining, skills to design, analyse and develop algorithms and implement using high-level programming languages and to define and critically analyse mining approaches for various domains.

PSO2: High-Level Programming:

Acquire skills to design, analyse and develop algorithms and implement those using high-level programming languages, to maintain web server services required to host a website, Install, configure, design and develop mobile application development tools.

PSO3: Practices and tools in Information Security:

Acquire a practical overview of theissues involved in the field of information security and assurance; acknowledge the ethical considerations in all dimensions of information security, and utilize the software tools to explore, rectify or prevent the unauthenticated actions in the domain.

Mapping of PO to PEO

Program Educational Objectives	PEO1	PEO2	PEO3	PEO4
Program Outcomes				
PO1: Computational Knowledge		√		
PO2: Problem Analysis			√	
PO3: Design /Development of Solutions	√			

PO4: Conduct Investigations of Complex Computing Problems			√	
PO5: Modern Tool Usage			√	
PO6: Professional Ethics	√			√
PO7: Life-long Learning	√			
PO8: Project management and finance				√
PO9: Communication Efficacy			√	
PO10: Societal and Environmental Concern	√			√
PO11: Individual and Team Work	√			√
PO12: Innovation and Entrepreneurship			√	
PSO1: Data Analytics		√	√	
PSO2: High-Level Programming		√	√	
PSO3: Practices and tools in InformationSecurity		√	√	

Eligibility Criteria

A candidate seeking admission to MCA course must have

- A pass with not less than 50% marks in any recognized Regular Bachelors Degree of minimum three years duration in any discipline with Mathematics atplus two level OR at Graduation level with Mathematical Science (Mathematics /Statistics /Computer Science / Operation Research /Quantitative Techniques)as one of the Subjects.
  OR
- A pass with not less than 50% marks in BCA/BSc. Computer Science/ BSc.Information Technology/ B.Tech. from a recognized University.
  OR
- Candidates with such degrees awarded by the parent University or any otherdegree recognized as equivalent to the above listed courses by the Mahatma Gandhi University also are eligible to apply.
- Relaxation in eligibility and minimum marks of the qualifying examination are subject to the respective regulation and existing rules of parent University.
- Candidates who have passed the qualifying examination with more than one chance will have their percentage marks de-rated at the rate of 5% for every additional appearance for the purpose of ranking

Programme Structure and Duration

The duration of the programme shall be 4 semesters. The duration of each semester shall be 90 working days. Odd semesters from June to October and even semesters from

November to march.

A student may be permitted to complete the programme, on valid reasons, within a periodof 8 continuous semesters from the date of commencement of the first semester of the programme.

The medium of instruction shall be English.

Attendance

The minimum requirement of attendance for each course during a semester for appearing at the end-semester examination shall be 75%. Condonation of shortage of attendance to a maximum of 15 days in a semester subject to a maximum of two times during the whole period of the programme may be granted by the Principal, Rajagiri College of Social Sciences (Autonomous), Kalamassery.

Those who could not register for the examination of a particular semester due to shortage of attendance may repeat the semester along with junior batches, without considering sanctioned strength, subject to the existing Rules of the institution.

A Regular student who has undergone a programme of study under earlier regulation/scheme and could not complete the Programme due to shortage of attendance may repeat the semester along with the regular batch subject to the condition that he has to undergo all the examinations of the previous semesters as per the 2021 Regulations

A student who had sufficient attendance and could not register for fourth semester examination can appear for the end semester examination in the subsequent years with the attendance and progress report from the Principal.

Promotion

A student who registers for a particular semester examination shall be promoted to the next semester.

A student having 75% attendance for each course and who fails to register for examination of a particular semester will be allowed to register notionally and is promoted to the next semester, provided application for notional registration shall be submitted with 15 days from the commencement of the next semester.

Evaluation

The evaluation scheme for each credit course shall contain two parts; (a) Semester Evaluation (ESE) [External Evaluation] and (b) Continuous Evaluation (CE) [Internal Evaluation]. 25 marks shall be given to internal evaluation and the remaining 75 marks to external evaluation

Continuous Assessment Criteria

Theory Course

Sl No	Component		Marks
1	Continuous Assessment Examination (CAE)	CAE 1	7.5 Marks
2		CAE 2	7.5 Marks
3	Assignment/Project/Term paper (Individual)/Class Participation/Presentation/Quizzes/Seminars/Case Studies/ Group Project work/VIVA voce/ MOOC Course etc... **(Any two** is compulsory)		(5+5) Marks
Total			25 Marks

Practical Course

Lab Performance	5 marks
Continuous Assessment Examination	10 marks (Lab Exercise + viva)
Lab Record	5 marks
Assignment/Project/ Test/Quiz	5 marks
Total	25 marks

Internship

Interim Presentation	70 marks
Guide – Prompt Communication, Clarity on the Tasks	50 marks
Interim Report	30 marks
Internal Marks	150 marks

External evaluation

Pattern of ESE

The Question paper must cover question sets from all modules of the subject carrying equal weightage. The question set from each module can be given as a choice of either-or Questions. The sub questions in the set can be a combination of short answers, descriptive questions, problem solving questions as well as practical questions.

Internship

Viva 1. Attainment of Internship objectives 2. Skills gained during the internship 3. Commitment towards the tasks allocated 4. The challenges faced and the strategy to overcome 5. Key contributions of the intern to the organization	100 marks
Final Report	50 marks
Internal Marks	150 marks

Grading

The performance of a student in the programme is evaluated using indirect grading system.

Course Grade

The performance of a student in each course is evaluated in terms of percentage of marks with a provision for conversion to grade points. The grading system followed is that of relative grading on a ten-point scale. Letter grade corresponding to total marks in percentage, M (Internal + External) and the corresponding grade point in a ten-point scale is described in the table:

Range of Marks in %(M)	Grade Letter	Performance	Grade Point
90 – <= 100	S	Outstanding	10
80 – <90	A	Excellent	9
70 – <80	B	Very Good	8
60 – <70	C	Good	7
55 – <60	D	Average	6
50 – <55	E	Pass	5
0 – <50	F	Fail	0

Semester Grade

The overall grade point of a student in a semester is measured as Semester Grade Point Average (SGPA). It is the ratio of sum of the product of the number of credits with the grade points scored by a student in all the courses taken by a student and the sum of thenumber of credits of all the courses undergone by a student.

$$SGPA\ (Si) = \Sigma(Ci \times Gi) / \Sigma Ci$$

where Ci is the number of credits of the i^{th} course and Gi, is the grade pointscored by the student in the i^{th} course.

Semester Grade = Letter Grade corresponding to SGPA from the performance grade scale.

Overall grade of the programme

The overall grade point of a student in the entire MCA Programme is measured as Cumulative Grade Point Average (CGPA). The CGPA is calculated by taking into account all the courses undergone by a student for all the semesters of a programme,

$$CGPA = \Sigma(C \times Si) / \Sigma Ci$$

where Si is the SGPA of the i^{th} semester and Ci is the total number of credits inthat semester.

Overall Grade awarded in the Programme is the letter grade corresponding to CGPA from

the performance grade scale.

Note : All grade point averages should be rounded off to 2 decimal points

Conversion of CGPA to Percentage

Percentage marks equivalent to CGPA = (**CGPA** Obtained x 100) **/** Maximum

PASS REQUIREMENTS

Course

- A student who obtained Grades S to E (grade point not less than 5) shall be considered as passed. If a student secured "F" grade, he /she has to reappear for the examination. It is mandatory for a student to earn the required credits as mentioned in each semester.
- For a pass in a course with internal and external evaluation, a student shall secure minimum of 40% of the maximum marks prescribed in the end semester Examination and 50% of marks in the aggregate marks in the courseincluding sessional marks.
- No separate minimum is required for internal evaluation /sessional marks.
- The students who do not satisfy the above condition or the student who remains absent shall be deemed to have failed in that course and may reappearfor the external examination in the subsequent examinations. However, the Sessional marks awarded to the student/s at previous attempt in the concerned subject will be carried forward.
- If a student fails in a subject having only internal assessment/ evaluation, he/she has to redo the work for that subject along with the subsequent batch.

Semester

- For Successful completion of a semester, a student should pass all courses in the semester and score minimum E grade for the semester (SGPA should not be less than 5). However, a student is permitted to move to the next semester irrespective of his/ her SGPA provided they have completed all the requirements of attendance, payment of all fees due to the University and Institution and registration for the examinations in the earlier semesters.

Programme

- For successful completion of the programme, a student should successfully complete all four semesters with a minimum SGPA score of 5 and with a minimum CGPA score of 5.

Award Of Degree

The successful completion of all courses with minimum E grade within the stipulated period is the minimum requirement for the award of degree.

- A candidate who qualifies for the award of the degree securing E or above grade in all courses pertaining to all semesters in his/her first attempt within four consecutive semesters, and in addition secures SGPA of 7.5 or above for semestersI to IV shall be declared as passed the examination in FIRST CLASS WITH

DISTINCTION.

- A candidate who qualifies for the award of the degree by securing E or above grade in all subjects of all the semesters within a maximum period of eight semesters, after the commencement of his/her study in the 1st semester and secures CGPA not less than 6.00 shall be declared as passed the examination in FIRST CLASS.
- All other candidates who qualify for the award of the degree by securing E or above grade in all subjects of all semesters within a maximum period of eight semesters, after the commencement of his/her study in the 1st semester and secures CGPA not less than 5.0 shall be declared as passed the examination in SECOND CLASS.

Bridge Courses

#	Course Name	Hours
1	Programming concepts using C language	14
2	Basic Mathematics and Statistics	6
3	Digital Logic and Computer Organization	4
4	Basic Principles of Management	4
5	Skill and Personality Development Workshop	4 Full Working Days
	Evaluation after Bridge Course	3 Hour Lab Exam

Semester Courses

Semester I

Code	Course Name	Type	Hours			Exam	CAE	ESE	Credit
			Lecture	Tutorial	Practical	(hours)	Marks	Marks	
MCA101	Probability, Statistics and Computational Mathematics	Core	3	1	0	3	25	75	4
MCA102	Data Structures using C	Core	2		4	3	25	75	4
MCA103	Database Management System with SQL/PL-SQL	Core	2		4	3	25	75	4
MCA104	Computer Networks and Internet Architecture	Core	3	1	0	3	25	75	4
MCA105	Operating Systems with Linux as Case study	Core	3		2	3	25	75	4
MCA106	Hardware and Networking Lab	Lab			2	2	50		1
Total marks		550				Total Credits			21

Semester II

Code	Course Name	Type	Hours			Exam	CAE	ESE	Credit
			Lecture	Tutorial	Practical	(hours)	Marks	Marks	
MCA201	Operations Research	Core	3	1	0	3	25	75	4
MCA202	Java Programming	Core	2	0	4	3	25	75	4
MCA203	Agile Development and Case tools	Core	3	0	2	3	25	75	4
MCA204	Design and Analysis of Algorithms	Core	3	1	0	3	25	75	4
MCA205	Artificial Intelligence with Gen AI	Core	3	0	2	3	25	75	4
MCA206	Full stack development	Lab	2	0	4	3	25	75	3
Total marks		600				Total Credits			23

Semester III

Code	Course Name	Type	Hours			Exam	CAE	ESE	Credit
			Lecture	Tutorial	Practical	(hours)	Marks	Marks	
MCA301	Internet of things	Core	3	1	0	3	25	75	4
MCA302	Theory of Computation and Compilers	Core	3	1	0	3	25	75	4
MCA303	Machine learning	Core	3		2	3	25	75	4
MCA304	Cyber Security Essentials	Core	3	1		3	25	75	4
MCA3XX	Elective-I	Core	3	1		3	25	75	4
MCA306	Data Analytics using Python	Lab			6	3	25	75	3
MCA307	Android	Lab			4	3	25	75	2
Total marks		700				Total Credits			25

Semester IV

Code	Course Name	Type	Hours			Exam	CAE	ESE	Credit
			Lecture	Tutorial	Practical	(hours)	Marks	Marks	
MCA401	Parallel Computing	Core	3	1	0	3	25	75	4
MCA4XX	Elective-II	Core	3	1	0	3	25	75	4
MCA403	Comprehensive Viva-Voce							100	3
MCA404	Internship						100	100	10
Total marks		500				Total Credits			21
Total Credits for MCA	90	Total Marks For MCA	2350						

Elective Courses

Specialization Stream1		Specialization Stream2		Specialization Stream3	
AI and Machine Learning		Advanced Programming		Computer Security	
Code	Course Name	Code	Course Name	Code	Course Name
Pool1(MCA305)					
MCA311	Deep Learning	MCA321	Kotlin	MCA331	Ethical Hacking
MCA312	Inferential Statistics	MCA322	.NET	MCA332	Database Security
MCA313	Natural language Processing	MCA323	UX design	MCA333	Network Security
MCA314	Image Mining	MCA324	Game Development with Unity	MCA334	Digital Identity management
Pool2(MCA402)					
MCA413	Tableau	MCA423	Software testing with Selenium	MCA433	Cyber Security Services
MCA414	Bigdata Analytics	MCA424	Struts, Hibernate and Spring	MCA434	Block-Chain
MCA415	Genetic Algorithms	MCA425	Project Management Tools	MCA435	Firewall Management

Bridge courses

Programming concepts using C language

- Work with primitive types and expressions
- Understanding the basic structure of a C Program, the main function and using standard I/O
- Understand C Character Set and Tokens, Data Types, Variables and Constants
- Work with Operators and Expressions
- Control the flow of Program using Conditional statements and Loops
- Work with Built-in Functions and User Defined Functions with arguments, Passing arguments By Value and By Reference
- Work with Arrays and Strings
- Understand and Implement Pointers, Pointer to an array, Array of pointers, Pointers and functions
- Defining Structures and Union
- Data File Handling
- Debug C programs effectively

Basic Mathematics and Statistics

- Set theory

  Sets and their representations; The empty set; finite and infinite sets; equal and equivalent sets; subsets; power set; universal set; Venn diagrams; complement of a set operation on sets; applications of sets.
- Mathematical Logic

  Basic Logical connections; Conjunction; Disjunction; Negation; Negation of Compound Statements; Truth tables. Tautologies; Logical Equivalence; Applications.
- Modern algebra

  Binary Operation; Addition Modulo n; Multiplication modulo n
- Matrices and Determinants

  Definition of a matrix; Operations on matrices; Square Matrix and its inverse; determinants; the inverse of a matrix
- Basics Statistics

  Measures of central Tendency; Standard Deviation; Discrete series. variance.

Digital Logic and Computer Organization

- **Computer Evolution**: Brief history of Computer, Classification of Computer, Structure of a Computer System, Arithmetic Logic Unit, Control Unit, Bus Structure, Von Neumann Architecture. Bootstrapping.
- **Number systems** - Decimal, Binary, Octal, Hexadecimal conversion from one to another, Basic Arithmetic Operations: Integer Addition and Subtraction, Signed

numbers, Binary Arithmetic, 1's and 2's Complement Arithmetic, Fixed and Floating point numbers, Floating point representation.,

- **Digital Logic**: Logic gates, Boolean Algebra, Basic theorem and Properties of Boolean algebra. Basic concepts on Combinational Circuits and Sequential circuits
- **Control Unit Design**: Basic Concepts – Instruction execution cycle – sequencing of control signals
- **Memory Organization**: Characteristics of Memory Systems, Main Memory, Types of Random-Access Memory and ROM, Organization, Static and dynamic memories. Understanding Cache Memory and Virtual Memory
- **Input / Output Organization**: Accessing I/O devices – Understanding Programmed I/O, Interrupt I/O and Direct memory access (DMA)

Basic Principles of Management

- Introduction to principles of management: Planning, organizing, staffing, Budgeting, Controlling.
- Scope of IT applications in management, and its benefits
- Socioeconomic environment and information systems, and its impact
- Strategic role of IT in organizations
- Critical success factors as its role in implementing IT applications
- Case studies of successful / failed IT applications

Mandatory Tools

- GitHub
- LinkedIn
- Hacker Rank

SEMESTER I

MCA101 Probability, Statistics and Computational Mathematics

GENERAL DETAILS OF THE COURSE

Course Code	MCA101		
Course Name	Probability, Statistics and Computational Mathematics		
Course Type	Core		
Credit	4		
Semester	I		
Course Objectives	To understand the concept of probability, statistics and computational mathematics and it uses in computer science problems.		
Pre-requisites	Bridge Course in Mathematics.		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	3	1	0

COURSE OUTCOMES (CO)

CO#	CO Description
CO1	To gain fundamental understanding of Probability, conditional probability and Bayes theorem.
CO2	Understand and describe various probability distributions
CO3	To apply the concept of statistics in real life problems.
CO4	To gain fundamental understanding of mathematical logic.
CO5	To have the concept of counting and mathematical induction.

CO-PSO-PO MAPPING MATRIX

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO#1	2	2	1	2									2		
CO#2	2	2	1										2		
CO#3	2	3		3									2		
CO#4	2	3		3											
CO#5	2	2	1	2											

COURSE CONTENT

Module No		Module Content	Hours Required
1	1.1	Probability Theory: Sample space, Events, Different approaches to probability	12
	1.2	Addition and multiplication theorems on probability	

	1.3	Independent events, Conditional probability, Bayes Theorem	
2	2.1	Random variables and Distribution: Random variables, Probability density functions and distribution functions	12
	2.2	Marginal density functions, Joint density functions, mathematical expectations, moments and moment generating functions	
	2.3	Discrete probability distributions - Binomial, Poisson distribution	
	2.4	Continuous probability distributions- uniform distribution and normal distribution	
3	3.1	Basic Statistics: Measures of central tendency: - mean, median, mode;.	12
	3.2	Measures of dispersion: Range, Mean deviation, Quartile deviation and Standard deviation	
	3.3	Moments, Skewness and Kurtosis, Linear correlation	
	3.4	Karl Pearson's coefficient of Correlation, Rank correlation and linear regression	
4	4.1	Mathematical Logic: Propositional and Predicate Logic, Propositional Equivalences	12
	4.2	Normal Forms, Predicates and Quantifiers	
	4.3	Nested Quantifiers, Rules of Inference.	
5	5.1	Counting, Mathematical Induction: Basics of Counting, Pigeonhole Principle	12
	5.2	Permutations and Combinations	
	5.3	Inclusion- Exclusion Principle, Mathematical Induction.	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

Fundamentals of statistics: S. C. Gupta, 6thRevised and enlarged edition April 2004, Himalaya Publications
Fundamentals of Mathematical Statistics- S.C.Gupta ,V.K.Kapoor. Sultan Chand Publications.
Introduction to Mathematical Statistics -Robert V. Hogg &Allen T. Craig. Pearson education
Discrete Mathematical Structures with Applications to Computer Science by J. P. Tremblay and R. P. Manohar, Tata McGraw-Hill, 2001.
C. L. Liu, Elements of Discrete Mathematics, 2nd Edition, Tata McGraw-Hill, 2000.

MCA102 Data Structures using C

GENERAL DETAILS OF THE COURSE

Course Code	MCA102		
Course Name	Data Structures using C		
Credit	4		
Semester	I		
Course Objectives	To introduce the concept of linear and nonlinear data structures. To implement the concepts using arrays and linked list To apply it to advanced data structures.		
Pre-requisites	Bridge Course		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	2	0	4

COURSE OUTCOMES (CO)

CO#	CO Description
CO#1	To differentiate the linear and nonlinear data structures
CO#2	Implement the various kinds of sorting and searching techniques.
CO#3	To implement the concept of nonlinear data structures using arrays and linked list.
CO#4	Familiarize the concept of advanced data structures like red black trees, AVL trees etc.
CO#5	Implement the concept of balancing a tree and the rotations to do it.

CO-PSO-PO MAPPING MATRIX

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO#1	3						1				2	2		2	
CO#2			3				1				2	2		2	
CO#3	3		2				1				2	2		2	
CO#4			3				1				2	2		2	
CO#5			2				1				2	2		2	

COURSE CONTENT

Module No		Module Content	Hours Required
1	1.1	Introduction: Data Structures, Data Types, Structure.	12
	1.2	Arrays: Polynomial Representations, Polynomial addition, Polynomial Multiplication and sparse matrices	

	1.3	Stack: Definition and concepts, Operations on stacks.	
	1.4	Application of stacks- Infix to postfix conversion, Evaluation of Arithmetic Expression.	
2	2.1	Queue: Representation of queue, circular queue and double-ended queue. Priority queue: implementation by array using Heap Sort	12
	2.2	Dynamic Memory Allocation Functions: malloc, calloc, realloc and free	
	2.3	Linked List: Operations – insertion, searching, removing, updating, sorting and reversing.	
	2.4	Polynomial: Representations, Addition, Multiplication using Linked List	
3	3.1	Linear Data Structures: Linked stacks, Linked queues, Circular Linked List and Double Ended Queue, Doubly Linked List and Circular doubly linked list.	12
	3.2	Non-Linear Data Structures: Trees, Graphs.	
	3.3	Graph: Representation of Graph on Computer: Adjacency matrix and adjacency list, merits and demerits of graph representation	
	3.4	Searching: Linear Search, Binary Search	
4	4.1	Trees: Basic terminology, binary trees, binary search tree	12
	4.2	Binary search tree: Insertion, Deletion, searching and Traversal - inorder, pre-order and post-order.	
	4.3	Threaded Binary Tree: Operations	
	4.4	Balanced Trees: AVL Tree: properties, insertion, deletion and rotations	
5	5.1	Advanced Data Structures: Red black tree: properties.	12
	5.2	B-Trees: Data Structure on secondary storage, Definition of B trees, Basic operations on B Trees – searching, creating an empty node, splitting a node in B Tree, Inserting a key in to B Tree and Deleting a Key from a B Tree	
	5.3	Definition and Structure: B+ Trees	
	5.3	Data Structure for Disjoint Sets: Disjoint set operation, linked list representation of disjoint sets, Disjoint-set forests	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

Fundamentals of Data Structures in C by Horowitz, Sahni and Anderson-Freed.
Data Structures Through C in Depth by S.K Srivastava, Deepali Srivastava
Data Structures Using C Aaron M. Tenenbaum
Data Structures Using C, Reema Thareja

MCA103 Data Base Management Systems with SQL & PL-SQL

GENERAL DETAILS OF THE COURSE

Course Code	MCA103		
Course Name	Data Base Management Systems with SQL & PL-SQL		
Course Type	Core		
Credit	4		
Semester	I		
Course Objectives	The course aims to provide students with a comprehensive understanding of Database Management Systems (DBMS), and make students proficient in designing and managing relational databases, crafting simple and complex SQL queries, and utilizing PL/SQL to develop database-driven applications.		
Pre-requisites	Nil		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	2	0	4

COURSE OUTCOMES (CO)

CO#	CO Description
CO1	To understand the concepts of relational Relational model and to analyse a scenario and design conceptual models of a database using ER modeling for real life applications.
CO2	To analyze the existing design of a database schema and apply concepts of normalization to design an optimal database.
CO3	To construct simple database queries using Structured Query Language.
CO4	To construct moderately advanced database queries using Structured Query Language.
CO5	To construct simple and moderately advanced PL SQL codes using procedure, functions, cursors and triggers.

CO-PSO-PO MAPPING MATRIX

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO#1		3	2												
CO#2		3	2												
CO#3		3	2		2										
CO#4		3	2		2										
CO#5						3				3					

COURSE CONTENT

Module No		Module Content	Hours Required
1	1.1	Introductory concepts of DBMS- Introduction and applications of DBMS, Purpose of data base, Database users & DBA, View of Data.	12
	1.2	Relational Model: Structure of relational databases, Entity Relationship model: Basic concepts-ER diagrams, Mapping cardinalities, Strong & Weak entity sets, extended E-R features – generalization, specialization, aggregation	
	1.3	Reduction to E-R database schema.	
	1.4	Introduction to Database Transactions-ACID Properties	
2	2.1	Relational Database design -Functional Dependency – definition, trivial and non-trivial FD, closure of FD set, closure of attributes, irreducible set of FD	12
	2.2	Normalization – 1Nf, 2NF, 3NF, BCNF, Multivalued dependency, 4NF, Join dependency and 5NF	
	2.3	Decomposition using FD dependency preservation	
3	3.1	SQL- DDL structure – creation, alteration, dropping of database schemas, defining constraints – Primary key, foreign key, unique, not null, check, IN operator,	12
	3.2	DML Structure-Select, from, where clauses, Functions - Built-in functions – numeric, date, string functions, aggregate functions, set operations	
	3.3	Use of group by, having, order by	
4	4.1	Querying multiple tables, SQL Joins- types	12
	4.2	Subqueries, Correlated sub- queries, Exist, Any, All clauses in subqueries, View and its types	
	4.3	Transaction control commands – Commit, Rollback, Save point, DCL –Grant and Revoke	
	4.4	Implicit & Explicit Cursors, Stored Procedures and Functions, Triggers, DDL, and Event Database Triggers	
5	5.1	Basics of MS Access forms, designing forms for data entry and retrieval, validation rules, input masks,	12
	5.2	Develop socially relevant, small-scale applications aligned with UN Sustainable Development Goals (SDGs). The focus should be on usability and effective database integration. These projects should address critical societal challenges and promote community well-being.	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

Database System Concepts – Silberchatz, Korth and Sudarsan, Seventh Edition, McGraw Hill, 2019
Database Management Systems – Raghu Ramakrishnan and Johannes Gehrke, Third Edition, McGraw Hill, 2003
Database Systems: Design ,Implementaion and Management, Peter Rob, Thomson Learning, 7Edn.
Concept of Database Management, Pratt, Thomson Learning, 5Edn.
https://www.oracletutorial.com/
https://www.javatpoint.com/oracle-tutorial
https://livesql.oracle.com/

MCA104 Computer Networks and Internet Architecture

GENERAL DETAILS OF THE COURSE

Course Code	MCA104		
Course Name	Computer Networks and Internet Architecture		
Course Type	Core		
Credit	4		
Semester	I		
Course Objectives	To provide an in-depth understanding of computer networks, including their design, architecture, protocols, and management. To explore the structure and functioning of the Internet and its key components. To analyze and compare various network protocols and architectures. To prepare students for advanced topics in network research and development.		
Pre-requisites	Nil		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	3	1	0

COURSE OUTCOMES (CO)

CO	CO Description
CO1	Analyze network protocols and their functions across layers.
CO2	Evaluate different network architectures and their performance.
CO3	Implement routing algorithms in simulated networks.
CO4	Apply security measures to protect network communications.
CO5	Design network topologies using Cisco Packet Tracer.

CO-PSO-PO MAPPING MATRIX

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO#1	3	1			3										1
CO#2	3	3			2										1
CO#3	3	3			2										1
CO#4	3	3			2										1
CO#5	3	2			2										1

COURSE CONTENT

Module No	Module Content	Hours Required

1	1.1	Overview of Data Communication and Networking.	12
	1.2	Network Models: OSI Model, TCP/IP Model.	
	1.3	Physical Layer: Transmission Media, Switching, and Multiplexing.	
	1.4	Data Link Layer: Error Detection and Correction, Data Link Control.	
2	2.1	Introduction to the Network Layer.	12
	2.2	IP Addressing, Subnetting, and CIDR, Implementation using simulation- CISCO packet tracer.	
	2.3	Routing Algorithms: Link State, Distance Vector, and Path Vector.	
	2.4	Network Layer Protocols: IPv4, IPv6, ICMP.	
3	3.1	Introduction to the Transport Layer.	12
	3.2	TCP and UDP: Features, Differences, and Use Cases.	
	3.3	Flow Control and Congestion Control Mechanisms.	
	3.4	Quality of Service (QoS)	
4	4.1	Overview of the Application Layer.	12
	4.2	Protocols: HTTP, FTP, SMTP, DNS, DHCP.	
	4.3	Introduction to Network Security: Cryptography, Firewalls, and VPNs.	
	4.4	Security Protocols: SSL/TLS, IPSec.	
5	5.1	The Structure and Functioning of the Internet.	12
	5.2	Internet Infrastructure: ISPs, IXPs, and Autonomous Systems.	
	5.3	Cloud Computing and Content Delivery Networks (CDNs).	

REFERENCES (TEXTBOOKS/ RESEARCHES/ ONLINE URLS)

Data Communications and Computer Networks by Behrouz A. Forouzan
Computer Networks by Andrew S. Tanenbaum
Computer Networking: A Top-Down Approach by James F. Kurose and Keith W. Ross
High-Performance Communication Networks by Jean Walrand and Pravin Varaiya
Web Link: Cisco Networking Academy
Web Link:IETF

MCA105 Operating Systems with Linux as Case study

GENERAL DETAILS OF THE COURSE

Course Code	MCA105		
Course Name	Operating Systems with Linux as Case study		
Course Type	Core		
Credit	4		
Semester	I		
Course Objectives	To furnish students with a thorough and in-depth knowledge of the operating system principles, techniques and approaches. To comprehend how operating systems handle resources like memory and peripherals, coordinate CPU usage, and facilitate communication between applications, users, and the hardware. To acquire knowledge and competence in using the Linux operating system.		
Pre-requisites	Basic Computer Knowledge		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	3	0	2

COURSE OUTCOMES (CO)

CO#	CO Description
CO1	Elaborate the understanding of an operating system by giving emphasis on the characteristics of different types of operating systems and functionalities.
CO2	Understand the concept of Process Management and Inter Process communication Component of an Operating System.
CO3	Comprehend the primary memory control and interaction of an operating system.
CO4	Realize the importance and the implementation of protection mechanism used by an operating system
CO5	Learn the concepts of operating system through experimental practice using Linux operating system

CO-PSO-PO MAPPING MATRIX

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO#1			3												
CO#2	1		3												
CO#3	2		3												
CO#4			3												
CO#5			1		2										

COURSE CONTENT

Module No		Module Content	Hours Required
1	1.1	Introduction to OS: Evolution of operating systems:-Serial processing, Batch Processing, multiprogramming. Types of operating systems-Batch Operating System, Multi-programming-Time sharing, Real time, distributed operating systems. Operating Systems Structures :- Systems Components, Operating System Services, System Calls, System Programs, System Structures	12
	1.2	Linux: History of Linux: Linux Operating System Layers, The Linux Shell Process: (parent and child processes), Files and Directories (File Structure and directory structure)	
	1.3	Linux Basic commands: pwd, cd, mkdir, rm, mv, touch, man, cp, locate, echo, cat, touch, ls, cut, paste and other basic shell management commands	
2	2.1	Process Management and Concurrency management: Process Attributes, PCB, Process States, Context Switching, CPU and I/O bound process, Process Scheduling - Long term Short term, Medium term schedulers	12
	2.2	CPU Scheduling algorithms - FCFS, SJF, SRT, RR, Priority scheduling, MLQ, MLFQ, Inter- process Communication (IPC) - Message Passing, Shared Memory, Pipe, Message Queue,	
	2.3	Process related commands: fork, exec, ps, kill, nice, foreground process, background process	
3	3.1	Concurrency Management Process Synchronization- Semaphore, Peterson's solution, Mutex, Mutual Exclusion,	12
	3.2	Classical IPC Problems- Producer-Consumer Problem, Readers Writers problem, Dining Philosophers Problem. Deadlocks: Detection-Resource-Allocation Graph Algorithm, Prevention-Banker's Algorithm, Deadlock Avoidance and Recovery.	
	3.3	Shell variables, redirection, filters Shell Scripting	
4	4.1	Memory Management Logical Address Space and Physical Address Space, Conversion of Logical address to Physical address, Memory Partitioning, Contiguous Partitioning - Fixed and Variable sized partitioning, Best Fit, Worst Fit, First Fit, Next Fit methods, External and Internal fragmentation, Non-Contiguous Partitioning-Paging, Page table, Page Fault, TLB, Virtual Memory, Demand Paging,	12

		Thrashing	
	4.2	Page Replacement Algorithms- FIFO, OPR, LRU, MRU, Segmentation, Segmentation Table, Compaction.	
	4.3	Linux Commands: top, free, vmstat, and other memory related commands. Installation of Linux OS	
5	5.1	File Systems and Storage Management File System Basics and Structure, File Operations, File allocation methods Contiguous, Linked, Indexed, FAT, File Access methods- Sequential, Direct, Indexed sequential,	12
	5.2	Directory structure- Single level, Two level, hierarchical, acyclic, File Protection- File permission, encryption, ACL , Disk management- Disk structure - MBR, bad block, Disk scheduling algorithms- FCFS, SSTF, SCAN, C-SCAN, LOOK, C-LOOK, RAID	
	5.3	Protection and case STUDY: LINUX Access matrix, Implementation of access matrix, Revocation of access rights.	
	5.4	Linux OS – Administering Users and Groups: Administering User Accounts, Working with Group Accounts, Understanding the Root Account, installing packages	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

References
Silberschatz, Galvin, and Gagne, "Operating System Concepts", Eighth Edition, Wiley Publication, 2011.
Andrew S. Tanenbaum, "Modern Operating Systems", Second Edition, Pearson Education, 2004.
Gary Nutt, "Operating Systems", Third Edition, Pearson Education, 2004
Harvey M. Deital, "Operating Systems", Third Edition, Pearson Education, 2004.
Milan Milenkovic, "Operating Systems: Concept and Design", 2nd Edition, 2001.
Richard Petersen, "Linux: The Complete Reference", Sixth Edition, 2007

MCA106 Hardware and Networking Lab

GENERAL DETAILS OF THE COURSE

Course Code	MCA106		
Course Name	Hardware and Networking Lab		
Credit	1		
Semester	I		
Course Objectives	This course aims to provide students with hands-on experience in computer hardware assembly, troubleshooting, and maintenance, along with foundational networking skills. Students will learn to set up and manage local networks, configure network devices, and implement basic security measures. This course aims to build practical skills essential for hardware and network management in real-world scenarios.		
Pre-requisites	Basics of Programming Using C		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	0	0	2

COURSE OUTCOMES (CO)

CO#	CO Description
CO1	Students will be able to identify, assemble, and troubleshoot computer hardware components, ensuring optimal system functionality.
CO2	Students will gain practical skills in setting up local area networks (LANs), configuring network devices, and managing IP addressing and file sharing.
CO3	Students will utilize diagnostic tools like Wireshark to monitor and analyze network traffic and implement basic security measures such as firewalls and VPNs to safeguard network infrastructure.

CO-PSO-PO MAPPING MATRIX

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	P 0 10	PO 11	PO 12	PS O 1	PS O 2	PSO 3
CO1		1			2										
CO2		3													1
CO3					3										1

COURSE CONTENT

Module No		Module Content	Hours Required
1	1.1	Identification of motherboard, CPU, RAM, power supply, storage devices, and peripherals.	4
	1.2	Functions and compatibility of each component.	
	1.3	Steps to assemble and disassemble a PC.	
	1.4	Cable management and safety precautions. Diagnosing common hardware issues and replacing faulty components.	
2	2.1	Overview of LAN, WAN, MAN, and PAN. Understanding IP addressing and subnetting.	4
	2.2	Network Devices and Setup- Configuration of routers, switches, hubs, and access points. Connecting devices in a LAN environment.	
	2.3	Creating straight-through and crossover cables using RJ45 connectors. Testing cables with a network cable tester.	
3	3.1	Assigning IP addresses manually and via DHCP. Configuring file sharing and printer sharing.	4
	3.2	Using tools like Wireshark or Netstat for network diagnostics. Analysing network traffic and identifying issues.	
	3.3	Configuring firewalls and basic network security settings. Introduction to VPNs.	
4	4.1	Installing and updating drivers. Installing operating systems and configuring BIOS/UEFI settings.	3
	4.2	Creating system backups and restoring from recovery media.	
	4.3	Cleaning hardware, checking performance metrics, and optimizing system performance.	

SEMESTER II

MCA201 Operations Research

GENERAL DETAILS OF THE COURSE

Course Code	MCA201		
Course Name	Operations Research		
Course Type	Core		
Credit	4		
Semester	II		
Course Objectives	To introduce the students how to use variables for formulating complex mathematical models in management science, linear programming, game theory, queuing theory and simulation.		
Pre-requisites	Familiarity with Linear Algebra, MCA101		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	3	1	0

COURSE OUTCOMES (CO)

CO#	CO Description
CO1	Formulate a real-world problem as a mathematical programming model.
CO2	Understand the theoretical workings of the simplex method for linear programming and perform iterations of it by hand.
CO3	Solve specialized linear programming problems like the transportation and assignment problems
CO4	Understand the basic concept of game theory and queuing theory.
CO5	Understand the network analysis techniques and Simulation.

CO-PSO-PO MAPPING MATRIX

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3
CO1	2	1											2	1	
CO2			2												
CO3	2		1										2		
CO4		2	1											2	
CO5		2		2										2	

COURSE CONTENT

Module No		Module Content	Hours Required
1	1.1	Linear programming problems - Mathematical formulation,	12
	1.2	Graphical method of solution, **simplex method**	
2	2.1	Duality in linear programming problems	12
	2.2	dual simplex method, sensitivity analysis, transportation and assignment problems	
	2.3	Traveling salesman Problem	
3	3.1	Game theory Introduction, two-person zero-sum games, some basic terms	12
	3.2	The maxmini-minimax principle, games without saddle points- Mixed Strategies, graphic solution of 2 * n and m*2 games, dominance property.	
	3.3	CPM & PERT- project scheduling, critical path calculations, Crashing.	
4	4.1	Queueing theory -basic structure of queuing systems	12
	4.2	Roles of the Poisson and exponential distributions	
	4.3	Classification of queues basic results of M/M/1: FIFO systems, extension to multi-server queues.	
5	5.1	Simulation: simulation concepts, simulation of a queuing system using event list	12
	5.2	Pseudo random numbers, multiplication congruential algorithm, inverse transformation method	
	5.3	Basic ideas of Monte-Carlo simulation. Montecarlo simulation using windows POM QM software	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

Taha.H.A ,operation Research : An Introduction, McMilan publishing Co., 1982. 7th ed.
Ravindran A, Philips D.T &Solbery.J.J, Operations Research: Principles and practice, John Wiley & Sons, New York, 1987.
Frank S. Budnick, Dennis Mcleavey and Richard Mojena, Principles of Operations Research for Management. All India Traveler Book seller, Delhi.
Gillet.B.E., Introduction to Operations Research - A Computer oriented algorithmic approach, McGraw Hill, 1987.
Hillier.F.S&Liberman.G.J, operation Research, Second Edition, Holden Day Inc, 1974.

MCA202 Java Programming

GENERAL DETAILS OF THE COURSE

Course Code	MCA202		
Course Name	Java Programming		
Course Type	Core		
Credit	4		
Semester	II		
Course Objectives	The course aims to enhance problem-solving abilities, refine program design and coding skills by grasping the object-oriented programming concepts in Java, and applying it to create programs and applications in Java.		
Pre-requisites	Basics of Programming Using C		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	2	0	4

COURSE OUTCOMES (CO)

CO#	CO Description
CO1	To understand the fundamental object-oriented concepts
CO2	To apply object-oriented concepts of Java in solving simple problems
CO3	To practice the usage of various data structures in Java
CO4	To implement the concept of JDBC
CO5	To design a project using Swing

CO-PSO-PO MAPPING MATRIX

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3
CO1	2	2	1								2			2	2
CO2	2	2	1		2						2			2	2
CO3	2	2	2								2			1	2
CO4	2	2	2		2						2				2
CO5	2	2	2		2			1			2			3	2

COURSE CONTENT

Module No		Module Content	Hours Required
1	1.1	Object oriented paradigm-Differences between Object Oriented Programming and Procedure oriented programming	12
	1.2	Basic concepts of Object Oriented Programming, Encapsulation, Benefits of OOP	
	1.3	Java – Introduction, Program structure, Data types, operators, control structures, arrays, functions and recursion	
	1.4	Practical Sessions on Java program structure	
2	2.1	Object oriented concepts in java – Class, Object, Methods	12
	2.2	Inheritance, Polymorphism. Overriding and Overloading	
	2.3	Threads in Java – Thread Model, main thread, Creating threads	
	2.4	Practical Sessions on Java OOPs concepts	
3	3.1	Strings, Vectors, Collections	12
	3.2	Interfaces and Packages	
	3.3	Exception handling – Errors Vs Exceptions, keywords, creating user defined exceptions	
	3.4	Practical Sessions on Packages, Interfaces and Exception Handling	
4	4.1	Java I/O – Files and Directories	12
	4.2	JDBC – Connection class, Statements, Driver Types	
	4.3	ResultSet Meta Data, PreparedStatement, ParameterizedStatement	
	4.4	Practical Sessions on on Java I/O and JDBC	
5	5.1	Swing GUI	12
	5.2	Swing Project	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

JAVA The Complete Reference- Patrick Naughton and Herbert Schidt.- fifth Edition Tata McGraw Hill.
The Complete reference J2SE - Jim Keogh – Tata McGraw Hills
Programming and Problem Solving With Java, Slack, Thomson Learning, 1Edn.
Java Programming Advanced Topics, Wigglesworth, Thomson Learning, 3Edn.
Java Programming, John P. Flynt , Thomson Learning, 2Edn.
Ken Arnold and James Gosling, The Java Programming language, Addison Wesley, 2nd Edition, 1998

Patrick Naughton and Herbert Schidt. The Complete Reference, JAVA fifth Edition Tata McGraw Hill.
Maydene Fisher, Jon Ellis, Jonathan Bruce; JDBC API Tutorial and Reference, ThirdEdition, Publisher:Addison-Wesley
Thinkingjava–BruceEckel–PearsonEducationAssociation

MCA203 Agile Development and Case tools

GENERAL DETAILS OF THE COURSE

Course Code	MCA203		
Course Name	Agile Development and Case tools		
Course Type	Core		
Credit	4		
Semester	II		
Course Objectives	To provide a comprehensive understanding of Agile methodologies and their application in software development. To explore the effective use of CASE (Computer-Aided Software Engineering) tools in improving software quality and productivity.		
Pre-requisites	Basic understanding of software development life cycle (SDLC) and programming concepts.		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	3	0	2

COURSE OUTCOMES (CO)

CO#	CO Description
CO1	Explain the basics of Agile methodologies and how they differ from traditional methods.
CO2	Use Agile tools to plan and manage projects effectively.
CO3	Identify and describe different types of CASE tools and their uses.
CO4	Apply CASE tools to design, develop, and test software.
CO5	Combine Agile methods with CASE tools to improve teamwork and project delivery.

CO-PSO-PO MAPPING MATRIX

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO#1	2	3	2					2	3					1	
CO#2	2	2			2			2						1	
CO#3		2			2			2						1	
CO#4						1			2		3			1	
CO#5						1			3		2			1	

COURSE CONTENT

Module No		Module Content	Hours Required
1	1.1	Introduction: Overview of Agile software development, Agile Manifesto, and principles. Agile Frameworks: Scrum, Kanban, and Extreme Programming (XP).	12
	1.2	Roles in Agile: Scrum Master, Product Owner, Development Team. Agile vs Traditional Development: Comparison between	
	1.3	Agile and Waterfall models. Agile Practices: Iterative development, continuous feedback, and flexibility in project requirements	
2	2.1	Introduction to Agile Project Management: Basics of managing projects using Agile methodologies. Sprint Planning: Creating and managing sprints, setting goals, and defining deliverables.	12
	2.2	Backlog Management: Managing product and sprint backlogs, prioritizing tasks. User Stories: Writing and evaluating user stories, defining acceptance criteria.	
	2.3	Agile Metrics: Burndown charts, velocity tracking, and measuring team performance.	
3	3.1	Introduction to CASE Tools, Types of CASE Tools: Upper CASE (requirements, design), Lower CASE (coding, testing), and Integrated CASE tools.	12
	3.2	CASE Tool Architecture: Overview of CASE tool components and their interaction. Benefits of CASE Tools: Improving productivity, quality, and project management.	
	3.3	Limitations of CASE Tools: Cost, complexity, and resistance to adoption.	
4	4.1	Application of CASE Tools: Requirements Analysis Tools: Tools for gathering, analyzing, and documenting user needs. Design Tools: Tools for creating system models, architectural designs, and UML diagrams.	12
	4.2	Code Generation Tools: Tools that automate code generation from designs or models. Testing Tools: Tools for automating the testing process, including unit and integration testing.	
	4.3	Maintenance Tools: Tools for managing software maintenance, bug tracking, and updates.	
5	5.1	Integration of Agile and CASE Tools: Agile and CASE Tool Integration: Using CASE tools within Agile processes for development and collaboration.	12
	5.2	Continuous Integration and Delivery (CI/CD): Integrating Agile with DevOps practices and CASE tools for automated testing and	

		deployment. Collaborative Tools: Tools like JIRA, Trello, and Git for team collaboration and version control.	
	5.3	Automation in Agile Projects: Automating tasks like builds, testing, and deployments in an Agile environment. Enhancing Productivity: How Agile and CASE tools together streamline project development and management.	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

"Agile Software Development: Principles, Patterns, and Practices" by Robert C. Martin (2002)
"Scrum: The Art of Doing Twice the Work in Half the Time" by Jeff Sutherland (2014)
"Fundamentals of Software Engineering" by Rajib Mall (2004)
"CASE Tools for Software Development" by Ivan Mistrík (2002)
"Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation" by Jez Humble and David Farley (2010)
"Agile Software Development: Principles, Patterns, and Practices" by Robert C. Martin (2002)

MCA204 Design and Analysis of Algorithms

General Details of the Course

Course Code	MCA204		
Course Name	Design and Analysis of Algorithms		
Course Type	Core		
Credit	4		
Semester	II		
Course Objectives	The course aims to equip students with a comprehensive understanding of algorithm design and analysis. It focuses on fundamental concepts, algorithmic strategies, and their practical applications. By the end of the course, students will possess the skills to analyze, design, and implement efficient algorithms for solving a wide range of computational problems.		
Pre-requisites	Data Structures		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	3	1	0

Course Outcomes (CO)

CO#	CO Description
CO1	Analyse the performance of basic algorithms.
CO2	Apply, Analyse and Evaluate Algorithm Design Strategies such as Divide and Conquer, Branch and Bound, Backtracking strategies and solve related problems.
CO3	Apply, Analyse and Evaluate Algorithm Design Strategies such as Dynamic programming, Greedy Strategy and solve related Problems:
CO4	Analyse and Evaluate various Graph Algorithms and Solve related Problems
CO5	Understand NP Hard and NP Complete Problems and Approximation Problems

CO-PSO-PO Mapping Matrix

CO/PSO	PO #1	PO# 2	PO# 3	PO# 4	PO# 5	PO# 6	PO# 7	PO# 8	PO# 9	PO# 10	PO# 11	PO# 12	PSO #1	PSO #2	PSO #3
CO#1		3		2											
CO#2		3	3	2											
CO#3		3	3	2											
CO#4		3	1	2											
CO#5		1		2											

COURSE CONTENT

Module No		Module Content	Hours Required
1	1.1	Introduction: Algorithm, Concepts in performance analysis – space complexity and time complexity, Asymptotic Notations	12
	1.2	Sorting: Analysis of - Bubble sort, Selection sort and Insertion sort.	
	1.3	Searching: Analysis of - Linear Search, Binary Search and Interpolation Search.	
	1.4	Hashing Techniques: Significance of hashing in searching- simple hash functions, handling collisions using chaining-Analysis	
2	2.1	Divide and Conquer Strategy: General method	12
	2.2	Finding the maximum and minimum, Analysis Quick sort and Merge sort	
	2.3	Branch and Bound: Travelling Sales Man Problem	
	2.4	Backtracking: The 8 queen's problem, sum of subsets.	
3	3.1	Dynamic Programming: Introduction, Drawback of Recursion, Elements of Dynamic Programming	12
	3.2	Matrix Chain Multiplication, Longest Common subsequence, 0-1 knapsack problem.	
	3.3	Greedy Algorithms: Elements of Greedy Strategy, Huffman Codes	
	3.4	Activity Selection Problem, fractional knapsack problem	
4	4.1	Graph Algorithms: Breadth First Search, Depth First Search.	12
	4.2	DFS Applications: Strongly Connected Components and Topological Sort	
	4.3	Minimum Spanning tree: Kruskal and Prims algorithms	
	4.4	Shortest path: Single Source Shortest path (Dijkstra's Algorithm) and all pair shortest path	
5	5.1	NP Hard and NP Complete Problems: Basic concepts, non-deterministic algorithm, class of NP- hard and NP- complete	12
	5.2	Approximation Problems: vertex coloring problem – center selection problem, travelling sales man problem.	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). Introduction to Algorithms (3rd ed.). MIT Press.
Dasgupta, S., Papadimitriou, C. H., &Vazirani, U. V. (2006). Algorithms (1st ed.). McGraw-Hill.
Levitin, A. (2011). Introduction to the Design and Analysis of Algorithms. Pearson.
Baase, S., & Van Gelder, A. (2020). Computer Algorithms: Introduction to Design and Analysis (3rd ed.). Addison-Wesley.
Kleinberg, J., &Tardos, É. (2021). Algorithm Design (1st ed.). Pearson.
https://www.javatpoint.com/daa-tutorial

MCA205 Artificial Intelligence with Gen AI

GENERAL DETAILS OF THE COURSE

Course Code	MCA205		
Course Name	Artificial Intelligence with GenAI		
Course Type	Core		
Credit	3		
Semester	II		
Course Objectives	To provide a strong foundation of fundamental concepts in Artificial Intelligence To provide a basic exposition to the goals and methods of Artificial Intelligence To enable the student to apply these techniques in applications which involve perception, reasoning and learning		
Pre-requisites	Probability, Statistics and Computational Mathematics		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	3	0	2

COURSE OUTCOMES (CO)

CO#	CO Description
CO1	Understand the fundamental principles of search algorithms
CO2	Comprehend adversarial search algorithms to optimize game-playing
CO3	Implement Backtracking and Forward Checking algorithms for solving CSP
CO4	Implement and analyze Feedforward Neural Networks (FNN) through a case study.
CO5	Apply sentiment analysis techniques to analyze customer reviews.

CO-PSO-PO MAPPING MATRIX

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2													
CO2	3	2													
CO3	3	2													
CO4	3	2	2	2											
CO5	3	2	2	2											

COURSE CONTENT

Module No		Module Content	Hours Required
1	1.1	Search Algorithms – Heuristics, Optimality & Completeness	12
	1.2	Uninformed search algorithms - Depth-first search, Breadth-first search, Implementation using Python	
	1.3	Heuristic search algorithms - A* algorithm	
	1.4	Local search algorithms - Hill climbing, Simulated annealing	
2	2.1	Adversarial Search and Game Playing-Minimax algorithm for game playing	12
	2.2	Alpha-Beta Pruning	
	2.3	Evaluation functions and heuristics for game playing	
	2.4	Multi-player games and extensions of Minimax	
3	3.1	Constraint Satisfaction Problems (CSP), Constraint propagation	12
	3.2	Backtracking and forward checking	
	3.3	Heuristics in CSP solving	
	3.4	Reinforcement learning concepts - Introduction to reinforcement learning, Markov Decision Processes (MDP), Q-learning and Deep Q Networks (DQN)	
4	4.1	Neural networks and deep learning - Feedforward Neural Networks (FNN) – Case Study	12
	4.2	Transfer learning - Inductive and Transductive transfer learning; Ensemble methods: Bagging, Boosting, Stacking	
	4.3	Natural Language Processing (NLP) – Key concepts	
	4.4	Sentiment analysis – Customer Review Case study	
	4.5	Text summarization and generation – TF-IDF & Page Ranking Algorithm	
5	5.1	Introduction to Generative Models: Overview of generative models, Types of generative models (e.g., GANs, VAEs)	12
	5.2	Autoencoders: Basics of autoencoders, Variational Autoencoders (VAEs)	
	5.3	Generative Adversarial Networks (GANs): Introduction to GAN architecture, Training GANs, Applications of GANs	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

Russell, S., Norvig, P. (2016). Artificial Intelligence: A Modern Approach. (n.p.): CreateSpace Independent Publishing Platform.

Luger, G. F. (2021). Knowing Our World: An Artificial Intelligence Perspective. Germany: Springer International Publishing.
Rich, E. (2019). Artificial Intelligence 3E (Sie). India: Tata McGraw-Hill Publ.
Jackson, P. C. (2019). Introduction to Artificial Intelligence: Third Edition. United States: Dover Publications.
Nilsson, N. J. (2014). Principles of Artificial Intelligence. United States: Elsevier Science.
Artificial Intelligence authors/titles recent submissions (arxiv.org)
AI Weekly – AI News & Leading Newsletter on Deep Learning & Artificial Intelligence
fast.ai – fast.ai – Making neural nets uncool again

MCA206 Full Stack Development

General Details of the Course

Course Code	MCA206		
Course Name	Full Stack Development		
Credit	4		
Semester	II		
Course Objectives	• To develop robust web applications using Django, understanding MVC architecture and ORM • To implement web interfaces with HTML, CSS, and JavaScript, integrating with Django templates. • To deploy and secure Django applications, optimizing performance and applying best security practices.		
Pre-requisites	Basic understanding of software development and version control.		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	0	0	6

Course Outcomes (CO)

CO#	CO Description
CO1	Understand basic web technologies and setup a Django project environment.
CO2	Design and implement database models using Django's ORM for CRUD operations.
CO3	Create dynamic web pages by developing views, templates, and URL routing in Django.
CO4	Construct and validate user forms, manage authentication and user roles in Django applications.
CO5	Deploy Django applications, apply security measures, and optimize performance for production environments.

CO-PSO-PO Mapping Matrix

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3
CO1		2			3									3	
CO2		2			3									3	
CO3		2			3									3	
CO4		2			3									3	
CO5		2			3	3				3				3	

COURSE CONTENT

Module No		Module Content	Hours Required
1	1.1	Overview of Web Technologies: HTML, CSS, JavaScript, and their roles in web development.	10
	1.2	Basics of HTTP and Web Servers: How web servers work, request-response cycle.	
	1.3	Introduction to Django: Overview of Django framework and its architecture (MVT pattern).	
	1.4	Setting Up the Development Environment: Installing Django, setting up a virtual environment, and creating a basic Django project.	
2	2.1	Understanding Django Models: Creating models, migrations, and admin interface.	8
	2.2	Database Setup: Configuring databases (SQLite, PostgreSQL).	
	2.3	CRUD Operations: Creating, reading, updating, and deleting data using Django ORM.	
3	3.1	Creating Views: Function-based views vs class-based views	7
	3.2	Template System: Creating and using templates, template inheritance	
	3.3	URL Routing: Mapping URLs to views, and dynamic URLs.	
4	4.1	Django Forms: Creating and processing forms and form validation.	10
	4.2	Authentication: User authentication, permissions, and user roles	
	4.3	Building Login and Registration Systems: Implementing user registration and login functionalities.	
5	5.1	Deployment: Preparing a Django application for deployment, using platforms like Heroku or AWS.	10
	5.2	Security Best Practices: Securing Django applications, handling common security issues.	
	5.3	Develop socially relevant, small-scale applications aligned with UN Sustainable Development Goals (SDGs). The focus should be on usability and effective database integration. These projects should address critical societal challenges and promote community well-being.	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

Django Official Documentation: https://docs.djangoproject.com/en/5.1/
MDN Web Docs (HTML, CSS, JavaScript): https://developer.mozilla.org/en-US/

SEMESTER III

MCA301 Internet of Things

GENERAL DETAILS OF THE COURSE

Course Code	MCA 401		
Course Name	Internet of Things		
Course Type	Core		
Credit	4		
Semester	III		
Course Objectives	The course aims to equip students with a comprehensive understanding of IoT concepts, hardware components, communication protocols, data analytics, and project development skills, fostering the ability to apply this knowledge in real-world scenarios and address the challenges associated with designing and implementing IoT systems.		
Pre-requisites	Basic programming skills (Python or C/C++), fundamental knowledge of networking concepts, and a grasp of electronics fundamentals to ensure effective engagement with IoT concepts and hands-on exercises.		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	3	1	0

COURSE OUTCOMES (CO)

CO#	CO Description
CO1	Understand IoT concepts, history, key components, and address privacy and security.
CO2	Use IoT hardware (sensors, actuators, microcontrollers) through hands-on exercises with Arduino and Raspberry Pi.
CO3	Implement communication protocols, wireless technologies, and IoT network setup for practical data exchange.
CO4	Apply data analytics tools, visualize data, and use machine learning for meaningful analysis in IoT applications.
CO5	the entire IoT project lifecycle, showcasing creativity and problem-solving in planning, coding, troubleshooting, and presentation.

CO-PSO-PO MAPPING MATRIX

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	P01 0	PO1 1	PO1 2	PSO 1	PSO 2	PSO 3
CO#1	2		2		3										
CO#2	2		2		3										
CO#3	2		3		3										
CO#4	2		2		3					3					
CO#5	2		3		3						2			3	

COURSE CONTENT

Module No		Module Content	Hours Required
1	1.1	Overview of IoT concepts and applications.	12
	1.2	Evolution and history of IoT.	
	1.3	Key components of IoT systems (devices, connectivity, platforms), IoT architectures and communication protocols.	
	1.4	Privacy and security considerations in IoT.	
2	2.1	Introduction to IoT hardware components (sensors, actuators, microcontrollers).	12
	2.2	Understanding sensor types (temperature, humidity, motion, etc.)	
	2.3	Basics of Arduino and Raspberry Pi for IoT projects, Connecting sensors and actuators to microcontrollers.	
	2.4	Hands-on exercises on basic sensor interfacing.	
3	3.1	Basics of communication protocols (MQTT, CoAP, HTTP).	12
	3.2	Wireless communication technologies for IoT (Wi-Fi, Bluetooth, Zigbee).	
	3.3	Setting up IoT networks, Introduction to IoT cloud platforms.	
	3.4	Hands-on exercises on sending and receiving data over IoT networks.	
4	4.1	Introduction to data analytics in IoT, Basics of data visualization tools.	12
	4.2	Analysing and interpreting IoT data.	
	4.3	Machine learning for IoT applications.	
	4.4	Hands-on exercises on basic data analytics using IoT data.	
5	5.1	Overview of IoT project lifecycle.	12
	5.2	Planning and designing an IoT project Implementation and coding for IoT applications.	
	5.3	Testing and troubleshooting IoT projects.	
	5.4	Final project presentation and evaluation.	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

IoT: Concepts and Applications - Rajkumar Buyya, Christian Vecchiola, S. Thamarai Selvi.
Building the Internet of Things: Implement New Business Models, Disrupt Competitors, Transform Your Industry - Maciej Kranz.
IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things - David Hanes, Gonzalo Salgueiro, Patrick Grossetete.
Hands-On IoT Solutions with Blockchain: Build effective blockchain IoT projects using Hyperledger and Ethereum - Arvind Ravulavaru.
Practical Internet of Things Security - Brian Russell, Drew Van Duren.
Python Programming for Arduino - Pratik Desai.
IoT for ALL: A comprehensive platform covering various aspects of IoT, including articles, case studies, and trends - https://www.iotforall.com/
IEEE IoT Initiative: Explore the latest research, standards, and resources from the Institute of Electrical and Electronics Engineers (IEEE) - https://iot.ieee.org/
Arduino Official Website: Access official documentation, tutorials, and community forums for Arduino development - https://www.arduino.cc/
Raspberry Pi Foundation: The official website provides documentation and resources for Raspberry Pi development - https://www.raspberrypi.org/
MQTT.org: Learn about the MQTT (Message Queuing Telemetry Transport) protocol, a key communication protocol in IoT - http://mqtt.org/
Kaggle: Kaggle offers datasets and competitions related to IoT and data analytics, providing practical examples for learners - https://www.kaggle.com/

MCA302 Theory of Computation and Compilers

GENERAL DETAILS OF THE COURSE

Course Code	MCA302		
Course Name	Theory of Computation and Compilers		
Course Type	Core		
Credit	4		
Semester	III		
Course Objectives	To understand the basic mathematical model of computation To assess the working of a compiler.		
Pre-requisites	Knowledge in Programming languages		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	3	1	0

COURSE OUTCOMES (CO)

CO#	CO Description
CO1	Demonstrate knowledge of basic mathematical models of computation and describe how they relate to formal languages.
CO2	Understanding the concept of pushdown automata and context free grammar.
CO3	Understand the phases of a compiler.
CO4	Analyse various parsing techniques.
CO5	To apply the design and implementation of parsers.

CO-PSO-PO MAPPING MATRIX

CO/PSO	PO #1	PO #2	PO #3	PO #4	PO #5	PO #6	PO #7	PO #8	PO #9	PO #10	PO #11	PO #12	PSO #1	PSO #2	PSO#3
CO#1	3						1				2	2		2	
CO#2		3					1				2	2		2	
CO#3	3	2					1				2	2		2	
CO#4		3					1				2	2		2	
CO#5		2					1				2	2		2	

COURSE CONTENT

Module No		Module Content	Hours Required
1	1.1	Formal Language, Non-Computational Problems, Diagonal Argument, Russels's Paradox	12
	1.2	Deterministic Finite Automaton (DFA), Non-Deterministic Finite	

		Automaton (NDFA), Equivalence of DFA and NDFA	
	1.3	Regular Languages, Regular Grammars, Regular Expressions	
	1.4	Properties of Regular Language, Pumping Lemma, Non Regular Languages.	
2	2.1	Pushdown Automaton (PDA), Non-Deterministic Pushdown Automaton (NPDA)	12
	2.2	Context Free Grammar, Chomsky Normal Form, Greibach Normal Form	
	2.3	Ambiguity, Parse Tree Representation of Derivation Trees	
	2.4	Equivalence of PDA's and Context Free Grammars; Properties of Context Free Language. Lab Exercises	
3	3.1	Introduction to compiling, Compilers	12
	3.2	Analysis of a source program, the phases of a compiler	
	3.3	Lexical analysis:-The role of the lexical analyser, Input buffering, specification of tokens	
	3.4	Recognition of tokens.	
4	4.1	Syntax analysis: - the role of the parser, Top down parsing, Bottom up parsing	12
	4.2	Syntax directed translation, syntax directed definition, Construction of Syntax Tree	
	4.3	LL parsers, Operator precedence grammar	
	4.4	LR(0) , SLR parser, LALR(1) parser.	
5	5.1	Intermediate code generation-postfix notation, syntax tree	12
	5.2	three-address code, basic blocks and flow graph, Back patching	
	5.3	Code optimization: - The principal sources of optimization, optimization of basic blocks, loops in flow graphs	
	5.4	Peephole optimization Code Generations: - Issues in the design of a code generator	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

Peter Linz, An Introduction to Formal Languages and Automata, Third Edition, Jones and Bartlett, 2001.
Introduction to Automata Theory, Languages, and Computation By John E. Hopcroft,Rajeev Motwani and Jeffrry D Ullman.
Compilers Principles, Techniques and Tools- Alfred V Aho, Ravi Sethi, Jeffrry D Ullman
Steven S Muchnik, "Advanced Compiler Design and Implementation"

MCA303 Machine Learning

GENERAL DETAILS OF THE COURSE

Course Code	MCA303		
Course Name	Machine Learning		
Course Type	Core		
Credit	4		
Semester	III		
Course Objectives	The course will teach students the basics of machine learning algorithms, including supervised, unsupervised, and deep learning, along with the math behind them. Students will practice using popular tools like scikit-learn, TensorFlow, Keras, Orange, and NLTK to build and test machine learning models. The course will help students learn how to work with natural language processing techniques, such as text processing and sentiment analysis, using tools like NLTK, SpaCy, and TensorFlow.		
Pre-requisites	Basic mathematics		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	3	0	2

COURSE OUTCOMES (CO)

CO#	CO Description
CO1	Understand and apply basic data mining and machine learning concepts, including data preprocessing techniques and data visualization methods.
CO2	Implement and evaluate various supervised learning algorithms, including regression and classification, using tools like scikit-learn and Orange.
CO3	Apply unsupervised learning algorithms, such as clustering and dimensionality reduction techniques, and evaluate their effectiveness using scikit-learn and Orange.
CO4	Implement advanced machine learning techniques, such as ensemble methods and neural networks, and optimize models using tools like TensorFlow and Keras.
CO5	Apply natural language processing techniques, such as text preprocessing and sentiment analysis, using tools like NLTK, SpaCy, and TensorFlow.

CO-PSO-PO MAPPING MATRIX

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3
CO1		3			2								3		2
CO2		3			1								3		2
CO3		2			2								3		2
CO4		3			1								3		2
CO5		3			2								3		2

Course Content

Module No		Module Content	Hours Required
1	1.1	Data Warehousing, Multidimensional Data Model, OLAP Operations, Knowledge Discovery in Databases (KDD) Process, Data Mining Functionalities, Classification of Data Mining Systems	12
	1.2	Data Preprocessing (Data Cleaning, Integration, Transformation, Reduction, Discretization, Concept Hierarchy Generation)	
	1.3	Exploring Data and Visualization Techniques (Visualizing Higher-Dimensional Data).	
	1.4	Tools: Python (Matplotlib, Seaborn), Orange	
2	2.1	Regression (Linear Regression, Nonlinear Regression, Lasso Regression, Ridge Regression)	12
	2.2	Classification (Decision Tree Induction, Attribute Selection Measures, Pruning, Naïve Bayesian Classification, Rule-Based Classification, K-Nearest Neighbour Classifiers, Support Vector Machines (SVM)),	
	2.3	Evaluation of Supervised Learning Models (Accuracy, Precision, Recall, F1-Score, Cross-Validation), Tools: scikit-learn, Orange.	
3	3.1	Clustering (Partitioning Methods: K-Means, K-Medoids, CLARANS), Hierarchical Methods (Agglomerative, Divisive)	12
	3.2	Density-Based Clustering (DBSCAN), Grid-Based Clustering (STING), Dimensionality Reduction (Principal Component Analysis (PCA), t-SNE).	
	3.3	Evaluation of Clustering Methods	
	3.4	Tools: scikit-learn, Orange.	
4	4.1	Ensemble Methods (Bagging, Boosting: AdaBoost, Gradient Boosting, Random Forests), Neural Networks (Perceptrons, Multi-Layer Perceptrons (MLPs.	12
	4.2	Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs)),	
	4.3	Optimization Techniques (Gradient Descent Variants,	

		Hyperparameter Tuning: Grid Search, Random Search),	
	4.4	Tools: TensorFlow, Keras, XGBoost	
5	5.1	Introduction to NLP, Text Preprocessing (Tokenization, Stopword Removal, Lemmatization, Stemming)	12
	5.2	Text Representation (Bag-of-Words, TF-IDF, Word Embeddings: Word2Vec, GloVe), Sentiment Analysis, Named Entity Recognition, Topic Modeling.	
	5.3	Tools: NLTK, SpaCy, scikit-learn, TensorFlow.	

References (Text books/ Researches/ Online URLs)

Tom Mitchell, Machine Learning, McGraw Hill, 1997.
Christopher M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.
Jiawei Han and Micheline Kamber, Data Mining: Concepts and Techniques, Elsevier, 2nd Edition.
Aurélien Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, O'Reilly Media, 2nd Edition.
Foundations of Statistical Natural Language Processing by Christopher D. Manning and Hinrich Schütze.

MCA304 Cyber Security Essentials

GENERAL DETAILS OF THE COURSE

Course Code	MCA304		
Course Name	Cyber Security Essentials		
Course Type	Core		
Credit	4		
Semester	III		
Course Objectives	This course aims to equip students with the knowledge and skills to implement effective digital security measures, including protecting devices, securing networks, and mitigating cyber threats.		
Pre-requisites	Higher Secondary Level Mathematics		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	3	1	0

COURSE OUTCOMES (CO)

CO#	CO Description
CO1	Understanding of Digital Security Fundamentals: to demonstrate a comprehensive understanding of digital security principles, including protecting personal computers and devices from various threats such as viruses and malware.
CO2	Proficiency in Identity and Access Management: gain proficiency in identity, authentication, and authorization mechanisms, recognizing the importance of strong credentials and implementing measures to keep them secure.
CO3	Competence in Server Security: will be able to apply the principles of protecting servers through both physical and logical security measures, ensuring the confidentiality, integrity, and availability of data stored on servers.
CO4	Knowledge of Web Security Protocols and Practices: understanding of the World Wide Web, the HTTP protocol, and the security measures required to safeguard browser-to-web server interactions, mitigating risks associated with cyber-attacks such as SQL injection, cross-site scripting, and denial of service.
CO5	Proficiency in Cybersecurity Operations: develop skills in operations security, including monitoring, identifying threats, and effectively remediating them. They will also gain an understanding of data security principles, data privacy, and compliance standards, along with ethical considerations in computer security practices.

CO-PSO-PO MAPPING MATRIX

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO#1	2	3	2												2
CO#2	2	2	3												2
CO#3	3														1
CO#4	1					3				1					3
CO#5	1	2			3										2

COURSE CONTENT

Module No		Module Content	Hours Required
1	1.1	Basics of digital security, protecting personal computers and devices, protecting devices from Virus and Malware	12
	1.2	Identity, Authentication and Authorization, need for strong credentials, keeping credentials secure	
	1.3	Protecting servers using physical and logical security,	
	1.4	World Wide Web (www), the Internet and the HTTP protocol, security of browser to web server interaction	
2	2.1	Introduction to cyber-attacks, application security (design, development and testing),	12
	2.2	Operations security, monitoring, identifying threats and remediating them,	
	2.3	Principles of data security - Confidentiality, Integrity and Availability,	
	2.4	Data Privacy, Data breaches, preventing attacks and breaches with security controls	
3	3.1	Compliance standards, Computer Ethics.	12
	3.2	OWASP Top 10: Types of Web attacks- SQL Injection, Cross site scripting,	
	3.3	Brute Force, Buffer Overflow, Man in the middle attack,	
	3.4	Denial of Service	
4	4.1	Hash Functions and MAC: Properties of hash functions, birthday attack, hash cash	12
	4.2	Message Authentication code Algorithms, MAC protocols, HMAC, CMAC.	
	4.3	Digital Signature: Classification of signature schemes: RSA Signature, Digital Signature Standard,	
5	5.1	Overview of ELGamal and Schnorr schemes, One-time signature schemes, Attacks on Digital Signatures, Blind Signatures	12

	5.2	Firewalls, packet filter and stateful firewalls, application aware firewalls, personal firewalls-IPtables, Proxies, NAT	
	5.3	Intrusion Detection System-Snort, Signature and Anomaly based detection,	
	5.4	Honeypots and Honeynets. Network Log management-syslog or SPLUNK;	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

Sammons, John, and Michael Cross. The basics of cyber safety: computer and mobile device safety made easy. Elsevier, 2016.
Charles P. Pfleeger, Shari Lawrence, Pfleeger Jonathan Margulies; Security in Computing, Pearson Education Inc . 5th Edition, 2015
Bryan Sullivan and Vincent Liu, Web Application Security, A Beginner's Guide, McGraw-Hill Education, 2012
Brooks, Charles J., Christopher Grow, Philip Craig, and Donald Short.Cybersecurity essentials. John Wiley & Sons, 2018
Introduction to Modern Cryptography Mihir Bellare1 Phillip ogaway May 11, 2005
Handbook of applied cryptography, by A. Menezes, P. Van Oorschot, and S. Vanstone, CRC Press, 1996.
Stallings, W., Cryptography and Network Security. Principles and Practice, 4th edition, Prentice Hall.

MCA306 Data Analytics using Python

GENERAL DETAILS OF THE COURSE

Course Code	MCA306		
Course Name	Data Analytics using Python		
Course Type	Core		
Credit	3		
Semester	III		
Course Objectives	To provide an understanding of programming concepts using Python To learn the underlying concepts of Data science and implement using python		
Pre-requisites	NIL		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	0	0	6

COURSE OUTCOMES (CO)

CO #	CO Description
CO1	Understand the basic syntax and principles of Python programming language.
CO2	Ability to understand object-oriented programming concepts and write programs in python. Handling Errors and Exceptions
CO3	Ability to design and develop database applications
CO4	Ability to solve data analysis problems using python
CO5	Analyze complex datasets using advanced Python techniques and algorithms for deeper insights

CO-PSO-PO MAPPING MATRIX

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	P01 0	PO1 1	PO1 2	PSO 1	PSO 2	PSO 3
CO#1	3													1	3
CO#2			2		3									1	
CO#3			2										2		
CO#4		2	2		3								2		
CO#5	3													1	3

COURSE CONTENT

Module No	Module Content	Hours Required

1	1.1	Introduction to Python: - using the Python interpreter, Overview of programming in Python, Expressions and Variables-String Operations.	9
	1.2	Python Data Structures: lists & Tuple –Sets - Dictionaries.	
	1.3	Programming Fundamentals: Conditions and Branching- Loops Functions:	
	1.4	formal arguments, variable-length arguments.	
2	2.1	Classes, files and modules Introduction to Classes and Objects:	9
	2.2	-classes, class attributes, instances, instance attributes, binding and method invocation, inheritance, polymorphism, Built-in functions for classes and instances.	
	2.3	Files and input/output, reading and writing files, methods of file objects using standard library functions,	
	2.4	dates and times Exceptions, detecting and handling exceptions.	9
3	3.1	Database and web programming Python database application programmer's interface (DB- API),	
	3.2	connection and cursor objects,	
	3.3	Type objects and constructors, python database adapters.	
	3.4	Creating simple web clients, introduction to CGI, CGI module, building CGI applications.	
4	4.1	Introduction to Data Science using Python Python libraries: Numpy- Scikit- Pandas-Matplotlib. - Data Visualization.	9
	4.2	Importing Datasets: Importing and Exporting Data in Python-Basic Insights from Datasets.	
	4.3	Data cleansing and pre-processing: Identify and Handle Missing Values.	
	4.4	Summarizing the Data Frame: Descriptive Statistics- Basic of Grouping- ANOVACorrelation	
5	5.1	Model Development and Evaluation Regression Models:	9
	5.2	Linear Regression (SLR & MLR) - Logistic Regression-Decision Tree- Random Forest.	
	5.3	Clustering Techniques: K means clustering – Apriori algorithm.	
	5.4	Model Evaluation: Over-fitting, Underfitting.	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

Core Python Programming by Wesley J. Chun, 2nd Edition, Pearson Education
An Introduction to Python by Guido Van Russom, Fred L.Drake, Network Theory Limited.
Beginning Python: From Novice To Professional By Magnus Lie Hetland, Second Edition Apress

Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython ,2nd edition, Wes McKinney, O'Reilly Media (2017)
Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems ,AurélienGéron, O'Reilly Media (2017)
Data Science from Scratch: First Principles with Python, Joel Grus, O'Reilly Media (2015)

MCA307 Android

GENERAL DETAILS OF THE COURSE

Course Code	MCA307		
Course Name	Android		
Course Type	Lab		
Credit	2		
Semester	III		
Course Objectives	● To create apps based on android platforms ● To access and work with databases under the Android operating system ● To create apps for solving real time problems		
Pre-requisites	● Object oriented programming using Java ● Operating Systems ● Data Base Management Systems		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	0	0	4

COURSE OUTCOMES (CO)

CO#	CO Description
CO1	Able to develop simple apps by understanding the architecture of android operating system
CO2	Able to implement apps based on different types of menus for navigating between activities
CO3	Make decision to solve a problem using package, library and threads Handling Errors and Exceptions
CO4	Ability to analyze, design and develop database applications
CO5	Able to design and create mobile applications works with internet applications

CO-PSO-PO MAPPING MATRIX

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	P01 0	PO1 1	PO1 2	PSO 1	PSO 2	PSO 3
CO#1			1		3						2			2	
CO#2			1		3						2			2	
CO#3			2								2			1	
CO#4			2		3						2			3	
CO#5			2		3						2			3	

COURSE CONTENT

Module No		Module Content	Hours Required
1	1.1	Mobile Computing & Development Introduction: Mobile system architecture and development challenges	5
	1.2	The Android Platform: Android SDK Features, Introduction to the Development Framework, Android Development Tools	
	1.3	Android Application Life Cycle, Activity, Service, Intent, MVC and User Interfaces	
	1.4	Application Structure: AndroidManifest.xml, uses-permission & uses-sdk, Resources & R.java, Assets, Layouts & Drawable Resources, Activities and Activity lifecycle	
2	2.1	Android Graphical User Interface: Linear Layout, Relative Layout, Table Layout, Grid View, Tab Layout, List View, Custom List View Element	5
	2.2	Fragments.	
	2.3	Popup Menus:- Option menu , Context menu,	
	2.4	Sub menu, menu from xml, menu via code.	
3	3.1	Intents – Explicit Intents, Implicit intents, intents and broadcast receivers, intent filters.	5
	3.2	Adapters and Widgtes:-ArrayAdapters, BaseAdapters, ListView and ListActivity.	
	3.3	Notifications: Broadcast Receivers,	
	3.4	Services – Types of services	
4	4.1	Introduction to Content Providers – Types of content providers	5
	4.2	Databases and Content Providers: - SQLite Databases: Basics of SQLite DB, Various Data Types	
	4.3	SQLite Queries, Adding / Updating / Deleting Contents of SQLite	
	4.4	Android Webserver communication with external database	
5	5.1	Advanced Features: JSON Parsing.	5
	5.2	Maps, GPS, Location based Services	
	5.3	Accessing Phone services (Call, SMS, MMS), Network connectivity services	
	5.4	Hardware Sensors: - Sensors and Sensor Managers, Monitoring device movement and orientation, Environmental sensors	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

Professional Android 4 application development – Reto Meier
Android Wireless Application Development By Lauren Darcey and Shane Conder, Pearson Education, 2nd ed.
Beginning Android Application Development By Wei-Meng Lee, Wrox Publication
Unlocking Android Developer's Guide By Frank Ableson and Charlie Collins and Robi Sen, Manning Publication Co.
Android Cookbook by Ian F. Darwin:
Android Programming for Beginners by John Horton
Android Programming: The Big Nerd Ranch Guide by Bill Phillips and Chris Stewart
Head First Android Development by Dawn Griffiths and David Griffiths
https://www.javatpoint.com/android-tutorial
https://developer.android.com/guide
https://www.tutorialspoint.com/android/index.htm

SEMESTER IV

MCA 401 Parallel Computing

GENERAL DETAILS OF THE COURSE

Course Code	MCA401		
Course Name	Parallel Computing		
Course Type	Core		
Credit	4		
Semester	IV		
Course Objectives	To give an overview of modern parallel computer architectures and parallel processing techniques and their applications from, basic concepts to state-of-the-art computer systems and fundamentals, design complexity, power, and reliability at all levels basic parallel programming concepts using OpenMP, MPI and CUDA		
Pre-requisites	MCA102, MCA103		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	3	1	0

COURSE OUTCOMES (CO)

CO#	CO Description
CO1	To comprehend the working of the parallel architectures
CO2	To parallel solve complex problems using task/channelmodel
CO3	To implement Message passing model in parallel programs.
CO4	To implement shared memory model in parallelprograms
CO5	To learn and implement Basic Parallel programs

CO-PSO-PO MAPPING MATRIX

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO#1	1		1	2											
CO#2	2		1	3											
CO#3	2		1												
CO#4			1	3	3										
CO#5			1	3											

COURSE CONTENT

Module No		Module Content	Hours Required
1	1.1	Fundamentals of Parallel Computing Hardware Introduction to Parallel Computing	12

	1.2	Flynn's Taxonomy- SISD, SIMD, MISD, MIMD.	
	1.3	Pipelining, Architectures. Interconnection Networks, Shared vs. Distributed Memory Systems	
2	2.1	Parallel Algorithm Design and Performance Models Parallel Algorithm Design - Foster's Design Methodology	12
	2.2	Decomposition and Mapping Techniques Boundary Value Problem, Finding the maximum, n-body problem	
	2.3	Parallelism- Data Level, Instruction level, Thread Level, Cache Coherence-Directory based Protocol.	
	2.4	Amdahl's Law and Gustafson's Law	
3	3.1	Message passing Model, MPI, MPI_Init, MPI_Finalize, MPI_comm_rank, MPI_comm_Size, MPI_reduce, MPI_Wtime, MPI_Circuit satisfiability.	12
	3.2	OpenMP- Pragma- Parallel for-private- firstprivate-lastprivate-critical-reduction-inverting loop- conditionally executing loop	
	3.3	scheduling loop- single- nowait-section, omp_get_thread_num, omp_get_num_threads	
4	4.1	Shared Memory Model in parallel Programming, Fork- Join Concept	12
	4.2	Introduction to Shared Memory Programming with OpenMP Parallel Directives and Work-sharing Constructs Synchronization (Critical, Atomic, Barriers)	
	4.3	Basic OpenMP Programs (Loop Parallelization, Task Parallelism)	
5	5.1	Introduction to Free Parallel Programming Tools	12
	5.2	Hands-on Lab Exercises: Parallelizing Simple Algorithms	
	5.3	Familiarize the tools: Visual Studio Code with MPI and OpenMP extensions. Eclipse Parallel Tools Platform (PTP) – IDE for parallel application development.	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

Shameem Akhter and Jason Roberts, "Multi-core Programming", Intel Press, 2006.
Michael J Quinn, Parallel programming in C with MPI and OpenMP, Tata Macgraw Hill,2003.
John L. Hennessey and David A. Patterson, " Computer architecture – A quantitative approach", Morgan Kaufmann/Elsevier Publishers, 4th. Edition, 2007.
David E. Culler, Jaswinder Pal Singh, "Parallel computing architecture: A hardware/software approach" , Morgan Kaufmann/Elsevier Publishers, 1999.
Parallel Programming with MPI By Peter S. Pacheco
Using MPI: Portable Parallel Programming with the Message-Passing Interface, By William Gropp, Ewing Lusk, Anthony Skjellum

Web R	https://www.tutorialspoint.com/cuda/index.htm
Web R	https://www.nvidia.com/docs/IO/116711/sc11-cuda-c-basics.pdf

MCA403 Comprehensive Viva Voce

Course Code	MCA403
Course Name	Comprehensive Viva Voce
Course Type	Core
Credit	3
Semester	IV
Course Details	Will be conducted at the end of Semester. A comprehensive Viva based on subjects learned during the course, by Internal Examiner for internal Evaluation and by an external Examiner

MCA404 Internship

Course Code	MCA404
Course Name	Internship
Course Type	Core
Credit	10
Semester	IV

COURSE OUTCOMES (CO)

CO#	CO Description
CO1	Assist the student's development of employer-valued skills such as teamwork, communications and attention to detail.
CO2	Expose the student to the environment and expectations of performance on the part of Software Analysts/developers in professional practice, private/public companies or government entities.
CO3	Enhance and/or expand the student's knowledge of a particular area(s) of Software Engineering
CO4	Expose the student to professional role models or mentors who will provide the student with support in the early stages of the internship and provide an example of the behaviours expected in the intern's workplace.

CO-PSO-PO MAPPING MATRIX

CO/PSO	PO #1	PO #2	PO #3	PO #4	PO #5	PO #6	PO #7	PO #8	PO #9	PO #10	PO #11	PO #12	PSO#1	PSO#2	PSO#3
CO#1		2	2		3	3		2			3	1			
CO#2		2	2		3	3		2			3	1			
CO#3		2	2		3	3					3	1			
CO#4		2	2		3	3			2	3	3	1			

COURSE CONTENT

Course Description
The MCA Internship Course allows MCA students to gain practical experience in the workplace before receiving their Graduation Degrees. The student identifies companies willing to hire him/her on a full-time basis for an 8-week period (minimum required) during their last semester. Responsibilities of an Intern Work closely with teams at the workplace to facilitate the rapid development of high-quality applications which may include: Develop quality software and web applications Analyze and maintain existing software applications Design & implement highly scalable, testable code Discover and fix programming bugs Contribute to the design strategy of the UI and UX of the platform
Internship Guidelines:
Step 1: Request Letter/ Email from Internship Coordinator of the college should go to industry to allot various slots of 8 weeks as internship periods for the students. Students request letter/profile/ interest areas may be submitted to industries for their willingness for providing the training. Step 2: Industry will confirm the training slots and the number of seats allocated for internships via Confirmation Letter/ Email to the Internship Coordinator Step 3: Students on joining Training at the concerned Industry / Organization, submit the Joining Report/Letters / Email. Step 4: Students undergo industrial training at the concerned Industry / Organization. In-between Faculty Member(s) evaluate(s) the performance of students once/twice by visiting the Industry/Organization/ through Online Interactions and Evaluation Report of the students is submitted in department office with the consent of Industry persons Step 5: Students will submit training report after completion of internship along with the Attendance Log to the Internship Coordinator. Step 6: Training Certificate to be obtained from industry and a copy to be submitted to the Office of the Coordinator. Step 7: Assessment of the Internship Outcomes through a Comprehensive Viva and extensive evaluation of the Internship Report.
INTERNSHIP REPORT GUIDELINES
Every student is required to write an Internship report upon completion of their internship and required to submit **two copies** (student copy + department copy in pdf) of the report to Internship Coordinator (along with certificate given by the company) for final evaluation and awarding of Credit Scores. Before submitting the report to the Internship Coordinator, the student required to go through multiple rounds of revision in collaboration with the department internship mentor/coordinator/supervisor. The Internship Report serves multiple purposes: Help the student develop written communication skills. Serve as an archival record of the internship experience.

Give the student an opportunity to reflect on the professional aspects of the internship
experience and the skills that were learned.
Allow the student to describe the science content of the internship.
Have the student to reflect on the initial goals of the internship and how they were (or were not) achieved during the internship.

Text Format in the report:

Cambria 12 or similar, with 1.5 line spacing.
Margins 1.5" left and 1" all other side.

Binding & report length:

Soft binding & report length of minimum 20 pages with one side printing with a designed Cover Page

General information:

Student is eligible for internship evaluation if only if he/she completed 8 weeks of internship training. (Minimum of 40 Working days)

EACH INTERNSHIP REPORT WILL FOLLOW THE FORMAT DESCRIBED:

Title Page
College certificate Page
Internship certificate provided by the internship institution

Acknowledgement

Executive summary/Abstract (2 pages) A
paragraph each on:
The company
The problem or opportunity
Methodology

Key parts of the report & your findings and solutions provided in the report.
Benefits to the company/institution through your report.

Index

List of the contents of the internship report and where they can be found in the report.

Learning Objectives/Internship Objectives

A single page that lists the original objectives of the internship.

Weekly overview of internship activities
Introduction (2 or 3 pages)

The introduction should include a description of the internship site and the scope of the work completed during the internship. This Section may include a detailed explanation of the Organization and their scope of Work. It may include background information necessary to understand the work completed during the internship.

Internship Discussion

This section contains a discussion of the internship and should address the following points:

How the objectives achieved?

What skills (scientific and professional) were learned during the internship?
Results/observations/work experiences get in the internship company.

What challenges did you experience during the internship?

Conclusion

Bibliography

Include references to books, articles, reports referred to in the report.

Note: A handbook with the formats of Certificate and Details will be given at the start of Semester IV.

ELECTIVE COURSES

TheElectiveCoursesareofferedinspecializationtracksfocusedonthreemajordomains

1. AI and Machine Learning
2. Advanced Programming
3. Computer Security

Elective courses are offered through Semester 3 and Semester 4 as **MCA3XX** and **MCA4XX**, respectively. The specializations are offered as two pools for each semester. The courses are offered through majority selection by the students, in consensus with the department, and approved by the Dean of Computer Science.

AI AND MACHINE LEARNING

MCA311 Deep Learning

GENERAL DETAILS OF THE COURSE

Course Code	MCA311		
Course Name	Deep Learning		
Course Type	Specialization		
Credit	4		
Semester	III		
Course Objectives	The course aims to equip students with a comprehensive understanding of deep learning fundamentals, including neural network basics, TensorFlow implementation, and advanced topics like Convolutional and Recurrent Neural Networks, Generative Adversarial Networks, and Transfer Learning, fostering the ability to apply these techniques to real-world problem-solving		
Pre-requisites	• Data Mining • Numerical Analysis • Artificial Intelligence • Data Analytics using Python		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	3	1	0

COURSE OUTCOMES (CO)

CO#	CO Description
CO1	To understand basic concepts of Deep Learning
CO2	To learn Tensor flow and basic functionalities
CO3	To evaluate problems using convolutional neural networks and recurrent Neural Networks
CO4	To build and train different deep learning architectures and reinforce their

	theoretical knowledge.
CO5	To practically implement the deep learning algorithms using Tensor flow

CO-PSO-PO MAPPING MATRIX

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	P01 0	PO1 1	PO1 2	PSO 1	PSO 2	PSO 3
CO#1	1	2		3	3								3		
CO#2	1	2		3	3								3		
CO#3	1	2		3	3								3		
CO#4	1	2		3	3								3		
CO#5	1	2		3	3								3		

COURSE CONTENT

Module No		Module Content	Hours Required
1	1.1	Deep Learning Fundamentals – Neural networks basics	12
	1.2	Basics: Biological Neuron, Idea of computational units- Artificial Neural Network	
	1.3	Perceptron's and activation functions – Feedforward and backpropagation	
	1.4	Convergence theorem for Perceptron Learning Algorithm, Single Layer Perceptron, Multi-layer Perceptron's	
2	2.1	Tensorflow, simple ML examples. Basic operations, constants, variables, Control dependencies	12
	2.2	Data pipeline, TensorBoard	
	2.3	Linear and Logistic Regression, Tensorflow's	
	2.4	Hands on implementation on Optimizers, tf.data-Birth rate – life expectancy, MNIST dataset	
3	3.1	Loss Functions and Optimization – Image features,	12
	3.2	Optimization, stochastic gradient descent algorithm's	
	3.3	Convolutional Neural Networks, Convnet in TensorFlow- image classification	
	3.4	Solving a problem with CNNs on Tensorflow.	
4	4.1	Recurrent Neural Networks, Language modelling	12
	4.2	Image captioning, Soft attention Back propagation through time	
	4.3	Long Short-Term Memory, LSTMs, Bidirectional RNNs	
	4.4	Solving a problem with RNNs on Tensorflow	
5	5.1	Generative Adversarial Networks (GANs) and Transfer Learning	12

	5.2	GANs for image generation – Transfer learning for improved model performance	
	5.3	Tensorflow Regression implementation	
	5.3	Tensorflow Classification implementation	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

Goodfellow, Y. Bengio, A. Courville, Deep Learning, MIT Press, 2016. http://www. deeplearningbook.org.
K. P. Murphy, Machine Learning: A Probabilistic Perspective, MIT Press, 2012.
C. M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.
Neural Networks and Deep Learning by Michael Nielsen, Online
Hands-On Machine Learning with Scikit-Learn and TensorFlow, by AurélienGéron

MCA312 Inferential Statistics

GENERAL DETAILS OF THE COURSE

Course Code	MCA414		
Course Name	Inferential Statistics		
Course Type	Specialization		
Credit	4		
Semester	III		
Course Objectives	This course provides a thorough understanding of inferential statistics and its application in real-world data analysis. Students will learn to draw conclusions about populations from sample data, applying various statistical methods to test hypotheses and make predictions.		
Pre-requisites	A basic understanding of descriptive statistics, probability theory, and algebra		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	3	1	0

COURSE OUTCOMES (CO)

CO#	CO Description
CO1	Understand the key concepts and methods of inferential statistics and their applications.
CO2	Calculate and interpret confidence intervals for different population parameters.
CO3	Conduct hypothesis tests, including one-sample and two-sample tests, and evaluate the results.
CO4	Apply ANOVA techniques to compare means across multiple groups and understand interaction effects.
CO5	Apply regression and non-parametric methods for modeling and analyzing data, making accurate predictions and decisions.

CO-PSO-PO MAPPING MATRIX

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO#1		3											2		
CO#2		2											2		
CO#3		3											2		
CO#4		2			2								2		

CO#5		2										2		

COURSE CONTENT

Module No		Module Content	Hours Required
1	1.1	Introduction to Inferential Statistics	12
	1.2	Overview of statistics: Descriptive vs. Inferential Statistics	
	1.3	Populations and samples, Sampling methods and sampling distributions	
	1.4	Central Limit Theorem and its implications, Introduction to statistical software tools	
2	2.1	Estimation and Confidence Intervals Point estimates and interval estimates	12
	2.2	Confidence intervals for population mean, proportion, and variance	
	2.3	Confidence intervals for large and small samples Margin of error and significance level, Practical examples and applications	
3	3.1	Hypothesis Testing- Introduction to hypothesis testing Null and alternative hypotheses	12
	3.2	Types of errors (Type I and Type II errors), Test statistics and p-value	
	3.3	One-sample and two-sample hypothesis tests (for means and proportions)	
4	4.1	Analysis of Variance (ANOVA), One-way ANOVA: Assumptions and methodology	12
	4.2	Two-way ANOVA and interaction effects, Post-hoc tests (Tukey's HSD)	
	4.3	F-statistic and its interpretation, Applications of ANOVA in experimental design	
5	5.1	Non-Parametric Methods and Regression Analysis- Introduction to non-parametric tests: Chi-square test, Mann-Whitney U test, Kruskal-Wallis test	12
	5.2	Simple linear regression: Estimation and interpretation, Multiple regression analysis: Assumptions and applications	
	5.3	Correlation analysis and interpretation, Model diagnostics and validation	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

"Statistics for Business and Economics" by Paul Newbold, William L. Karla, and Betty Thorne

(2013)
"Introduction to the Practice of Statistics" by David S. Moore, George P. McCabe, and Bruce A. Craig (2017)
"Probability and Statistics for Engineering and the Sciences" by Jay L. Devore (2015)
"Applied Multivariate Statistical Analysis" by Richard A. Johnson and Dean W. Wichern (2018)
"Statistics: The Art and Science of Learning from Data" by Alan Agresti and Christine A. Franklin (2017)

MCA313 Natural language Processing

GENERAL DETAILS OF THE COURSE

Course Code	MCA313		
Course Name	Natural language Processing		
Course Type	Specialization		
Credit	4		
Semester	III		
Course Objectives	The course aims to enable students to understand the need for speech processing, appreciate various speech analysis techniques, and recognize the models developed for speech recognition.		
Pre-requisites			
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	3	1	0

COURSE OUTCOMES (CO)

CO#	CO Description
CO1	Understand the fundamentals of speech processing techniques.
CO2	Choose and apply appropriate speech analysis techniques.
CO3	Design and implement an acoustic model for input speech.
CO4	Illustrate and evaluate models for speech recognition.
CO5	Develop and apply advanced deep learning models for Automatic Speech Recognition.

CO-PSO-PO MAPPING MATRIX

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	P 0 10	PO 11	PO 12	PS O 1	PS O 2	PS O 3
CO1		2			1								3	3	
CO2		2			1								3	3	
CO3		2											3	3	
CO4		2			1								3	3	
CO5		2			1								3	3	

COURSE CONTENT

Module No		Module Content	Hours Required
1	1.1	**Speech Fundamentals and Signal Processing-** Basic Concepts of Speech: Articulatory Phonetics - Production and Classification of Speech Sounds	12
	1.2	Acoustic Phonetics: Acoustics of Speech Production	
	1.3	Review of Digital Signal Processing Concepts	
	1.4	Short-Time Fourier Transform (STFT), Filter-Bank, and LPC Methods	
2	2.1	Speech Analysis and Feature Extraction- Features and Feature Extraction Techniques for Speech	12
	2.2	Speech Distortion Measures: Mathematical and Perceptual (Log Spectral Distance, Cepstral Distances, etc.)	
	2.3	Likelihood Distortions and Spectral Distortion with Warped Frequency Scale, Extraction Methods: LPC, PLP, MFCC Coefficients	
	2.4	Time Alignment and Normalization: Dynamic Time Warping and Multiple Time-Alignment Paths	
3	3.1	Automatic Speech Recognition (ASR) and Acoustic Models- Introduction to Automatic Speech Recognition (ASR): Basic Architecture and Components	12
	3.2	Conventional Acoustic Models: Gaussian Mixture Models (GMM), Random Variables, and Parameter Estimation	
	3.3	Hidden Markov Models (HMM): Markov Chains, HMM Sequences, EM Algorithm, Viterbi Decoding	
	3.4	HMM Variants for Generative Speech Modeling and Recognition	
4	4.1	Deep Neural Networks and Hybrid Systems for ASR	12
	4.2	Introduction to Deep Neural Network (DNN) Architecture and Parameter Estimation (Error Backpropagation)	
	4.3	Practical Considerations: Data Preprocessing, Model Initialization, Weight Decay, Dropout, Batch Size Selection DNN-HMM Hybrid Systems: Key Components of CD-DNN-HMM	
	4.4	Kullback-Leibler Divergence-Based HMM-Hybrid Models for Improved Speech Recognition Performance	
5	5.1	Advanced Deep Models and Representation Learning- Representation Learning in Deep Neural Networks: Joint Learning of Feature Representation and Classifier	12
	5.2	Robustness of Features: Flexibility Across Conditions, Adaptation of DNNs, Multitask and Transfer Learning for Speech Recognition Multilingual and Cross-lingual Speech RecognitionMultiobjective Training and Exploiting Audio-Visual Information for Robust Speech Recognition	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

Daniel Jurafsky, James H. Martin, Peter Norvig and Stuart Russell, "Speech and Language Processing", Pearson, 3rd Edition 2020, ISBN: 978-0131873216, 0131873210.
Yu, Dong, Deng, Li, "Automatic Speech Recognition: A Deep Learning Approach" SpringerVerlag London, 2015, ISBN : 978-1-4471-5778-6, 978-1-4471-5779-3.
Lawrence Rabiner, Biing-Hwang Juang, B. Yegnanarayana, "Fundamentals of Speech Recognition", Pearson Education, 2009, ISBN: 9788177585605, 8177585606.
Himanshu Mohan, MeghaYadav , "Speech Recognition System and its Application" LAP LAMBERT Academic Publishing, 2019, ISBN: 978-6200117847, 6200117845.
Uday Kamath, John Liu, James Whitaker, "Deep Learning for NLP and Speech Recognition", Springer,1st edition, 2019, ISBN: 978-3030145989, 3030145980

MCA314 Image Mining

General Details of the Course

Course Code	MCA314		
Course Name	Image Mining		
Course Type	Specialization		
Credit	4		
Semester	III		
Course Objectives	This course aims to provide students with an understanding of image mining techniques, focusing on extracting meaningful information from image data. Students will learn to apply various computational techniques, including machine learning and pattern recognition, to solve real-world problems in image analysis.		
Pre-requisites	Data mining basics		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	3	1	0

Course Outcomes (CO)

CO#	CO Description
CO1	Understand the fundamental concepts and techniques used in image mining and image analysis.
CO2	Apply feature extraction methods to represent image data for various mining tasks.
CO3	Implement classification and clustering algorithms for image data analysis.
CO4	Use deep learning models such as CNN for advanced image mining tasks.
CO5	Explore the applications and challenges of image mining in real-world domains, including medical imaging and surveillance.

CO-PSO-PO Mapping Matrix

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3
CO1	2			1								3	3		
CO2	2			1								3	3		
CO3	2											3	3		
CO4	2			1								3	3		
CO5	2			1								3	3		

COURSE CONTENT

Module No		Module Content	Hours Required
1	1.1	Introduction to Image Mining- Overview of Image Mining and its Applications	12
	1.2	Types of Image Data: Text, Audio, and Visual Data, Image Representation: Pixels, Color, Texture, Shape	
	1.3	Introduction to Image Preprocessing Techniques, Basic Concepts of Computer Vision and Image Processing	
2	2.1	Feature Extraction and Representation- Understanding Image Features: Color, Texture, Shape, and Edge	12
	2.2	Techniques for Feature Extraction: Histogram, Gabor Filters, Wavelets Dimensionality Reduction for Image Data (PCA, LDA)	
	2.3	Image Descriptors: SIFT, HOG, SURF, Feature Matching and Object Recognition	
3	3.1	Image Classification and Clustering- Overview of Image Classification Methods, Supervised Learning Techniques: k-NN, SVM, Decision Trees, Neural Networks	12
	3.2	Unsupervised Learning Techniques: k-Means, DBSCAN, Hierarchical Clustering	
	3.3	Evaluation Metrics for Classification and Clustering (Accuracy, Precision, Recall)	
	3.4	Applications of Image Classification in Real-World Scenarios	
4	4.1	Pattern Recognition and Mining Techniques- Pattern Recognition in Image Mining	12
	4.2	Introduction to Convolutional Neural Networks (CNNs), Image Mining with Deep Learning Models	
	4.3	Image Mining Algorithms for Object Detection and Tracking Case Studies in Medical Imaging, Security, and Surveillance	
5	5.1	Advanced Image Mining Techniques- Image Retrieval Systems and Content-Based Image Retrieval (CBIR)	12
	5.2	Image Segmentation Techniques, Transfer Learning in Image Mining	
	5.3	Multimodal Image Mining (Integration of Image, Text, and Audio Data), Ethical Considerations and Privacy Concerns in Image Mining	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

"Image Mining: A Survey of Information Processing Techniques" by Shashi Shekhar and Sanjay Chawla (2003)
"Digital Image Processing" by Rafael C. Gonzalez and Richard E. Woods (2018)

"Pattern Recognition and Machine Learning" by Christopher M. Bishop (2006)
"Computer Vision: Algorithms and Applications" by Richard Szeliski (2010)
"Deep Learning for Computer Vision" by RajalingappaaShanmugamani (2018)
"Content-Based Image Retrieval: A Survey" by Mohamed S. Kamel and Wanlei Zhou (2006)

MCA413 Tableau

General Details of the Course

Course Code	MCA413		
Course Name	Tableau		
Course Type	Specialization		
Credit	4		
Semester	IV		
Course Objectives	This course aims to provide students with comprehensive skills in using Tableau for data visualization, analysis, and dashboard creation, enabling effective communication of insights. Students will also learn advanced techniques such as integrating R and Python for enhanced analytics and managing Tableau Server for secure and efficient deployment.		
Pre-requisites	Basic understanding of data analysis, basic statistics, database fundamentals, programming, and data visualization principles		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	3	1	4

Course Outcomes (CO)

CO#	CO Description
CO1	Gain a comprehensive understanding of Tableau's evolution, interface, and connecting to diverse data sources.
CO2	Develop proficiency in handling data types, performing preparation, and creating basic visualizations with sorting, filtering, and grouping.
CO3	Master advanced visualization techniques, geospatial mapping, interactive dashboards, and data blending for storytelling.
CO4	Acquire expertise in data analysis, statistical methods, forecasting, and exploring data through sets, groups, and hierarchies.
CO5	Gain hands-on experience integrating Tableau with R and Python for advanced analytics, predictive modeling, and machine learning.

CO-PSO-PO Mapping Matrix

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3
CO1					3								3		
CO2		1			3								3		
CO3		1			3								3		

CO4					3								3		
CO5					3								3		

COURSE CONTENT

Module No		Module Content	Hours Required
1	1.1	Understanding Data Visualization and its Significance in Data Analysis.	12
	1.2	Exploring the History and Evolution of Tableau	
	1.3	Tableau Products Overview: Tableau Desktop, Tableau Server, Tableau Prep, Navigating Tableau Interface: Workbooks, Worksheets, and Dashboards.	
	1.4	Connecting to Various Data Sources: Databases, Files, APIs.	
2	2.1	Exploring Different Data Types and Structures in Tableau.	12
	2.2	Data Preparation Techniques: Cleaning, Transforming, and Aggregating Data.	
	2.3	Creating Basic Visualizations: Bar Charts, Line Graphs, Pie Charts. Implementing Sorting, Filtering, and Grouping Functions in Tableau	
	2.4	Utilizing Parameters and Calculated Fields for Data Manipulation.	
3	3.1	Constructing Advanced Charts: Gantt Charts, Bullet Graphs, Box Plots.	12
	3.2	Geospatial Mapping and Spatial Analysis using Tableau.	
	3.3	Building Interactive Dashboards for Storytelling and Insight Presentation.	
	3.4	Utilizing Advanced Calculations and LOD Expressions, Implementing Data Blending and Data Relationships	
4	4.1	Exploring Aggregation and Granularity in Data Analysis	12
	4.2	Statistical Analysis: Correlation, Regression, and Clustering in Tableau	
	4.3	Forecasting Methods and Trend Analysis Tools	
	4.4	Implementing Advanced Filtering Techniques: Context Filters, Top N Filters, Handling Sets, Groups, and Hierarchies in Tableau	
5	5.1	Introduction to Integration of Tableau with R and Python Performing Advanced Analytics using R and Python Scripts in Tableau	12
	5.2	Leveraging Predictive Modeling and Machine Learning in Tableau Exploring Data Science Applications within Tableau Environment	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

"Tableau Your Data!: Fast and Easy Visual Analysis with Tableau Software" by Dan Murray
"Learning Tableau" by Joshua N. Milligan
"Communicating Data with Tableau" by Ben Jones
Tableau's official documentation and online training resource
Tableau Community Forums and Knowledge Base
"Data Visualization with Tableau" by Ashutosh Nandeshwar

MCA414 BigData Analytics

GENERAL DETAILS OF THE COURSE

Course Code	MCA414		
Course Name	BigData Analytics		
Course Type	Specialization		
Credit	4		
Semester	IV		
Course Objectives	The course objectives aim to equip students with a foundational understanding of big data analytics, from data management and processing to advanced analytics and real-world applications. The course should also instill ethical considerations in handling big data and provide practical experience through hands-on projects and case studies.		
Pre-requisites	Basics of data and programming concepts,		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	3	1	0

COURSE OUTCOMES (CO)

CO1	To understand the theoretical foundations and relevance of big data technologies
CO2	To understand Hadoop Distributed File System and implement HDFS for storing and retrieving big data.
CO3	To understand the anatomy of Map Reduce algorithm and implement Map Reduce for processing big data streams in a distributed manner.
CO4	To understand advanced and supplementary tools to work with big data, such as Pig and Hive.
CO5	To understand the privacy and ethical concerns while handling Big Data

CO-PSO-PO MAPPING MATRIX

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	P01 0	PO1 1	PO1 2	PSO 1	PSO 2	PSO 3
CO#1	1	2		3	3								3		
CO#2	1	2		3	3								3		
CO#3	1	2		3	3								3		
CO#4	1	2		3	3								3		
CO#5	1	2		3	3								3		

COURSE CONTENT

Course Content			
Module No		Module Content	Hours Required
1	1.1	Classification of Digital Data, Structured and Unstructured Data - Introduction to Big Data:	12
	1.2	Characteristics – Evolution – Definition - Challenges with Big Data - Other Characteristics of Data - Why Big Data -	
	1.3	Traditional Business Intelligence versus Big Data - Data Warehouse and Hadoop Environment	
	1.4	Big Data Analytics: Classification of Analytics – Challenges - Big Data Analytics , Ethical considerations in Big Data	
2	2.1	NoSQL, Comparison of SQL and NoSQL, RDBMS Versus Hadoop - Distributed Computing Challenges	12
	2.2	Hadoop Overview - Hadoop Distributed File System - HDFS Daemons, Anatomy of File Read,	
	2.3	Anatomy of File Write, Processing Data with Hadoop - Managing Resources and Applications with Hadoop YARN	
	2.4	Interacting with Hadoop Ecosystem	
3	3.1	MAP REDUCE Introduction to Map Reduce: The map tasks, Grouping by key, The reduce tasks, Combiners,	12
	3.2	Details of Map Reduce Execution, Coping with node failures. Algorithms Using Map Reduce:	
	3.3	Matrix-Vector Multiplication, Computing Selections and Projections, Union, Intersection, and Difference	
	3.4	Natural Join. Extensions to Map Reduce: Workflow Systems, Recursive extensions to Map Reduce, Common map reduce algorithms.	
4	4.1	Pig : Introduction to PIG, Execution Modes of Pig, Comparison of Pig with Databases, Grunt, Pig Latin, User Defined Functions	12
	4.2	Data Processing operators. Hive : Hive Shell, Hive Services, Hive Metastore,	
	4.3	Comparison with Traditional Databases, HiveQL, Tables, Querying Data and User Defined Functions. Hbase	
	4.4	HBasics, Concepts, Clients, Example, Hbase Versus RDBMS. Big SQL : Introduction	
5	5.1	Privacy – Re identification of Anonymous People	12
	5.2	Why Big Data Privacy is self-regulating? – Ethics – Ownership	
	5.3	Big Data Security – Organizational Security.	

	5.3	Ethical Guidelines	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

Seema Acharya, Subhashini Chellappan, "Big Data and Analytics", Wiley Publications, First Edition,2015
Judith Huruwitz, Alan Nugent, Fern Halper, Marcia Kaufman, "Big data for dummies", John Wiley & Sons, Inc.(2013)
Tom White, "Hadoop The Definitive Guide", O'Reilly Publications, Fourth Edition,2015
Dirk Deroos, Paul C.Zikopoulos, Roman B.Melnky, Bruce Brown, Rafael Coss, "Hadoop For Dummies", Wiley Publications,2014Robert D.Schneider, "Hadoop For Dummies", John Wiley & Sons, Inc.(2012)
Paul Zikopoulos, "Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data, McGraw Hill, 2012 Chuck Lam, "Hadoop In Action", Dreamtech Publications, 2010

MCA415 Genetic Algorithm

General Details of the Course

Course Code	MCA415		
Course Name	Genetic Algorithm		
Course Type	Specialization		
Credit	4		
Semester	IV		
Course Objectives	Understand Nature Inspired Optimization Techniques Apply fundamental concepts of Multi-objective Optimization Techniques Understand possibilities of hybrid Optimization techniques Evaluate the possibility of solving real world problems using ACO Analyse real world problems using PSO		
Pre-requisites	ta Structures and Algorithms, Artificial Intelligence		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	3	1	0

Course Outcomes (CO)

CO#	CO Description
CO1	Analyse nature inspired algorithms for optimization
CO2	Understand multi-objective optimization technique
CO3	Analyse the application of hybrid systems
CO4	Analyse routing problem using ACO
CO5	Apply PSO for feature selection

CO-PSO-PO Mapping Matrix

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3
CO1		1			3								2	1	
CO2		1			3								2	1	
CO3		1			3								2	1	
CO4		1			3								2	1	
CO5		1			3								2	1	

COURSE CONTENT

Module No		Module Content	Hours Required
1	1.1	Natural Evolution – Chromosomes, Natural Selection, Crossover, Mutation	12
	1.2	Genetic Algorithms – chromosomes, Fitness Function, Population, GA Operators, Elitism, GA Parameters, Convergence	
	1.3	Case Study – Practical Application using GA	
2	2.1	Multi objective Genetic Algorithms – Problem Formulation, Pareto-Optimal Front	12
	2.2	Pareto-Optimal Ranking, Muti-objective Fitness, Multi-objective GA process	
	2.3	Case Study – Practical Application using MOGA	
3	3.1	Hybrid Systems – Neuro-genetic systems, Fuzzy Neural systems, Fuzzy genetic systems	12
	3.2	Applications of Hybrid systems	
4	4.1	Ant Colony Optimization – Biological Inspiration, Algorithm Overview	12
	4.2	Problem Representation, Pheromone Updating, Solution Construction,	
	4.3	Convergence and Termination	
	4.4	Case study on Applications in Routing Problems	
5	5.1	Particle Swarm Optimization – Biological Inspiration of PSO	12
	5.2	Algorithm and Problem Representation, Particle Movement, Convergence and Termination	
	5.3	Variants of PSO, Multi-Swarm PSO, Case study on Application of PSO	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

Soft Computing by Samir Roy, Udit Chakraborty
Introduction to Evolutionary Algorithms, Xinjie Yu, Mitsuo Gen, Springer, 2010
Introduction to Genetic Algorithms by Melanie Mitchell
Evolutionary Computation: A Unified Approach by Kenneth A. DeJong, MIT Press, 2006, ISBN: 02620419444
Ant Colony Optimization by Marco Dorigo, Thomas Stützle
Particle Swarm Optimization by Maurice Clerc and James Kennedy

ADVANCED PROGRAMMING

MCA321 Kotlin

GENERAL DETAILS OF THE COURSE

Course Code	MCA321		
Course Name	Kotlin		
Course Type	Specialization		
Credit	4		
Semester	III		
Course Objectives	To acquire knowledge of Kotlin To understand the difference between Java and Kotlin To develop application using Kotlin with Javascript or Android		
Pre-requisites	JAVA, DBMS		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	3	1	0

COURSE OUTCOMES (CO)

CO#	CO Description
CO1	To Understand both the basic concepts of Kotlin
CO2	To Understand the object-oriented concepts of Kotlin
CO3	To Analyze the functional programming in Kotlin
CO4	To integrate Kotlin with javascript and Android
CO5	To implement the advanced concepts in Kotlin

CO-PSO-PO MAPPING MATRIX

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	P01 0	PO1 1	PO1 2	PSO 1	PSO 2	PSO 3
CO#1			1											2	
CO#2			1		2									2	
CO#3			2											1	
CO#4			2		2										
CO#5			2		2			1						3	

COURSE CONTENT

Module No		Module Content	Hours Required
1	1.1	Kotlin – Overview, Features, Environment Setup, Editing,	12

		Compiling, and Running, Execute for Android	
	1.2	Kotlin Basics: Basic Types, Defining Local Variables, Comments, Functions, Packages and Imports, If Expression, When Expression,	
	1.3	For Loops, While Loops, Break and continue Loop Understand functions (basic with return type, three dots operator, new	
	1.4	functions: default arguments and named arguments, extension). Recall exceptions and examine why checked exceptions are not supported.	
2	2.1	OOPs in Kotlin: Object Oriented Programming, Association, Composition, Aggregation Inheritance	12
	2.2	Lists Sets, and Maps, Interfaces, and Implementation,	
	2.3	Design by Contract, Classes and Objects, Constructors,	
	2.4	Static methods, public-private internal protected class, methods Objects, Delegation	
3	3.1	Functional Programming in Kotlin: Functions as objects, Creating and returning functions, Nested functions	12
	3.2	Dynamic functions, Anonymous(lambda) functions,	
	3.3	Unit returning functions, Inline Functions, Extension Functions Couroutines	
	3.4	Dictionary and Set Comprehensions, Collection interconversion patterns	
4	4.1	Kotlin JavaScript Interaction: Calling JavaScript from Kotlin, Tools in Kotlin	12
	4.2	Calling Java from Kotlin, Calling Kotlin from Java	
	4.3	Kotlin for Android: Understand extensions. Examine why view Binding is more recommended.	
	4.4	Develop the application project as a recap.	
5	5.1	Destructuring Declarations, Collections, Ranges,	12
	5.2	Type Checks and Casts, This expression, Equality, Operator overloading	
	5.3	Null Safety, Exceptions, Annotations Reflection,	
	5.3	Type-Safe Builders, Dynamic Type	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

Kotlin in Action - Dmitry Jemerov and Svetlana Isakova (2017): A practical guide to Kotlin for modern application development.
The Joy of Kotlin - Pierre-Yves Saumont (2019): Focuses on functional programming principles in Kotlin.

Mastering Kotlin - Nate Ebel (2019): Advanced techniques and tools for mastering Kotlin development.
Kotlin for Android Developers - Antonio Leiva (2017): A comprehensive guide for Android development using Kotlin.
Kotlin Programming Cookbook - Samuel Urbanowicz (2018): Hands-on recipes for solving common Kotlin programming challenges.

MCA322 .NET

General Details of the Course

Course Code	MCA322		
Course Name	.NET		
Course Type	Specialization		
Credit	4		
Semester	III		
Course Objectives	Understand the fundamentals of ASP.NET Core and the MVC architecture. Develop database-driven web applications using Entity Framework Core. Develop a comprehensive understanding of modern web development practices.		
Pre-requisites	Basic Programming Knowledge, Basic Understanding of Database Concepts		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	3	1	0

Course Outcomes (CO)

CO#	CO Description
CO1	Ability to solve problems using only pure object oriented concepts and frameworks
CO2	Ability to design and develop database applications
CO3	Able to develop networking and distributed applications
CO4	Ability to design GUI applications
CO5	Design and develop Web applications

CO-PSO-PO Mapping Matrix

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO#1			1											2	
CO#2			1		2									2	
CO#3			2											1	
CO#4			2		2										
CO#5			2		2			1						3	

Course Content

Module No		Module Content	Hours Required
1	1.1	Introduction to C# and .NET	12

		Setting up Visual Studio and the .NET development environment, .NET architecture, Data Types, Variables, and Constants. Control Flow and Decision Making- Conditional Statements (if, else, switch),Loops (for, while, do-while),Operators	
	1.2	Exception Handling: try-catch blocks. Methods and Functions – Method overloading, Scope and Visibility. Object-Oriented Programming (OOP) Basics- Classes and Objects, Constructors and Destructors, Properties and Fields.	
	1.3	Inheritance – Types of inheritance, Method Overriding, Polymorphism, Encapsulation, Abstract Classes and Interfaces, Collections and Generics.	
2	2.1	ADO.Net Basic window control, Architecture of ADO.Net	12
	2.2	Comparison with ADO, Connected and Disconnected Database,	
	2.3	Create Connection using ADO.NET Object Model, Connection Class, .Net Data provider, Data Adapter, Data Set, Data Row, Data Column, Data Relation, command, Data Reader, Language integrated query(LINQ).	
3	3.1	I/O and Object Serialization Object serialization and Remoting: System.IO, Streams, TextWriter, TextReader	12
	3.2	Binary Writer, BinaryReader, Serialized Object Persistence and formatters	
	3.3	binary formatter, soap formatter	
4	4.1	Remoting- Distributed Applications, COM/DCOM in Distributed Environment, Drawbacks of DCOM,	12
		NET Remoting – New distributed environment, Advantages & Disadvantages, . Implementing a Simple Remoting Client and Server.	
	4.3	Network programming: Socket programming, TCP/IP, UDP	
5	5.1	Windows Forms Application Introduction to Windows Forms, Basics of Windows Forms and setup.	12
	5.2	Creating a simple GUI application. Common controls -Textbox, Button, CheckBox, RadioButtons, ComboBox, GroupBox etc. and event handling.	
	5.3	Creating menus and toolbars, Using dialog boxes and managing multiple forms, Forms Database programming, Binding controls to data sources	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)			
C# 2012 Programming, Covers .Net 4.5, Black Book			
Professional .NET programming – wrox publication			
Professional ASP.NET 4.5 in C# – Jason N. Gaylord (Author), Christian Wenz (Author), Pranav Rastogi (Author), Todd Miranda (Author)			
Professional C# Web Services: Building .NET Web Services with ASP .NET and· .NET Remoting – Zach Greenvoss and Christian Nagel			

MCA323 UX design

General Details of the Course

Course Code	MCA323		
Course Name	UX design		
Course Type	Specialization		
Credit	4		
Semester	III		
Course Objectives	Develop a foundational understanding of UX design principles and concepts. Learn how to put users at the center of the design process and prioritize their needs and preferences. Gain hands-on experience in creating wireframes and interactive prototypes.		
Pre-requisites	An understanding of web and mobile applications. Creativity and a strong interest in user-centered design.		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	3	1	0

Course Outcomes (CO)

CO#	CO Description
CO1	Apply UX design principles to create user-friendly and effective product interfaces.
CO2	Develop a user-centered design approach and mindset that prioritizes user needs.
CO3	Gain practical skills in designing wireframes and interactive prototypes using industry-standard tools.
CO4	Create organized and user-friendly information architectures for digital products.
CO5	Implement responsive design techniques to ensure products are accessible and functional across different devices.

CO-PSO-PO Mapping Matrix

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3
CO1		3	2		3									2	
CO2		2			3									2	
CO3		3	2		3	1								2	
CO4		2	2		3	1								2	
CO5		2	2		3									2	

COURSE CONTENT

Module No		Module Content	Hours Required
1	1.1	Introduction to UX Design UX design and its importance, UX design principles	12
	1.2	Difference between User Experience Design and User Interface Design	
	1.3	User-centered design and its impact on product success.	
2	2.1	User Research and Personas Methods for user research (interviews, surveys, etc.)	12
	2.2	Creating user personas	
	2.3	Using personas in design	
3	3.1	Wireframing and Prototyping Low-fidelity and high-fidelity wireframes,	12
	3.2	Interactive prototyping using tools like Figma or Sketch,	
	3.3	Rapid prototyping techniques	
4	4.1	Information Architecture and Interaction Design Principles of information architecture, Navigation design.	10
	4.2	Interaction patterns and usability.	
5	5.1	Usability Testing and Feedback Planning and conducting usability tests, Analyzing and incorporating user feedback, Iterative design improvements	14
	5.2	Responsive Design and Portfolio Development- Implement responsive design techniques for web and mobile platforms, Building a professional design portfolio.	
	5.3	Presenting and explaining design decisions in a portfolio	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

"The Design of Everyday Things" by Don Norman (2013)
"Interaction Design: Beyond Human-Computer Interaction" by Jenny Preece, Yvonne Rogers, and Helen Sharp (2015)
"The Elements of User Experience: User-Centered Design for the Web and Beyond" by Jesse James Garrett (2010)
"100 Things Every Designer Needs to Know About People" by Susan Weinschenk (2011)

MCA324 Game Development with Unity

General Details of the Course

Course Code	MCA324		
Course Name	Game development using Unity		
Course Type	Specialization		
Credit	4		
Semester	III		
Course Objectives	Grasp Fundamental Game Development Concepts: Understand the basics of game design, mechanics, and the development process. Master Unity Tools and Environment: Gain proficiency in Unity's interface, tools, and project management. Develop C# Scripting Skills: Learn to write scripts using C# for various game functionalities. Apply Physics and Animation Techniques: Implement realistic game elements using physics and animation principles. Stay Updated with Industry Standards: Understand current trends and industry best practices in game development.		
Pre-requisites			
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	3	1	0

Course Outcomes (CO)

CO#	CO Description
CO1	Students will learn fundamental game development concepts, navigate Unity's interface, and write C# scripts to create basic games.
CO2	Students will apply physics principles, master animation techniques, and optimize game performance within Unity.
CO3	Students will grasp game design principles, create engaging environments, and design interfaces for better player experiences.
CO4	Students will understand multiplayer game development, implement networking functionalities, and manage multiplayer aspects in Unity
CO5	Students will explore advanced game development concepts, study industry trends, and execute a comprehensive game development project showcasing multiple learned skills.

CO-PSO-PO Mapping Matrix

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	P 0 10	PO 11	PO 12	PS O 1	PS O 2	PS O 3
CO1		3	2		3									2	

CO2		2			3								2	
CO3		3	2		3								2	
CO4		2	2		3								2	
CO5		2	2		3								2	

COURSE CONTENT

Module No		Module Content	Hours Required
1	1.1	Introduction to Game Development, History and Evolution of Games, Understanding Game Engines, Basics of Unity Interface and Environment	12
	1.2	Scripting in C# for Unity, Creating and Managing Game Objects	
	1.3	User Input and Interaction in Unity, Implementing Audio in Games	
	1.4	Basics of Game Design Principles, Game Development Project Setup	
2	2.1	Understanding Physics in Unity, Implementing Colliders and Rigidbodies, Unity's Physics Material and Effects	12
	2.2	Working with 2D and 3D Animation in Unity, Animation Controllers and States	
	2.3	Creating Player and Object Movement, Implementing Camera Controls, Handling Particle Systems in Unity,	
	2.4	Scripting Interactions with Physics, Optimizing Game Performance in Unity.	
3	3.1	Game Design Theory and Process, Storytelling and Narrative Design in Games, Level Design and Environment Creation	12
	3.2	User Interface (UI) and User Experience (UX) in Games, Implementing Menus and Heads-Up Displays (HUD), UI/UXTesting and Iteration	
	3.3	Game Balancing and Playtesting, Integrating Visual Effects in Games	
	3.4	Designing for Different Platforms and Resolutions, Sound Design in Games.	
4	4.1	Introduction to Multiplayer Game Development, Networking Fundamentals for Games	12
	4.2	Implementing Multiplayer using Unity's Network System	
	4.3	Syncing Player Actions and Game State, Peer-to-Peer and Client-Server Networking	
	4.4	Creating Game Lobbies and Matchmaking, Handling Latency and Lag in Multiplayer Games, Security and Anti-Cheat Measures	
5	5.1	Procedural Content Generation, Artificial Intelligence in Games, Virtual Reality (VR) and Augmented Reality (AR) in Unity,	12
	5.2	Integrating Machine Learning in Games, Mobile Game Development with Unity	

	5.3	Monetization and Publishing Games ,Continuous Integration and Deployment (CI/CD) for Games	
	5.3	Advanced Shader Programming in Unity, Case Studies and Industry Trends, Final Game Development Project.	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

"Unity in Action" by Joseph Hocking
"Game Design Workshop: A Playcentric Approach to Creating Innovative Games" by Tracy Fullerton
"Learning C# by Developing Games with Unity 2020" by Harrison Ferrone
"Networked Graphics: Building Networked Games and Virtual Environments" by Anthony Steed, Manuel Fradinho Oliveira, and NunoGuimarães
"Game Programming Patterns" by Robert Nystrom

MCA423 Software testing with Selenium

General Details of the Course

Course Code	MCA423		
Course Name	Software Testing using Selenium		
Course Type	Specialization		
Credit	4		
Semester	IV		
Course Objectives	Acquire proficiency in software testing using Selenium, from basic concepts to advanced techniques, enabling participants to create robust automated test scripts and integrate them into the software development process.		
Pre-requisites	Basic knowledge of the software development life cycle (SDLC). Proficiency in at least one programming language Familiarity with web development and HTML/CSS. Experience with basic testing concepts.		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	3	1	0

Course Outcomes (CO)

CO#	CO Description
CO1	Assess and compare manual and automated testing methodologies
CO2	Configure Selenium WebDriver and execute basic commands, showcasing proficiency.
CO3	Employ advanced Selenium techniques for dynamic web elements, demonstrating proficiency.
CO4	Design and implement data-driven tests using testing frameworks.
CO5	Integrate Selenium with Jenkins and Selenium Grid.

CO-PSO-PO Mapping Matrix

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3
CO1	2	3	2					2	3					1	
CO2	2	2			2			2						1	
CO3		2			2			2						1	
CO4						1			2		3			1	
CO5					3	1			3		2			1	

COURSE CONTENT

Module No		Module Content	Hours Required
1	1.1	Introduction to Software Testing: Basics of software testing	12
	1.2	Types of testing: manual vs. automated	
	1.3	Importance of testing in the software development life cycle (SDLC)	
2	2.1	Introduction to Selenium: Overview of Selenium and its components.	12
	2.2	Setting up Selenium WebDriver	
	2.3	Basic Selenium commands and operations	
3	3.1	Advanced Selenium Techniques: Locating web elements using different strategies	12
	3.2	Handling dynamic web elements	
	3.3	Synchronization strategies in Selenium	
4	4.1	Testing Frameworks and Best Practices: Introduction to testing frameworks (TestNG, JUnit).	12
	4.2	Data-driven testing and parameterization	
	4.3	Best practices in Selenium automation	
5	5.1	Advanced Topics and Integration: Introduction to Selenium Grid	12
	5.2	Integration of Selenium with other tools (e.g., Jenkins)	
	5.3	Best practices for maintaining automated test scripts	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

Selenium WebDriver: From Foundations to Framework by Yujun Liang and Alex Collins
Mastering Selenium WebDriver by Mark Collin.
Selenium Design Patterns and Best Practices by Dima Kovalenko
Test Automation using Selenium WebDriver with Java by Yujun Liang
Web Links
SeleniumHQ: https://www.selenium.dev/
TestNG Documentation: https://testng.org/doc/
Selenium Grid Documentation: https://www.selenium.dev/documentation/en/grid/
Jenkins Documentation: https://www.jenkins.io/doc/
Applitools - Visual Testing with Selenium: https://applitools.com/tutorials/selenium.html

MCA424 Struts, Hibernate and Spring

General Details of the Course

Course Code	MCA424		
Course Name	Struts, Hibernate and Spring		
Course Type	Specialization		
Credit	4		
Semester	IV		
Course Objectives	To acquire knowledge of MVC architecture To design and develop enterprise applications using Frameworks To familiar with Hibernate and Transactions in SQL To build web applications using Spring Framework		
Pre-requisites			
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	3	1	0

Course Outcomes (CO)

CO#	CO Description
CO1	To understand the Model-View-Controller (MVC) design pattern and how it is best applied to Java Web application development with respect to a scenario.
CO2	To design mapping between entities and attributes using modern tools
CO3	To create different types of persistent classes and Map java inheritance hierarchy with database tables using various mapping techniques
CO4	To analyse data effectively from database using traditional SQL and Hibernate Query Language
CO5	To design and develop Enterprise applications using Spring and Springboot

CO-PSO-PO Mapping Matrix

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	P0 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3
CO1	1	2	2												
CO2					3										
CO3			2												
CO4	1				2										
CO5											3	2	2		3

COURSE CONTENT

Module No		Module Content	Hours Required
1	1.1	Introduction to Struts MVC Architecture - Framework Application Flow, Components Model, View and Controller	12
	1.2	Building a simple web application using Struts 1 Struts Validator - Introduction to validator plugin, using different types of validators, Configuring the application,	
	1.3	Applying validators, Building custom validators, Declarative exception handling	
	1.4	Struts2 Action - Action Interface, ActionSupport class.	
2	2.1	Basics of Hibernate - Hibernate Introduction, Hibernate Architecture, Understanding First Hibernate application	12
	2.2	Hibernate Application - Hibernate with annotation, Hibernate Web application, Hibernate Generator classes, Hibernate Dialects	
	2.3	Inheritance Mapping - Table per Hierarchy, Table Per Concrete, Table Per Subclass	
	2.4	Table per Hierarchy using Annotation, Table Per Concrete using Annotation, Table Per Subclass using Annotation	
3	3.1	Collection Mapping - Mapping List, One-to-many by List using XML, Many to Many by List using XML,	12
	3.2	One To Many by List using Annotation, Mapping Bag, One-to-many by Bag, Mapping Set, One-to-many by Set, Mapping Map, Many-to-many by Map	
	3.3	Bidirectional Lazy Collection Component Mapping	
	3.4	Association Mapping - One-to-one using Primary Key, One-to-one using Foreign Key	
	3.5	Transaction Management ,HQL – Hibernate Query Language, HCQL, Named Query	
	3.6	Hibernate Caching - First Level Cache, Second Level Cache	
	3.7	Integration - Hibernate and Struts, Hibernate and spring	
4	4.1	Basics of Spring - What is Spring, Spring Modules, Spring Application, IOC container	12
	4.2	Dependency Injection - Constructor Injection, CI Dependent Object, CI with collection, CI with Map, CI Inheriting Bean, Setter Injection, SI Dependent Object, SI with Collection, SI with Map, CI vs SI, Auto wiring, Factory Method	
	4.3	Spring with ORM- Spring with Hibernate, Spring with JPA	

	4.4	SpEL- SpEL, Operators in SpEL, variable in SpEL ,Web Integration- Spring with Struts2	
5	5.1	Introduction to Spring Boot: Overview and Features Setting up Spring Boot with Spring Initializr, Basic Spring Boot Application Structure	12
	5.2	Introduction to Dependency Injection (DI) ,Constructor Injection and Setter Injection, Autowiring with @Autowired, Basic Bean Configuration using @Component.	
	5.3	Building a Simple RESTful API , Introduction to Spring Boot Security ,Returning JSON from Controllers ,Introduction to Spring Boot and Databases (H2, MySQL),	
	5.4	Running Spring Boot applications ,Integrating Spring Boot with Hibernate	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

Beginning Apache Struts – Arnold Doray
Struts: The Complete Reference Book
Mastering Jakarta Struts
Struts in Action – Ted Husted, Cedric Dumoulin, George Franciscus, David Winterfeld
Just Spring Integration – Madhusudhan Konda
Spring Data – Mark Pollack, Oliver Gierke
Beginning Apache Struts – Arnold Doray
Struts: The Complete Reference Book
Mastering Jakarta Struts

MCA425 Project management Tools

General Details of the Course

Course Code	MCA425		
Course Name	Project management Tools		
Course Type	Specialization		
Credit	4		
Semester	IV		
Course Objectives	The course aims to provide a solid understanding of project management principles and equip students with the practical skills to use project management tools effectively. It focuses on planning, scheduling, collaboration, tracking, and advanced features to manage projects successfully.		
Pre-requisites			
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	3	1	0

Course Outcomes (CO)

CO#	CO Description
CO1	To understand the fundamental concepts and phases of project management and their application in real-world projects.
CO2	To analyze and effectively use project management tools for planning, scheduling, and tracking project progress.
CO3	To develop collaboration and communication skills using integrated project management platforms.
CO4	To create and manage reports, dashboards, and KPIs for efficient project monitoring.
CO5	To integrate advanced features and automation in project management tools for optimizing workflows and resource management.

CO-PSO-PO Mapping Matrix

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	P01 0	PO1 1	PO1 2	PSO 1	PSO 2	PSO 3
CO#1	2	3	2					2	3					1	
CO#2	2	2			2			2						1	
CO#3		2			2			2						1	
CO#4						1			2		3			1	
CO#5						1			3		2			1	

COURSE CONTENT

Module No		Module Content	Hours Required
1	1.1	Introduction to Project Management Tools: - Fundamentals of Project Management: Definition, objectives, and importance.	12
	1.2	Key phases: Initiation, Planning, Execution, Monitoring, and Closure. Overview of Project Management Tools: Categories (Scheduling, Collaboration, Resource Management, Reporting)	
	1.3	Popular tools: Trello, Asana, Jira, Microsoft Project. Practical session on setting up and exploring basic features of a project management tool.	
2	2.1	Planning and Scheduling Tools: Project Planning Essentials: Work Breakdown Structure (WBS), Gantt Charts, Critical Path Method (CPM), and PERT.	12
	2.2	Scheduling tasks with tools like Microsoft Project and Trello.	
	2.3	Creating timelines and dependencies. Resource allocation and workload distribution.	
3	3.1	Collaboration and Communication Tools: Importance of collaboration in projects and the role of tools in improving communication.	12
	3.2	Features of collaboration tools: Shared calendars, file sharing, and activity tracking- Overview of tools: Slack, Microsoft Teams, and Basecamp.-	
	3.3	Practical session on setting up communication channels and integrating collaboration tools with project management platforms.	
4	4.1	Tracking, Reporting, and Risk Management: - Monitoring project progress through dashboards and reports.	12
	4.2	Key Performance Indicators (KPIs) and tracking progress. Risk management tools: Identifying and mitigating risks.	
	4.3	Tools for issue tracking: Jira and Bugzilla- Practical session on generating reports and using issue-tracking tools.	
5	5.1	Advanced Features and Integration- Advanced project management features: Automation and managing multiple projects.	12
	5.2	Integrating project management tools like Slack, Jira, and Trello for seamless project execution- Practical session on configuring integrations and using APIs for custom workflows.	

	5.3	Case study on end-to-end project setup using multiple tools and analyzing outcomes.	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

Project Management for Dummies by Stanley E. Portny

Microsoft Project 2019 Step by Step by Carl S. Chatfield and Timothy D. Johnson

Agile Project Management with Scrum by Ken Schwaber

Trello for Business by J. L. Williams

COMPUTER SECURITY

MCA331 Ethical Hacking

GENERAL DETAILS OF THE COURSE

Course Code	MCA331		
Course Name	Ethical Hacking		
Course Type	Specialization		
Credit	4		
Semester	III		
Course Objectives	This course provides students with the mathematical foundations necessary for Information Security and related fields. It helps students to know about how Security problems can be formulated with mathematical models and have the necessary core skills of mathematically analyzing Information Security in real world problems.		
Pre-requisites	Computer network, information security		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	3	1	0

COURSE OUTCOMES (CO)

CO#	CO Description
CO1	Understand the basics of information Gathering
CO2	Understand the concept of Sniffing
CO3	Analyse Password Cracking
CO4	Demonstrate IP spoofing
CO5	Demonstrate Email Phishing

CO-PSO-PO MAPPING MATRIX

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO#1	1			1											2
CO#2	1			2											2
CO#3			1	2		1									3
CO#4			1	2											3
CO#5			1	2											3

COURSE CONTENT

Module No		Module Content	Hours Required
1	1.1	Information Gathering- Introduction to Ethical Hacking, What is hacking?,Definition of Hackin, Types of Hackers Introduction to Information Security, CIA Triad, Services & Techniques, Actives and Passive Threats and Exploit, etc.	12
	1.2	Introduction to Information Gathering, Phases of Information Gathering, Reconnaissance, Banner Grabbing, Web Ripping, Website at Offline Mode, Downloading Server Side Code, Foot Printing,Name Space Lookup,Trace Routing Techniques, Whois Lookup Query, Fingerprinting	
	1.3	Hands-on lab on Information Gathering and its countermeasures	
2	2.1	SNIFFING, ARP CACHE POISONING & MITM ATTACKS. Sniffing, ARP Cache Poisoning, Man in the Middle (MITM) Attacks,	10
	2.2	Hands on Lab on Trojan Virus & its Countermeasures	
3	3.1	PASSWORD CRACKING Password Hashes, Password Cracking types, Dictionary Attack, Brute Force Attacks, Cracking Passwords using John the Ripper, Other password Cracking tools	14
	3.2	How passwords are stored in Linux,/etc/passwd and /etc/shadow,How passwords are stored in Windows	
	3.3	Testing SSH Password and Hardening of SSH,Password Cracking Countermeasures	
4	4.1	IP SPOOFING & DENIAL OF SERVICE IP Spoofing, Denial of Service (DoS), TCP SYN Flood Attack using hping3	12
	4.2	Detecting TCP Syn Flood attacks using Wireshark, Detecting TCP Syn Flood attacks using netstat,Suggesting & Implementing Countermeasures	
	4.3	Hands on lab on IP Spoofing , Denial of Service (DoS) and its countermeasures	
5	5.1	E-MAIL SPOOFING and PHISHING Concept of Email,SMTP,POP3 and IMAP, Email Spoofing, Types of Phishing, Email Phishing,	12
	5.2	E-Mail Tracking by Header, Concept of Fake E-mails, Protections, SPF, DKIM and DMARC Records, Using nslookup to check SPF/DKIM/DMARC records Concept of Fake E-mails	
	5.3	Hands on lab on demonstration on phishing mail and its countermeasures.	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

Introduction to Computer Networks and Cybersecurity -- C-H. Wu and J. D. Irwin
Stuart McClure, Joel Scambray and Goerge Kurtz, "Hacking Exposed Network Security Secrets & Solutions", 5th Edition, Tata Mc Graw Hill Publishers, 2010.
Rafay Baloch, "A Beginners Guide to Ethical Hacking
Allen Harper, Shon Harris, Jonathan Ness, Chris Eagle, "Gray Hat Hacking The Ethical Hackers Handbook", 3rd Edition, McGraw-Hill Osborne Media paperback(January 27, 2011)
Stuart McClure, Joel Scambray and Goerge Kurtz, "Hacking Exposed Network Security Secrets & Solutions", 5th Edition, Tata Mc Graw Hill Publishers, 2010.
Stuart McClure, Joel Scambray and Goerge Kurtz, Hacking Exposed 7: Network Security Secrets & Solutions, Tata Mc Graw Hill Publishers, 2010.
Bensmith, and Brian Komer, Microsoft Windows Security Resource Kit, Prentice Hall of India, 2010.

MCA332 Database Security

General Details of the Course

Course Code	MCA332		
Course Name	Database Security		
Course Type	Specialization		
Credit	4		
Semester	III		
Course Objectives	The objective is to equip learners with the expertise needed to protect sensitive data, mitigate security risks, and ensure privacy in database systems.		
Pre-requisites	Database management Systems		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	3	1	0

Course Outcomes (CO)

CO #	CO Description
CO1	Understand the fundamentals of security, and how it relates to information systems and analyse risks and vulnerabilities in operating systems from a database perspective
CO2	Acquire the knowledge and skills for administration of user, profiles, password policies, privileges and roles.
CO3	Understand and apply the various database security models.
CO4	Audit database for security and reliability.
CO5	Learn and implement privacy preserving data mining algorithms.

CO-PSO-PO Mapping Matrix

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P010	PO11	PO12	PSO1	PSO2	PSO3
CO#1	3				2			2				1			2
CO#2	1				2			3				2			2
CO#3					2			3				2			2
CO#4	2				2							3			2
CO#5					2			3				3			2

COURSE CONTENT

Module No		Module Content	Hours Required
1	1.1	Security Architecture: Introduction-Information Systems-Database Management Systems	12
	1.2	Information Security Architecture- Database Security–Asset Types and Value-Security Methods.	
	1.3	Operating System Security Fundamentals: Introduction-Operating System Overview- Security Environment – Components	
	1.4	Authentication Methods-User Administration-Password Policies, Vulnerabilities-E-mail Securit	
2	2.1	Administration of Users: Introduction-Authentication-Creating Users, SQL Server User-Removing, Modifying Users-Default, Remote Users	12
	2.2	Database Links-Linked Servers-Remote Servers-Practices for Administrators and Managers-Best Practices	
	2.3	Profiles, Password Policies, Privileges and Roles: Introduction-Defining and Using Profiles	
	2.4	Designing and Implementing Password Policies-Granting and Revoking User Privileges-Creating, Assigning and Revoking User Roles-Best Practices	
3	3.1	Database Application Security Models: Introduction-Types of Users-Security Models- Application, Types- Data Encryption	12
	3.2	SQL Injection, Types, prevention.	
	3.3	Virtual Private Databases: Introduction-Overview of VPD-Implementation of VPD using Views	
	3.4	Application Context in Oracle-Implementing Oracle VPD-Viewing VPD Policies and Application contexts using Data Dictionary, Policy Manager Implementing Row and Column level Security with SQL Server	
4	4.1	Database Auditing Models, Auditing Database Activities	12
	4.2	Using Oracle Database Activities-Creating DLL Triggers with Oracle, Auditing Database Activities with Oracle	
	4.3	Auditing Server Activity with SQL Server-Security and Auditing Project	
	4.4	Case Study	
5	5.1	Privacy Preserving Data Mining Techniques: Introduction- Privacy Preserving Data Mining Algorithms	12
	5.2	General Survey-Randomization Methods-Group Based Anonymization	
	5.3	Distributed Privacy Preserving Data Mining--Curse of	

		Dimensionality	
	5.3	Application of Privacy Preserving Data Mining	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

Hassan A. Afyouni, "Database Security and Auditing", Third Edition, Cengage Learning,
Charu C. Aggarwal, Philip S Yu, "Privacy Preserving Data Mining": Models and Algorithms, Kluwer Academic Publishers
Ron Ben Natan, "Implementing Database Security and Auditing", Elsevier Digital Press
Database Security: What Students and Professionals Need to Know by Rebecca Gurley Bace and Peter T. Davis
Konrad, E., & Skinner, S. (2019). Oracle Security: Protecting Oracle Database 12c. Apress
Chapple, M., & Dalton, A. (2020). Securing SQL Server: Protecting Your Database from Attackers. Apress.

MCA333 Network Security

General Details of the Course

Course Code	MCA333		
Course Name	Network Security		
Course Type	Specialization		
Credit	4		
Semester	III		
Course Objectives	This course of Network Security aims to teach students how to protect computer networks from unauthorized access, data breaches, and cyber threats through the understanding of security concepts, technologies, and practices.		
Pre-requisites	Computer Networks and Data Communication		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	3	1	0

Course Outcomes (CO)

CO#	CO Description
CO1	To understand the fundamentals of Network Security
CO2	To apply security protocols in the Application layer
CO3	To comprehend network layer security concepts and security protocols
CO4	To understand wireless network security, encompassing wireless security and mobile device security
CO5	To apply mobile and IoT security

CO-PSO-PO Mapping Matrix

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO#1	3							2				1			2
CO#2	1							3				2			2
CO#3								3				2			2
CO#4	2											3			2
CO#5								3				3			2

Course Content

Module No		Module Content	Hours Required
1	1.1	Introduction: Introduction to Security in Networks	12

	1.2	Characteristics of Networks – Intrusion	
	1.3	Kinds of security breaches – Plan of attack - Points of vulnerability	
	1.4	Methods of defense – Control measures – Effectiveness of controls	
2	2.1	Application Layer Security:PGP and S/MIME	12
	2.2	Email	
	2.3	PGP –S/MIME – SSL	
	2.4	Architecture –Handshake ,Change Cipher Space, Alert And Record Protocols – SSL Message	
3	3.1	Network Layer Security:Modes – Two Security Protocols – Security Association – Security	12
	3.2	Policy – Internet Key Exchange – System Security: Description – Buffer Overflow And	
	3.3	Malicious Software – Malicious Programs – Intrusion Detection System – Firewall –	
	3.4	Types of Firewall – Firewall Configuration- Virtual Private Networks	
4	4.1	Wireless Network Security: Wireless Security	12
	4.2	Mobile Device Security -Wireless LAN Overview	
	4.3	Wireless LAN Security - Wireless Application Protocol Overview	
	4.4	Wireless Transport Layer Security - WAP End-To-End Security	
5	5.1	Security In Mobile And Iot: Security - Threats To SDN	12
	5.2	NFV Security Attack Surfaces – ETSI Perspective – Cloud Security – Security Issues – Risks	
	5.3	Data Protection – Security As A Service – Addressing Cloud Security -IOT Security	
	5.4	Vulnerability Patching – Requirements By ITU-T – Security Framework	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

Behrouz A Forouzan,Cryptography and Network Security , McGraw-Hill Education
William Stallings, Network Security Essentials: Applications and Standards, Prentice Hall India
Foundations of Modern Networking: SDN, NFV, QoE, IoT, and Cloud" William Stallings Publisher: Addison-Wesley
William Stallings, Cryptography and Network Security: Principles and Standards, Prentice Hall India

MCA334 Digital Identity management

GENERAL DETAILS OF THE COURSE

Course Code	MCA334		
Course Name	Digital Identity management		
Course Type	Specialization		
Credit	4		
Semester	III		
Course Objectives	To provide a comprehensive understanding of digital identity management, including its principles, technologies, security practices, applications, and ethical implications, to prepare learners for managing and innovating identity systems in a digital world.		
Pre-requisites	Basic knowledge of computer systems, networking, and cybersecurity concepts		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	3	1	0

COURSE OUTCOMES (CO)

CO#	CO Description
CO1	Understand the basic concepts and importance of digital identity.
CO2	Explain the components and systems used in digital identity management.
CO3	Identify and mitigate threats to digital identity while ensuring privacy and compliance.
CO4	Explore emerging technologies and their applications in digital identity.
CO5	Recognize ethical, social, and future trends in digital identity management.

CO-PSO-PO MAPPING MATRIX

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO#1		2			3	3				2					2
CO#2					3	3									2
CO#3					3	3									2
CO#4					3	3									2
CO#5					3	3				3					2

COURSE CONTENT

Module No		Module Content	Hours Required
1	1.1	Foundations of Digital Identity - Definition and Concepts of Digital	12

		Identity, Types of Digital Identities: Personal, Organizational, and Social	
	1.2	Evolution of Digital Identity Systems, Importance of Digital Identity in the Digital World	
	1.3	Ethical Considerations in Identity Management, Inclusion vs. Exclusion, Accessibility and Usability of Digital Identity	
2	2.1	Core Components and Systems of Digital Identity - Identity Attributes: Name, Biometric Data, Behavioral Data, Authentication Methods: Passwords, Biometrics, Two-Factor Authentication (2FA)	12
	2.2	Identity Verification Processes, Identity Lifecycle Management: Creation, Maintenance, and Deactivation	
	2.3	Identity and Access Management (IAM) Systems: Overview of IAM Systems, Identity Providers (IdPs) and Single Sign-On (SSO), Federation and Decentralized Identity (Blockchain-Based Identity Systems)	
		Access Control Models: Role-Based Access Control (RBAC), Attribute-Based Access Control (ABAC)	
3	3.1	Privacy, Security, and Legal Frameworks- Threats to Digital Identity: Identity Theft, Phishing, Data Breaches	12
	3.2	Security Practices: Encryption, Secure Communication Channels, Privacy Concerns: GDPR, CCPA, and Data Protection Regulations	
	3.3	User Consent and Data Minimization, Legal Frameworks Governing Digital Identity: Compliance and Regulatory Considerations	
4	4.1	Emerging Technologies and Applications- Emerging Technologies in Digital Identity: Blockchain and Self-Sovereign Identity (SSI), Biometric Authentication Systems, Artificial Intelligence in Identity Verification	12
	4.2	Applications of Digital Identity- E-Governance (e.g., Aadhaar in India), Banking and Finance, Healthcare Systems	
5	5.1	Ethical, Social, and Future Trends in Digital Identity- Ethical and Social Implications: Balancing Security and Privacy, Digital Identity Accessibility and Inclusion	12
	5.2	Future Trends in Digital Identity- Identity in the Metaverse and Web3, Global Initiatives for Universal Digital Identity	
	5.3	Challenges in Digital Identity Management- Scalability, Interoperability, and Ethical Concerns	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

"Digital Identity" by Phillip J. Windley (2005)

"The Age of Em: Work, Love, and Life when Robots Rule the Earth" by Robin Hanson (2016)
"Digital Identity Management: Concepts, Technologies, and Practice" by Sushil Jajodia, Pericles Loucopoulos, and Fabio Martinelli (2014)
"Identity and Access Management: Business Performance Through Connected Intelligence" by Ertem Osmanoglu (2011)
"Principles of Information Security" by Michael E. Whitman and Herbert J. Mattord (2018)

MCA433 Cyber Security Services

General Details of the Course

Course Code	MCA433		
Course Name	Cyber Security Services		
Course Type	Specialization		
Credit	4		
Semester	IV		
Course Objectives	The course objectives for cyber-security services encompass understanding fundamentals, risk assessment, strategy development, policy implementation and awareness of emerging trends to prepare students for comprehensive cybersecurity management.		
Pre-requisites	Computer Networks, Operating Systems, Computer Security		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	3	1	0

Course Outcomes (CO)

CO #	CO Description
CO1	To understand cybersecurity fundamentals and its services
CO2	To analyse the risk assessment and management
CO3	To comprehend cybersecurity services and strategy development
CO4	To implement and monitor security measures
CO5	To create awareness of emerging trends and technologies

CO-PSO-PO Mapping Matrix

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	P01 0	PO1 1	PO1 2	PSO 1	PSO 2	PSO 3
CO#1	2	3	2												2
CO#2	2	2	3												2
CO#3	3														1
CO#4	1					3				1					3
CO#5	1	2			3										2

COURSE CONTENT

Module No		Module Content	Hours Required
1	1.1	Understanding Cybersecurity Services :- Definition and significance of cybersecurity services.	12
	1.2	Types of Cyber Threats and Attack Vectors,	
	1.3	Principles of Information Security, Legal and Ethical Aspects of Cybersecurity, Cybersecurity Terminology and Concepts.	
	1.4	Role of CSSPs - Overview of the responsibilities of Cybersecurity Service Providers (CSSPs).	
2	2.1	Risk Assessment and Management - Identifying Information Assets and Data Classification	12
	2.2	Conducting Risk Assessments and Vulnerability Scanning,	
	2.3	Threat Modelling and Risk Prioritization, Risk Mitigation Strategies and Controls,	
	2.4	Risk Management Frameworks and Compliance, Case studies on effective risk management in cybersecurity.	
3	3.1	Cybersecurity services and Strategy Development - Creating a Cybersecurity Strategy	12
	3.2	Security Policy and Procedure Development	
	3.3	Security Standards and Frameworks (e.g., NIST, ISO 27001)	
	3.4	Incident Response and Disaster Recovery Planning, Business Continuity Planning	
4	4.1	Implementation and Monitoring - Security Awareness and Training Programs	12
	4.2	Security Controls and Technologies (Firewalls, IDS/IPS, Antivirus, etc.)	
	4.3	Security Operation Centers (SOCs) and Security Information and Event Management (SIEM) Systems	
	4.4	Continuous Monitoring and Security Auditing, Security Metrics and Reporting	
5	5.1	Emerging Trends and Future Directions in Cybersecurity services -	12
	5.2	Cryptography and Encryption Technologies	
	5.3	Cloud Security and Virtualization, Internet of Things (IoT) Security	
	5.4	AI and Machine Learning in Cybersecurity	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

Text book: - Cyber Security Foundation by LEE M. ZEICHNER http://www.cybersecurityfoundations.com/
Cyber-Security Essentials for State and Local Government:- https://www.naco.org/sites/default/files/documents/cyber_security_essentials.pdf
Strategic Cyber Security Management By Peter Trim, Yang-Im Lee

MCA434 Blockchain

General Details of the Course

Course Code	MCA434		
Course Name	Blockchain		
Course Type	Specialization		
Credit	4		
Semester	IV		
Course Objectives	The course aims to equip students with the knowledge and skills needed to work with blockchain technology, understand its fundamental design and architectural primitives, consensus protocols, types of blockchain systems, and security aspects, along with various use cases from different application domains.		
Pre-requisites	Cryptography and Network Security, Computer Networks Database Management System, Distributed Systems		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	3	1	0

Course Outcomes (CO)

CO #	CO Description
CO1	To articulate the fundamentals of blockchain and able to explain cryptographic concepts underlying blockchain technology.
CO2	To analyze the various protocols and mining techniques in Block chain
CO3	To create a wallet transactions, crypto tokens, analyse the block details and Blockchain network.
CO4	To apply Bitcoin techniques to develop solutions in the appropriate domains.
CO5	To analyse the future trends using hyperledgers and emerging developments in Blockchain

CO-PSO-PO Mapping Matrix

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	P010	PO11	PO12	PSO 1	PSO 2	PSO 3
CO#1	3						1				2	2			2
CO#2			3				1				2	2			2
CO#3	3		2				1				2	2			2
CO#4			3				1				2	2			2
CO#5			2				1				2	2			2

COURSE CONTENT

Module No		Module Content	Hours Required
1	1.1	Why Blockchain Technology - Blockchain Architecture, Conceptualization-Blockchain components	12
	1.2	Cryptocurrencies - Characteristics of cryptocurrencies	
	1.3	Crypto wallets - Creation of Blocks - Wallet Transactions, Transaction details in a Block - Merkle Tree - Hash functions	
	1.4	Puzzle friendly and collision resistant hash, public key cryptosystem, Generation of keys, Digital signatures, Zero-knowledge systems.	
2	2.1	Blockchain Types- Public Blockchain, Private Blockchain, Federated Blockchain, Permissionless, Permissioned Blockchain Networks, Ethereum blockchain, Go Ethereum	12
	2.2	Gas, Gas price, Gas Limit, ETH, MetaMask, Public Test Networks, set up a Ethereum node using Geth,	
	2.3	Mining in Blockchain, Steps in Minnig, Double spending	
	2.4	Consensus protocols, PoW, Hashcash, Attacks on Bitcoin, Sybil Attacks, 51% Attack, eclipse attacks, DDoS Attacks, Replay Attacks, Byzantine fault, node failure.	
3	3.1	BitCoin - Introduction – Transactions – Structure	12
	3.2	Transactions types – The structure of a block– The genesis block – The bitcoin network	
	3.3	Wallets and its types– Bitcoin payments– Bitcoin investment and buying and selling bitcoins	
	3.4	Bitcoin installation – Bitcoin programming and the command-line interface – Bitcoin improvement proposals (BIPs).	
4	4.1	Ethereum - Ethereum block chain- Elements of the Ethereum block chain	12
	4.2	Precompiled contracts – Accounts and its types – Block header- Ether – Messages	
	4.3	Mining - Clients and wallets – Trading and investment – The yellow paper	
	4.4	The Ethereum network - Applications developed on Ethereum - Scalability and security issues	
5	5.1	Smart Contract and Hyper ledger – History of Smart Contract, Remix, Compilation of smart contracts, Deployment environments	12
	5.2	Solidity mapping, Function overloading, Personal Blockchain	

		network	
	5.3	Ganache, Contract deployment to Ganache network	
	5.3	Hyper ledger projects – Hyperledger as a protocol – Fabric	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller and Steven Goldfeder, Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction, Princeton University Press (July 19, 2016).
Antonopoulos, Mastering Bitcoin: Unlocking Digital Cryptocurrencies Satoshi Nakamoto, Bitcoin: A Peer-to-Peer Electronic Cash System
Bashir, Mastering Blockchain: Distributed ledger technology, decentralization, and smart contracts explained, 2nd Edition, 2nd Revised edition edition. Birmingham: Packt Publishing, 2018.
Text Books BlockChain for dummies, Manav Gupta, Second IBM Limited Edition, 2018, John Wiley & Sons.
DR. Gavin Wood, "ETHEREUM: A Secure Decentralized Transaction Ledger," Yellow paper.2014.
e-Resources and other Digital Material https://nptel.ac.in/courses/106/104/106104220/ https://nptel.ac.in/courses/106/105/106105184/

MCA435 Firewall Management

General Details of the Course

Course Code	MCA334		
Course Name	Firewall Management		
Course Type	Specialization		
Credit	4		
Semester	IV		
Course Objectives	Equip undergraduates with comprehensive knowledge of firewall management, covering principles, technologies, configurations, advanced features, testing, and incident response, ensuring readiness for real-world network security challenges.		
Pre-requisites	Strong foundation in network fundamentals (TCP/IP, subnetting) and familiarity with core cybersecurity concepts (encryption, authentication) are required		
Course Details	Lecture Hours	Tutorial Hours	Practical Hours
	3	1	0

Course Outcomes (CO)

CO#	CO Description
CO1	Classify and evaluate firewall deployment scenarios
CO2	Analyse and critique firewall technologies
CO3	Design and implement firewall rules and policies
CO4	Integrate and assess advanced firewall features
CO5	Conduct penetration testing and develop incident response plans

CO-PSO-PO Mapping Matrix

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO#1		2	2		3										3
CO#2		3	2		3										3
CO#3		2	3		3										3
CO#4		2	1		3										3
CO#5		2	1		3										3

COURSE CONTENT

Module No		Module Content	Hours Required
1	1.1	Introduction to Firewalls Understanding network security fundamentals	12
	1.2	Types of firewalls: hardware vs. software	
	1.3	Firewall deployment scenarios and considerations	
2	2.1	Firewall Technologies and Protocols: In-depth study of firewall technologies	12
	2.2	Examination of common network protocols	
	2.3	Application-layer filtering and stateful inspection	
3	3.1	Firewall Configuration and Management: Configuration of firewall rules and policies	12
	3.2	Monitoring and logging in firewall systems	
	3.3	Implementing security best practices in firewall management	
4	4.1	Advanced Firewall Features: Intrusion Detection and Prevention Systems (IDPS)	10
	4.2	Virtual Private Networks (VPNs) and firewalls	
	4.3	High Availability (HA) and failover configurations	
5	5.1	Firewall Testing and Incident Response: Performing firewall penetration testing	14
	5.2	Developing an incident response plan for firewall breaches	
	5.3	Continuous improvement and optimization of firewall configurations	

REFERENCES (TEXT BOOKS/ RESEARCHES/ ONLINE URLS)

Firewalls and Internet Security: Repelling the Wily Hacker by William R. Cheswick, Steven M. Bellovin, and Aviel D. Rubin
Network Security Essentials: Applications and Standards by William Stallings
The Web Application Hacker's Handbook: Finding and Exploiting Security Flaws by Dafydd Stuttard and Marcus Pinto
Cisco ASA: All-in-One Next-Generation Firewall, IPS, and VPN Services by Jazib Frahim, Omar Santos, and Andrew Ossipov
Firewall Fundamentals by Wes Noonan and Ido Dubrawsky
Web Links
Cisco Firewall Solutions: https://www.cisco.com/c/en/us/products/security/firewalls/index.html
Palo Alto Networks - Firewall Management: https://www.paloaltonetworks.com/network-security/firewall

Firewall Basics – Norton: https://us.norton.com/internetsecurity-iot-firewall-what-is-a-firewall.html
Open Source Firewalls – pfSense: https://www.pfsense.org/
Firewall Testing Tools: https://owasp.org/www-community/attacks/Web_Application_Firewall_Testing