National Education Policy-2020, w.e.f. July-2024
Academic Program Handbook: Curriculum and Syllabus

Bachelor of Computer Applications (B.C.A.)
Master of Computer Applications (M.C.A.)
Doctor of Philosophy (Ph.D.)

Indira Gandhi National Tribal University (IGNTU), Amarkantak, M.P. (A Central University)

Department of Computer Science
Faculty of Computer Science
Amarkantak, Madhya Pradesh-484887, Bharat (India)

(To be implemented from Academic Year: 2024-24 onwards under NEP 2020)

Table of Content

Sr. No.	Particular	Page Number
A	**Name of the Academic Programme: Bachelor of Computer Applications (BCA)** **Preamble** 1. Program Educational Objectives 2. Pedagogical Teaching and Learning Methods 3. Program Outcomes	
B	**BCA Curriculum** 4. Introduction 5. Structure of the Curriculum BCA Course Prerequisites and Period Details of Curriculum 6. Student Evaluation Model Scheme of Instructions and Examination	
C	**B.C.A. Semester Wise Curriculum Framework Under NEP- 2020**	
D	**Name of the Academic Programme: Master of Computer Applications (MCA)** **Preamble** 7. Program Educational Objectives 8. Pedagogical Teaching and Learning Methods 9. Program Outcomes	
E	**MCA Curriculum** 10. About the Course 11. Structure of the Curriculum 12. MCA Course Prerequisites and Period Medium of Instruction and Examination 13. Dissertation/Research Work/Project/ Internship/Seminar Evaluation: 14. Dissertation/Research Work/Project 15. Committee for Evaluation 16. Seminar 17. Credit and Teaching Hours 18. Strategies: Keeping IT & Computer Science Syllabus (BCA-MCA) Current/ Relevant	
F	**M.C.A. Semester Wise Curriculum Framework Under NEP- 2020**	
G	**PhD. Curriculum & Syllabus Under NEP- 2020**	

Bachelor of Computer Applications (B.C.A.)- Curriculum and Syllabus for Four Year Degree Programme

(To be implemented from Academic Year: 2024-25 onwards under NEP 2020)

A. Name of the Academic Programme: Bachelor of Computer Applications

Preamble:

This syllabus is the extension of the existing syllabus which is currently being taught to B. C. A. of IGNTU for the last few years, but modified as per need of IT and R and D Industry and to be implemented in academia from the academic year 2020 onwards. However, there are few changes incorporated in the existing syllabus. It is believed that the proposed changes qualitative change in the way B.C.A. is taught, which will offer a more enriched learning experience. It aims to provide technology-oriented students with the knowledge and ability to develop creative solutions, and better understand the effects of future developments of computer systems and technology on people and society. The syllabus is about developing skills to learn new technology, grasping the concepts and issues behind its use and the use of computers.

1. Program Educational Objectives:

- To train the graduates to acquire in depth knowledge of fundamental concepts and programming skills for holistic development of the pupils.
- To prepare the graduates for productive careers in the software industry, corporate sector, Government Organizations, and R & D.
- To prepare graduates to acquire excellent computing ability so that they can analyze, design and create Solutions for real time problems.
- To apply the current tools and techniques to create systems for solving Industry oriented problems.
- To prepare graduates to gain multidisciplinary knowledge through real time case studies, projects and industry internships to meet the industry needs.

2. Pedagogical Teaching and Learning Methods:

We would prioritize the integration of modernized and scientifically designed teaching pedagogy. It's essential that any innovative teaching approach be complemented by an equally effective learning process, ensuring that both evolve together. Below are the key techniques we would implement:

- **Crossover Teaching and Learning:** This approach involves relating classroom concepts to real-life examples. For instance, while explaining a theoretical concept, we would draw parallels to practical scenarios that students can easily relate to, thereby enhancing their understanding.

- **Context-Based Teaching:** When discussing theoretical concepts, we use fictitious examples to simplify and clarify ideas. We often draw inspiration from examples cited by renowned professors from institutions like Stanford and MIT to ensure the examples are both relevant and educational.
- **Adaptive Teaching:** We tailor our teaching methods to the varying learning capacities of students. By categorizing students based on their grasp of the material, we develop specialized strategies to address the needs of each group, ensuring that all students have the opportunity to succeed.
- **Incidental Learning:** This involves teaching concepts in unexpected contexts. For example, if we find ourselves in a traffic jam, we might take that opportunity to explain the concept of deadlocks, making learning spontaneous and contextually relevant.
- **Computational Thinking:** To equip students with the skills to tackle complex problems, we emphasize computational thinking. This approach encourages students to break down larger problems into smaller, manageable parts, which can then be solved systematically.
- **Comparative Teaching:** Given the diverse student body, we support the use of comparative teaching, where students' strengths and weaknesses are highlighted through peer comparison. This method not only promotes self-awareness but also motivates students to improve by learning from their peers.
- **Academic Exchange Hub:** We believe in fostering academic growth through exchange programs. By leveraging our contacts at prestigious institutions such as IIT`s, NIT`s, IIIT`s, Thapar University and BITS Pilani, we aim to create a network for the exchange of ideas. This will enhance both faculty and student quality over time through continuous collaboration.

Scheme of Examination:

Mid Semester Examination (MSE)	: 10+10= 20 Marks
Internal Assessment (IA)	: 15+5 (Attendance)=20 Marks
End Semester Examination (ESE)	: 60 Marks

3. Program Outcomes:

- Apply knowledge of computing fundamentals, computing specialization, Programming Skills, and domain knowledge to conceptualize computing models. Problem analysis Identify, formulate, research literature, and solve complex computing problems reaching substantiated conclusions using fundamental principles of computing sciences, and relevant domain disciplines.
- The 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.
- Innovation and entrepreneurship identify a timely opportunity and use innovation to pursue that opportunity to create value and wealth for the betterment of the individual and society at large.

B. BCA Curriculum:

4. Introduction:

In the fastest growing era of Information Technology, almost every year new technology, techniques, programming languages are coming on the market. To be at par with vibrant changes, this undergraduate course BCA including 8 th semester degree program curriculum has been designed. This is a very challenging task to design curriculum with absorbing rapid changes in technologies, and accordingly identify the core and elective subjects. In designing this curriculum, we have accommodated core courses and elective courses covering vast domains of Information technologies under new education policy (NEP 2020). This curriculum also has a strong laboratory and project orientation in which the use of new tools is used. Most courses will have an associated laboratory and it is expected that they will be equipped with the latest software tools.

5 . Structure of the Curriculum:

Minimum Credit Required to award degree in each category as per UGC

SN	Broad Category of Courses	Minimum credit requirement		Remark
		3-Year UG	4-Year UG	
1	Major (Core)	60	80	1-8th Semesters
2	Minor Streams (two) One from the discipline (but different from Major), second from Vocational e.g. B.VoC.	12+12	16+16	1-8th Semesters
3	Multidisciplinary (Not studied at 12th level)	09	09	1-3rd Semesters
4	Ability Enhancement Course (AEC) (Common for all UG)	08	08	1-4th Semesters
5	Skill Enhancement Course (SEC)	09	09	1-3rd Semesters
6	Value added Course (VAC) (Common for all UG)	08	08	1-2nd Semesters
7	Summer Internship	02	02	5th Semester
8	Research Project (for Hons with Res)	-	12	8th Semester
	Total Credit	**120**	**160**	

BCA Course Prerequisites and Period

Students who have passed 10+2 examination with Mathematics from a recognized Board/Institution having scored a minimum of 45% aggregate in case of candidates belonging to General, EWS and OBC Category and 40% in case of students belonging to SC/ST categories are eligible to apply.

Details of Curriculum

- Major (core) subject indicates the department where the student has been admitted. No major paper should contain more than 4 credits in any semester.

- There is provision for two minor papers in all 8 semesters of a 4 year-UG Programme. The details of the minor papers are as follows:

- The first minor paper (Minor-1) carrying 2 credits will be offered by each department for students from other departments.

- Second Minor (Minor-2) carrying 2 credits shall be vocational in nature. There will be a pool of courses offered by the interested department at university level.

- The Dean of the Faculty of Vocational Education will be the Nodal Officer for Minor-2 and shall be assigned the task of teaching and evaluation as per the university norms. 4.

- Multi-disciplinary (First Three Semesters) will be offered at the faculty level. The Dean of the Faculty, HoD and faculty members shall design the multidisciplinary course. These courses shall be kept in a basket/pool of subjects. The concerned deans shall be the nodal officer for these multidisciplinary courses and coordinate for the formation of the syllabus, allotment of the paper to the concerned departments/teaching staff and formulation of the scheme for evaluation.

- Skill Enhancement Course (SEC) of 3 Credits shall be offered at the departmental level for the first three semesters only. The nature of the SEC papers will be skill-based.

- Four Papers of value-added courses (VAC) of (2+2 credits) will be introduced for the first two semester students at the university level. All the faculty members of the concerned departments would share the responsibility for teaching/evaluating VAC. Even if any faculty member is not directly engaged in the teaching they may also share the responsibility of evaluation. The Nodal officer shall distribute the teaching and evaluation assignments among faculty members concerned.

- A mandatory internship is required to complete any course under NEP.

- Those students who drop the undergraduate degree course after completing the first years must complete an internship to become eligible for a UG Certificate. In this case internship shall be arranged in the concerned semester.

- Those students who drop the undergraduate degree course after completion of the second year must complete an internship to become eligible for a UG Diploma. In this case internship shall be arranged in the concerned semester.

- Those students who are continuing the Three Year/ Four Year Undergraduate Degree have to do a mandatory summer internship in the 5th semester only.

- It is advised that the internship should be done either during summer break or winter break. Additional credits can be earned through the NCC/NSS programmes as per the UGC norms. However the additional credits earned for the NSS/NCC will not be added to the calculation of CGPA/SGPA.

- All those students opting for NCC/NSS must complete mandatory credits and pass all the papers. There will be no relaxation in the mandatory credits/papers for NCC/NSS students.

6. Student Evaluation:

The details are given below.

- **Model Scheme of Instructions and Examination**

The medium of instruction as well as examination will be English only.

End Semester Examination: 60 Marks

The END SEMESTER Paper shall be of 60 marks of 3 hours duration. The pattern of Questions asked shall be as per the format provided by CoE.

Note: A student shall be eligible to appear in the End Semester Examination of course if he/she appeared in Mid Semester Examination and Internal Assessment and fulfills the requirement attendance, failing which he/she will not be permitted to appear in the End Semester Examination of the respective course.

Dissertation/Research Work/Project/ Internship/Seminar Evaluation:

The dissertation/Research Work/Project work is evaluated based on the following heads as per the requirements of evaluation criteria mentioned in curriculum framework:

1. Mid 1- Presentation ; evaluation of Synopsis
2. Mid 2- Presentation ; evaluation of Progress of work
3. End Semester:
    a. Evaluation of Dissertation
    b. Presentation
    c. Viva

Dissertation/Research work/Project: Dissertation/research work/project copied from other students will be considered to have used unfair means. If two dissertations are found identical by more than 40% then zero marks will be awarded to both of them. In such a case the dissertation will have to be resubmitted on the new topic.

Committee for Evaluation:

The evaluation of components I and II is carried out by course coordinator assigned by

the Head of the Department.
Components III will be evaluated by a committee consisting of the Chairman of BoS or his/her nominee and an external examiner/internal examiner invited from other University/ Industry/ Society/ Community/Other Department or the same Department within the University.

Internship: Students are responsible to choose Internship program online/offline to learn any specific field of knowledge and provide certificate from the agency/society/institute/center for the fulfillment of the course/part of work. Another option is also provided for those students how have not chosen such kind of program is responsible to complete the assigned task provided by Department under any faculty.
This work will be provided by concerned faculty member and student has to complete the work and submit report to Department and evaluation will be based on such documents and reports.

Seminar: The purpose of a seminar is to enable students to improve their knowledge and understanding of a topic by engaging with key issues. Evaluation of seminar will be based on presentation and viva voce.

Credit and Teaching Hours:
The credit and teaching hours shall be distributed as under:

Theory
1- Credit = 15 hours / per semester
2- Credit = 30 hours / per semester
3- Credit = 45 hours / per semester
4- Credit = 60 hours / per semester

Practical
1- Credit = 15 hours / per semester
2- Credit = 30 hours / per semester

Tutorial
1- Credit = 15 hours / per semester
For clarity, the abbreviations used in this syllabus are as follows:

- L: Lecture
- T: Tutorial
- P: Practical
- C: Credit

*****

C. (B.C.A.)-Semester Wise Curriculum Framework Under NEP- 2020
Bachelor of Computer Applications

FIRST SEMESTER							
Name of Course	Name of course and code	Course Title	MM	L	T	P	C
Discipline Specific: Major 1	CSE MT-101	Programming in C	100	3	1	0	4
Minor	CSE MI-102	Computer Fundamentals	100	4	0	0	4
Multi Disciplinary Major	MDC-FCS-100	Organic Path to Sustainable Entrepreneurship	100	2	1	0	3
Ability Enhancement Course (AEC)	AEC-GH	General Hindi	50	2	0	0	2
Skill Enhancement Course (SEC)	SEC-104	Web Design and Development	100	2	1	0	3
Value Added	VAC-IKS VAC-HYW	1. Indian Knowledge System 2. Yoga, Health & Wellness	100	2+2	0	0	4
		Total Credits					20

SECOND SEMESTER							
Name of Course	Course Code	Course Title	MM	L	T	P	C
Discipline Specific: Major 1	CSE MT-201	Object Oriented Programming Using C ++	100	3	1	0	4
Minor	CSE MI-202	Semiconductor and Digital Electronics	100	4	0	0	4
Multi Disciplinary Major	MDC-FCS-200	Panch Parivartan for Vishwa Guru Bharat	100	2	1	0	3

Ability Enhancement Course (AEC)	AEC-GE	General English	50	2	0	0	2
Skill Enhancement Course (SEC)	SEC-204	Programming with Python	100	2	1	0	3
Value Added	VAC-EEDM VAC-TAC VAC-PA	1.Environmental Education and Disaster Management 2.Tribal Art and Culture 3.Performing Arts	100	2+2	0	0	4
Total Credits			20				

THIRD SEMESTER							
Name of Course	Name of course and code	Course Title	MM	L	T	P	C
Discipline Specific: Major 1	CSE MT-301	Fundamentals of Data Structures	100	3	1	0	4
Discipline Specific: Major 2	CSE MT-302	Operating System	100	3	1	0	4
Minor	CSE MI-303	Logic Circuit Design	100	4	0	0	4
Multi Disciplinary Major	MDC-FCS-300	Innovative Entrepreneurial Solutions	100	2	1	0	3
Ability Enhancement Course (AEC)	AEC-CS	Communication Skill	50	2	0	0	2
Skill Enhancement Course (SEC)	SEC-305	Computer Graphics	100	2	0	1	3
Total Credits			20				

FOURTH SEMESTER							
Name of Course	Name of course and code	Course Title	MM	L	T	P	C
Discipline Specific: Major 1	CSE MT - 401	Core Java Programming	100	3	1	0	4
Discipline Specific: Major 2	CSE MT - 402	Database Management System	100	3	1	0	4
Discipline Specific: Major 3	CSE MT - 403	Design and Analysis of Algorithms	100	3	1	0	4
Discipline Specific: Major 4	CSE MP- 404	Core Java Programming Laboratory	50	0	0	2	2
Minor 1	CSE MI- 405	Discrete Mathematics	50	2	0	0	2
(Minor 2) Vocational	CSE MI- 406	DBMS Laboratory	50	0	0	2	2
Ability Enhancement Course (AEC)	AEC-ICT	Information and Communication technology	50	2	0	0	2
		Total Credits					20

FIFTH SEMESTER							
Name of Course	Name of course and code	Course Title	MM	L	T	P	C
Discipline Specific: Major 1	CSE MT- 501	.NET Programming	100	3	1	0	4
Discipline Specific: Major 2	CSE MT- 502	Software Engineering	100	3	1	0	4
Discipline Specific: Major 3	CSE MT- 503	Computer Networks	100	3	1	0	4
Discipline Specific: Major 4	CSE MP- 504	.NET Programming Laboratory	50	0	0	2	2
Minor 1	CSE MI- 505	Computer System and Architecture	50	2	0	0	2

(Minor 2) Vocational	CSE MI-506	Software Engineering Laboratory	50	0	0	2	2
Internship	CSE-I-507	Internship	50	0	0	2	2
		Total Credits					20

SIXTH SEMESTER							
Name of Course	Name of course and code	Course Title	MM	L	T	P	C
Discipline Specific: Major 1	CSE MT-601	Fundamentals of Cyber Security	100	3	1	0	4
Discipline Specific: Major 2	CSE MT-602	Data Warehouse and Data Mining	100	3	1	0	4
Discipline Specific: Major 3	CSE MT-603	Artificial Intelligence	100	3	1	0	4
Discipline Specific: Major 4	CSE MP-604	Project Work	100	0	0	4	4
Minor 1	CSE MI-605	Basics to Quantum Computing	50	2	0	0	2
(Minor 2) Vocational	CSE MI-606	Artificial Intelligence Laboratory	50	0	0	2	2
		Total Credits					20

SEVENTH SEMESTER							
Name of Course	Name of course and code	Course Title	MM	L	T	P	C
Discipline Specific: Major 1	CSE MT-701	Advanced Programming	100	3	1	0	4
Discipline Specific: Major 2	CSE MT-702	Machine Learning and its application	100	3	1	0	4

Discipline Specific: Major 3	CSE MT-703	Data Structures and Algorithms	100	3	1	0	4
Discipline Specific: Major 4	CSE MP-704	Machine Learning Using Python Laboratory	50	0	0	2	2
Discipline Specific: Major 5	CSE MP-705	Data Structures and Algorithms Laboratory	50	0	0	2	2
Minor 1	CSE MI-706	Fundamentals of Big Data Analytics	50	2	0	0	2
(Minor 2) Vocational	CSE MI-707	Introduction to Data Science	50	2	0	0	2
		Total Credits					20

EIGHT SEMESTER (HONS.)							
Name of Course	Name of course and code	Course Title	MM	L	T	P	C
Discipline Specific: Major 1	CSE MT-801	Fundamentals of IoT and Cloud Computing	100	4	0	0	4
Discipline Specific: Major 2	CSE MT--802	Advanced Java Programming	100	3	1	0	4
Discipline Specific: Major 3	CSE MT-803	Advanced Database Management System	100	3	1	0	4
Discipline Specific: Major 4	CSE MT-804	Theory of Computation	100	3	1	0	4
Minor 1	CSE MP-805	Advanced Java Programming Laboratory	50	0	0	2	2
(Minor 2) Vocational	CSE MI-806	Advanced DBMS Laboratory	50	2	0	0	2
		Total Credits					20

EIGHT SEMESTER (HONS. RESEARCH)							
Name of Course	Name of course and	Course Title	MM	L	T	P	C

	code						
Research Specific: Major 1	CSER MT-801	Research Design and Methodology	100	3	1	0	4
Research Specific: Major 2	CSER MP-802	Dissertation	100	3	1	0	4
Research Specific: Major 3	CSER MT-803	Research Publication	100	0	0	3	4
Research Specific: Major 4	CSER MT-804	Viva -Voce	100	2	1	0	4
Minor 1	CSER MI-805	Advanced Research Laboratory	50	0	0	2	2
(Minor 2) Vocational	CSER MI-806	Seminar Presentation	50	0	0	2	2
		Total Credits					20

*****

BCA- First Year, I-Semester

Name of Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Discipline Specific: Major 1: CSE MT-101	Programming in C	03	01	00	20	20	60	100

Course Description: The course on C programming aims to equip students with a fundamental understanding of programming concepts and logic through the C language. Students will learn to design, write, and debug C programs, fostering skills in algorithmic thinking and problem-solving. The course emphasizes the mastery of C syntax, control structures, functions, and memory management.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Understand comprehensive introduction to C programming, covering algorithms, flowcharts, source and object programs, compilers, interpreters.

- Understand how to break a large problem into smaller parts, writing each part as a module or function.
- Apply code reusability with functions and pointers
- The students will be able to develop applications.

UNIT-I: Fundamentals of Programming and Development: 10 Hours

Introduction to Programming Concepts, History of languages, Algorithms, Flowchart, Source Program, Object Program, Compilers, Interpreters, Assemblers, Modular Programming: Structured Programming, Top- down approach, Stages of Program Development.

UNIT-II: Introduction to C Programming: 15 Hours

Introduction to C: character set, Identifiers and keywords. Data type, DeclarationsExpressions, statements and symbolic constants, Input-Output: getchar, putchar, scanf, printf, gets, puts, functions, Pre-processor commands, #include, #define, preparing and running a complete C program. Operators and expressions: Arithmetic, Unary, Logical, bit-wise, assignments and conditional Operator, Library functions.

UNIT-III: Control Flow and Data Structures in C: 10 Hours

Control statements: if, if-else, nested if statement, if else if ladder, Switch case, While, do-while, break, continue and goto statements, comma operator. Arrays: Defining and processing. Multi-dimensional arrays. String processing: in built string handling functions (strlen, strcpy, strcat and strcmp, puts, gets)

UNIT-IV: Functions and Storage Classes: 10 Hours

Functions: Defining and accessing: Passing arguments, Function prototypes, Use of library functions, User define function, Pre-define function, pass by value, pass by reference, Recursion function. Storage classes: Storage Classes: Automatic, External, Static, Register, Scope and life time of variables.

UNIT-V: Pointers, Structures, and File Handling: 15 Hours

Pointers: Declarations, Passing to a function. Operations on pointers, pointer and arrays. Array of pointers. Structure: Defining and processing. Passing to a function, Union. Data Files: Open, close, create, process unformatted data files.

Text/Reference Books:

1. Dr. Manmohan Singh, Rahul Sharma, Dr. Neeraj Kumar Rathore and Dr. Urmila S Soni, C Programming, Arcler Press-Canada, 2024, ISBN: 978-1-77469-960-7.
2. Let Us C: Authentic guide to C programming language - 19th Edition, Yashavant Kanetkar, 2022
3. Programming In Ansi C, 8th edition, Balagurusamy, 2019
4. Programming in C, by Pradip Dey & Manas Ghosh, Oxford

5. Kernighan, B.W. and Ritchie, D.M., The C Programming Language, 2nd edition Pearson, 2015
6. Computing Fundamentals And C Programming, 2nd Edition, 2017

*****

Name of course and code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Minor CSE MI-102	Computer Fundamentals	04	00	00	20	20	60	100

Course Description: This course establishes a foundational understanding of computers, encompassing introductory concepts, number systems, and the essential components of a computer system. It serves as a cornerstone for subsequent computer application courses, providing students with the fundamental knowledge and terminology necessary to succeed in more specialized areas of study.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Identify and describe the basic components of a computer system.
- Convert between different number systems and explain data representation methods.
- Differentiate between various types of software and operating systems.
- Compare and contrast different types of input/output devices, memory, and storage devices.
- Explain the advantages and disadvantages of different network topologies.

UNIT-I: Introduction to Computers: 10 Hours

Introduction, Definition of a computer, Characteristics of computers, Generations of computers Classification of computers, Computer Organization Basic components of a computer system - Input Unit, Output Unit, CPU - ALU and CU, Memory. Concepts of Hardware and Software, Data Representation, Number Systems - Decimal, Binary, Octal, Hexadecimal and their conversions. Data representation - ASCII, Unicode, concepts of a bit, byte, KB, MB, GB, TB. Software Concepts, Types of software - System Software, Application Software. Introduction to Operating Systems - Functions of OS, Types of OS - Single user, Multiuser, Multitasking, Time Sharing.

UNIT-II: Computer Hardware and Networks Devices: 10 Hours

Input and Output Devices, Keyboard, Mouse, Scanner, Printers, Plotters, Monitors, Memory Hierarchy, Primary Memory - RAM, ROM, Cache Memory, Secondary Storage Devices - Magnetic Disks, Optical Disks, Flash Memory. Introduction to Computer Networks, Need for Networking, Types of Networks - LAN, WAN, MAN. Network Topologies - Bus, Star, Ring, Mesh. Basic Network Devices - Hub, Switch, Router, Modem.

UNIT-III: The Internet and Emerging Trends: **10 Hours**

Introduction to the Internet, Definition, History and Evolution of the Internet, Internet Services - WWW, Email, FTP, Social Networking, Working with the Web, Web Browsers, Search Engines, URLs, Websites.

UNIT-IV: Concepts of E-commerce: **10 Hours**

Concepts of E-commerce, E-banking, E-learning, Basic Security Measures - Passwords, Firewalls, Antivirus. Introduction to Emerging Trends in Computing Cloud Computing, Mobile Computing, Big Data, Artificial Intelligence, Internet of Things.

UNIT V: Computer Software & Languages **10 Hours**

System Software: System software Vs. Application Software, Types of System Software, Introduction and Types of Operating Systems. Boot Loader, Diagnostic Programs, BIOS, Utility Programs. Application Software: Microcomputer Software, Interacting with the System, Trends in PC software, Types of Application Software, Difference between Program and Packages. Computer Languages: Definition, Generations of computer languages, Types of Languages, Language Processors: Assembler, Interpreter, Compiler, Linker and Loader. Programming constructs, Algorithm & flowchart.

Text/Reference Books:

1. Sinha, P. K. Computer fundamentals. BPB Publications.
2. Brookshear, J. G. Computer science: An overview. Addison-Wesley.
3. Forouzan, B. A. Data communications and networking. McGraw-Hill Education.
4. Stallings, W. Computer organization and architecture: Designing for performance. Pearson Education.
5. Tanenbaum, A. S. Computer networks. Pearson Education.

*****

Name of course and code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Multi Disciplinary Major MDC-FCS-100	Organic Path to Sustainable Entrepreneurship	02	01	00	20	20	60	100

Course Description: The Organic Path to Sustainable Entrepreneurship course is designed to empower aspiring students with the knowledge and skills needed to build and sustain successful enterprises rooted in ethical, local, and sustainable practices. By exploring the unique "DLEC"

concept—Decentralisation, Local, Entrepreneurship, and Cooperation—students will learn how to navigate the entrepreneurial landscape in India, harnessing local resources, traditional crafts, and innovative ideas to create resilient and impactful businesses. The course emphasizes the importance of ethical decisionmaking, financial transparency, and community engagement, guiding students through the entire entrepreneurial journey from idea generation to scaling and sustaining growth. Through a blend of theoretical insights, practical strategies, and inspiring case studies, this course prepares students to become leaders who not only thrive in business but also contribute to social and environmental wellbeing, leaving a lasting legacy for future generations.

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

- Develop a deep understanding of the organic approach to entrepreneurship, focusing on sustainable, ethical, and locally-driven business practices.
- Generate viable business ideas and secure seed money through personal savings, family support, and community resources, while adhering to ethical principles.
- Master core entrepreneurial skills and relationship building, essential for sustaining and growing a successful business.
- Apply the DLEC framework to create businesses that are decentralized, locally focused, innovative, and cooperative, aligning with national development goals.
- Scale their enterprises strategically, balancing innovation and risk with core values and customer trust, while maintaining business resilience in a fluctuating market.
- Incorporate social responsibility and sustainable practices into their business models, contributing to the economic empowerment of communities and promoting environmental stewardship.

UNIT-I: Foundations of Organic Path to Sustainable Entrepreneurship: 12 Hours

Understanding Organic approach for Entrepreneurship: Definition and importance, The Indian context, Idea Generation and Seed Money: The process of generating viable business ideas, Ethical considerations in obtaining seed money, Sources of seed money: Personal savings, family, and community support. Setting Up Your Enterprises: Minimizing costs: Working from home or shared spaces, The significance of starting small and scaling gradually. Importance of ethical practices in business, The longterm benefits of maintaining financial transparency. Case studies: Successful entrepreneurs who had followed organic path.

UNIT- II: Core Entrepreneurial Skills: 13 Hours

Marketing Strategies for Entrepreneurs: Understanding customer needs and market demands, Improving product quality based on customer feedback. Financial Management: Managing finances in a growing business, The role of customers as the best source of financing, Financial discipline: Budgeting, cash flow management, and reinvestment. Building Relationships: The importance of networking and creating healthy business relationships, Trustworthiness and reliability in business. Overcoming Challenges: Identifying and transforming challenges into opportunities, Positive mindset and proactive problem-solving. Case studies: Marketing success stories, Entrepreneurs who excelled through strong relationships.

UNIT- III: Scaling and Sustaining Business Growth **15 Hours**

Scaling Your Enterprise: When and how to scale up your business operations, Strategies for expansion: Thinking big, new, and out of the box, Balancing risk with opportunity. Sustaining Growth: Maintaining quality and customer trust during expansion, Innovating while staying true to your core values. Science Technology and Innovations: STI interventions, Adopting new technologies, Integrating technology into traditional business models. Building Resilience: Developing resilience against market and economic fluctuations, The role of adaptability in sustaining longterm growth, Strategies for maintaining business stability. Case studies: Entrepreneurs who successfully scaled their businesses. Entrepreneurship and Social Responsibility: Promoting and practicing local production and consumption, Aligning business goals with national development objectives. Sustainable and Inclusive Business Models: Case studies: Entrepreneurs who contributed to social and environmental causes. Leadership and Legacy: Building a legacy through ethical business practices. Case studies: Indian entrepreneurs who have shaped and inspired future generations.

UNIT- IV: DLEC Concept for Entrepreneurship Ecosystem **15 Hours**

Overview of the DLEC Concept (Decentralisation (D), Local (L), Entrepreneurship (E), and Cooperation (C)), DLEC Relevance to the Indian Entrepreneurship Ecosystem, Integration of DLEC in National and Regional Development Strategies, Aligning of DLEC with Government Policies (e.g., Atmanirbhar Bharat, Make in India), Role of Social Organizations, Decentralisation as a Pillar of Economic Empowerment, Implementation Strategies for Financial and Administrative Decentralisation, Vocal for Local (L) Focus in Entrepreneurial Development, Promoting Local Economies, Preserving and Promoting Traditional Entrepreneurship and Crafts, Local Branding on Global Markets, Strengthening Local Supply Chains, Examples and case study of Successful Local, Entrepreneurship (E) and Role of Startups in Economic Growth, Government Initiatives and Support for Entrepreneurs (e.g., Startup India, Mudra Yojana), Innovation and Technology Adoption, Cooperation (C) for Sustainable Entrepreneurship, Importance of Cooperative Models in Entrepreneurship, History and Evolution of Cooperatives in India, Cooperative Societies and Their Role in Economic Empowerment, Building Collaborative Networks, Community Based Organizations (CBOs), Examples and case study of Successful Cooperative Ventures in India

Text /Reference Books:

1. Prof. Raj Kumar Mittal, UGC-Member and Former Vice-Chancellor, Guru Govind Singh Indraprasth University, New Delhi, “rsEntreprenup - 37 Crore Startup ka Desh”, SBF New Delhi, Prabodh and Company
2. R. K. Mishra & K. Trivikram, Skill Development Programmes, Excel Books
3. Delwyn Clark, Elena Mamouni Limnios, Sophie Reboud, Research Handbook on Sustainable Co-operative Enterprise: Case Studies of Organisational Resilience in the Co-operative Business Model, Edward Elgar Publishing Ltd

4. Tauheed S Burke, Succeed Anyway!: The Entrepreneur Development Manual, Page Publishing Inc.

*****

Name of course and code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Ability Enhancement Course: AEC-GH	General Hindi	02	00	00	10	10	30	50

The syllabus will take from University Basket.

*****

Name of Course and code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Skill Enhancement Course: SEC-104	Web Design and Development	02	01	00	20	20	60	100

Course Description:: The purpose of this course is to enable the students to learn the basics of PHP, MySQL, and AJAX for creating dynamics websites. The Pre-requisites are the familiarity with web designing using HTML, CSS and JAVA Script.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Write PHP scripts to handle HTML forms.
- Write regular expressions including modifiers, operators, and meta characters.
- Write PHP programs that use various PHP library functions, and that manipulate files and directories.
- Understood the concepts of using MySQL and AJAX with PHP
- Create dynamic Website/ Web based Applications, using PHP, MySQL database

UNIT-I: Introduction of HTML, DHTML, and Java Script: 10 Hours

HTML, DHTML, HTML Tags and their properties, Introduction of CSS, Types of CSS : INLINE, EMBEDED, LINKED/EXTERNAL AND IMPORTED , CSS Properties and values, Using CSS and HTML TAGS for creating web page layout, Introduction of JavaScript, JavaScript Tokens-Identifiers, Keywords, Literals, Operators, Symbols, JavaScript Functions – Built-in functions & User-defined functions, Using Objects and Events , Using JavaScript in Website

UNIT-II: Introduction to PHP: **15 Hours**

Basic Knowledge of websites, Introduction of Dynamic Website, Introduction to PHP , Why and Scope of PHP , XAMPP and WAMP Installation, **PHP Functions** PHP Functions, Creating an Array , Modifying Array Elements , Processing Arrays with Loops, Grouping Form Selections with Arrays, Using Array Functions, Using Predefined PHP Functions, Creating User-Defined Functions PHP Programming Basics, Syntax of PHP, Embedding PHP in HTML, Embedding HTML in PHP, Introduction to PHP Variable, Understanding Data Types , Using Operators , Using Conditional Statements, If(), else if() and else if condition Statement, Switch() Statements , Using the while() Loop , Using the for() Loop.

UNIT-III: PHP with Laravel: **10 Hours**

Introduction Laravel Editor, Reading and Writing Files , Reading Data from a File , Managing Sessions and Using Session Variables , Destroying a Session , Storing Data in Cookies, Setting Cookies, Dealing with Dates and Times, Executing External Programs.

UNIT-IV: AJAX & Database Connectivity: **10 Hours**

AJAX Introduction, History of AJAX, How does AJAX work?, IE memory leaks, XML HTTP Request - GET or POST, Framework for AJAX, MySQL introduction, creating a database, creating a database table, populating a MySQL database, performing basic database operation (DML): Insert, Delete, Update, Select, Database APIs in PHP, connecting to MySQL with PHP, retrieving data from MySQL, working with retrieved data, creating records with PHP, Updating and deleting records with PHP

Text /Reference Books:

1. Rasmus Lerdorf and Levin Tatroe, Programming PHP – 'O'Reilly Publication,
2. Professional LAMP Linux, Apache, MySQL and PHPs Web Development, Wiley Dreamtech Publication
3. Ivan Bayross, Web Enabled Commercial Application Development Using HTML, JavaScript, DHTML and PHP, BPB Publication
4. Vikram Vaswani, MySQL: The Complete Reference - Tata McGraw-Hill Publishing
5. Steve Holzner, PHP: The Complete Reference Tata McGraw-Hill Publishing.
6. Heera Sheikh, Laravel Reference Guide Paperback by 2016.

*****

Name of Course and code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Value Added Course VAC-IKS VAC-HYW	1. Indian Knowledge System 2. Yoga, Health & Wellness	2+2	0	0	20	20	60	100

This syllabus will be taken from the University basket.

*****

BCA-First Year, II-Semester

Name of the Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Discipline Specific: Major 1 CSE MT--201	Object Oriented Programming Using C++	03	01	00	20	20	60	100

Course Description:: The course on Object Oriented Programming (OOPs) using C++ aims to equip students with a fundamental understanding of OOPs programming concepts and logic through the C++ language. OOPs has become the preferred programming approach by software industries, as it offers a powerful way to cope with the complexity of real world problems.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Understand a comprehensive introduction to OOPs programming concepts using C++.
- To write programs using OOP concepts like Abstraction, Encapsulation.
- To write programs using OOP concepts like Inheritance and Polymorphism.
- The students will be able to develop applications.

UNIT-I: Fundamental of OOP Concepts: 10 Hours

Principles of Object Oriented Programming (OOP), Software Evaluation, A Look at ProcedureOriented Programming, OOP Paradigm, Basic Concepts of OOP, Benefits of OPP, Application of OOP.

UNIT-II: OOP Concept in C++: 15 Hours

Introduction to C++ What is C++, A simple C++ Program, More C++ statements, Structure of C++ Program. Tokens, Expression and controls Structures Tokens , Keywords, Identifiers and Constants, C++ data types, Variables: Declaration, Dynamic initialization of variables, Reference variables, Operators in C++: Scope resolution operator, Member deferencing Operators, Memory Management Operators, Manipulators, Type cast operators, Expressions and Control Structures. Functions The main() function, Function Prototyping, Call by reference, Return by reference, Inline function, Function Overloading.

UNIT-III: Object and Class: 15 Hours

Classes and Objects Introduction, Specifying a Class, Defining member Functions, C++ Program with Class, Nesting of Member functions, Private member functions, Memory Allocation for

Objects, Static Data members, Static Member Functions, Arrays within a Class, Arrays of Objects, Objects as Function Arguments, Friendly Functions, Returning Objects. Pointers : Declaration and initializing, Manipulation of pointers, Pointers to objects, this pointers, Arrays of Pointers to Objects Constructors and Destructors Constructors, Parameterized Constructors, Multiple Constructors in a class, Copy constructor, Destructors. Operator overloading Defining Operator Overloading, Overloading Unary Operators, Overloading Binary Operators, Type Conversions.

UNIT-IV: Inheritance and Polymorphism: 10 Hours

Inheritance and Polymorphisms Introduction, Defining Derived Classes, Single inheritance, Multiple inheritance, Hierarchical inheritance, Multilevel inheritance, Hybrid inheritance, Virtual Base Classes, Polymorphism, static and dynamic binding, Constructor in Derived Classes, Pointers to Derived Classes, Virtual Functions, Pure Virtual Functions.

UNIT-V: Advanced File Handling: 10 Hours

Files – File stream classes – file modes – Sequential Read / Write operations – Binary and ASCII Files – Random Access Operation – Templates – Exception Handling - String – Declaring and Initializing string objects – String Attributes – Miscellaneous functions.

Text/Reference Books:

1. E. Balagurusamy - Object Oriented Programming with C++ - TMH, 8th Edition, 2021.
2. Robert Lafore - Object Oriented Programming in Microsoft C++ - Galgotia, 1999.
3. Ashok N Kamthane, “Object-Oriented Programming with ANSI and Turbo C++”, Pearson Education 2003.
4. Deitel and Deitel, C++ How to Program, Third Edition, Pearson Publication, 2000.
5. Joyce Farrell, Object-oriented programming using C++, Fourth Edition, Cengage Learning, 2013.

*****

Name of course and code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Minor CSE MI-202	Semiconductor and Digital Electronics	04	00	00	20	20	60	100

Course Description: This course provides a foundational understanding of digital logic principles and their application in designing and analyzing digital electronic circuits. Students will gain a comprehensive understanding of core concepts such as digital logic levels, number systems, Boolean algebra, and logic gates, enabling them to analyze and interpret the behavior of

various digital circuits. Furthermore, they will develop the practical skills to design and implement digital electronic circuits for real-world applications.

Course Learning outcomes: On completion of this course, the student will be able to:

- Students will be able to convert between different number systems (decimal, binary, hexadecimal, octal) and perform arithmetic operations in binary, including two's complement representation.
- Students will be able to simplify complex Boolean expressions using Boolean algebra laws and Karnaugh maps, leading to optimized logic circuit designs.
- Students will be able to analyze the characteristics (speed, power consumption, noise immunity) of different logic families and select the appropriate family for a given application.
- Students will be able to design and implement combinational circuits (adders, decoders, multiplexers) and sequential circuits (counters, registers) using appropriate logic gates and design techniques.
- Students will be able to explain the organization of memory systems, address decoding schemes, and the principles of interfacing memory to a processor using buses and memory maps.

UNIT-I Foundations of Digital Electronics: 10 Hours

Background of Digital Electronics, Difference between Analog and Digital Electronics, Sequire web and Digital Signals, Number System, Base of Number System, Types of Number System, Need of Number System, Conversion inbetween number systems, Octal Number System, Hexadecimal Number System, Decimal and Binary Number System, Different Type of Numbering Systems: Decimal, Octal, Binary, Hexadecimal, Conversation from one number system to another number system, Digital Signal, Modulation, Need of Modulation, Type of Modulations.

UNIT-II: Semiconductor Electronics and Boolean Algebra: 10 Hours

Semi-Conductors Electronics, P type and N type Semiconductor, Working of Semiconductor Devices, Diode and Transistor Characteristics, Diode And Transistor As A Switch, MOSFET, E-MOS, H-MOS, Evolution of Logic Gates,

UNIT-III: Logic Gate: **10 Hours**

NOT Gate, AND Gate, OR Gate, Exclusive-OR (XOR) Gate, Truth Tables for Logic Gates, Truth Tables for Combinational Logic.

UNIT-III: Boolean Algebra: 10 Hours

Boolean Algebra and Binary Math: Binary Addition, Binary Subtraction, Binary Complements, One's Complement, Two's Complement, Binary Subtraction using Two's Complement, Signed Magnitude, Need for Boolean Expressions, Symbols of Boolean Algebra, Boolean Expressions of Combinational Logic, Laws of Boolean Algebra, Rules of Boolean Algebra: NOT Rule, OR

Rules, AND Rules, XOR Rules, DeMorgan's Theorem, Standard Boolean Expression Formats: Sum-of-Products, Converting an SOP Expression to a Truth Table, Converting a Truth Table to an SOP Expression, Product - of-Sums, Converting POS to Truth Table, Converting a Truth Table to a POS Expression, NAND-NAND Logic,

UNIT-V: Combinational Circuits and State Machines: 10 Hours

Combinational Applications and State Machines: Creation of Different Combinational Circuits, Karnaugh Maps, Simplification Using Karnaugh Maps, Minimum Sum of Product Expressions Using the Karnaugh Map, "Don't Care" Conditions, Five-and Six-Variable Maps, Multiple Output Problems, Karnaugh Maps, Minimization of two variable Boolean Equaction through K-Map, Three variable Boolean Equaction through K-Map, Four variable Boolean Equaction through K-Map and draw of Minimization Boolean Circuit, Input and Output Table.

Text/Reference Books:

1. R.P. Jain, Modern Digital Electronics, Tata McGraw Hill
2. M. Morris Mano, "Digital Logic and Computer Design", PHI, 1996
3. Louis Neshelsky, "INTRODUCTION TO DIGITAL TECHNOLOGY", John Wiley & Sons, Third Edition, 1983.
4. Digital Logic Design - Ployd.
5. Neil H.E.Weste and Kamran Eshraghian, Principles of CMOS VLSI Design, A System Perspective, Pearson Education, India. 4. Ken Martin, Digital Integrated Circuits, Oxford Press.
6. CMOS Circuit Design, Layout and simulation: J. Baker, D.E. Boyce., IEEE press.

*****

Name of Course and code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Multi Disciplinary Major MDC-FCS-200	Panch Parivartan for Vishwa Guru Bharat	02	01	00	20	20	60	100

Course Description:

This course is designed to inspire and empower students to become leaders of societal transformation through the principles of Panch Parivartan (Fivefold Transformation) for Vishwa Guru Bharat towards Global Well-being. Rooted in the vision of a Developed India @2047, the course emphasizes the need for comprehensive reform at both personal and societal levels. It is not merely an academic exploration but a practical call to action, encouraging students to actively embody the changes they wish to see in the world. The five pillars of Panch Parivartan—Samrasata (social harmony with equality and fraternity), environment friendly

lifestyle stewardship, family awakening, selfhood rooted in Bharatiya values, and civic responsibility—will be explored in depth. Students will not only study these concepts theoretically but will also engage in experiential learning, applying these principles in realworld contexts to create tangible impacts in their communities. Through this course, students will:

- Understand the importance of social harmony and actively practice equality and fraternity in diverse settings.
- Adopt and promote an environment-friendly lifestyle, learning sustainable practices that can be implemented in daily life.
- Explore the concept of family awakening, focusing on the preservation of traditional familial values and their relevance in modern society.
- Develop a strong sense of 'Swa' (selfhood) based on Bharatiya principles, integrating these values into all aspects of personal and professional life.
- Engage in social activism by promoting and adhering to civic duties, leading by example to foster responsible citizenship.

This course aims to transform students into mediators of change, equipping them with the knowledge, values, and skills to drive positive societal transformation. By the end of the course, students will not only be well-versed in the principles of Panch Parivartan but will also be prepared to apply these principles in their lives and become torchbearers of a harmonious, sustainable, and value driven society.

Course Learning Outcomes:

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

1. Demonstrate Social Harmony (Samrasata):

- Understand and promote the principles of equality and fraternity in diverse social contexts.

- Actively contribute to fostering inclusive communities, transcending barriers of caste, religion, and class through real-life initiatives and collaborations.

2. Adopt and Advocate for Sustainable Living:

- Develop and apply knowledge of environmental conservation, adopting an eco-friendly lifestyle in both personal and community spheres.

- Lead sustainability initiatives by practicing and promoting sustainable resource management and environmental protection strategies.

3. Promote Family Enlightenment and Values:

- Recognize the importance of family structures and their role in building a cohesive society.

- Actively engage in family awakening initiatives by upholding and advocating for traditional familial values, ensuring their continuity and relevance in modern contexts.

4. Embody Selfhood ('Swa') Based on Bharatiya Values:

- Integrate Bharatiya values into all aspects of life, cultivating a strong sense of selfhood ('Swa') that informs personal decision-making and social responsibility.

- Serve as an example of self-reliance and self-awareness, using these principles to guide professional and personal endeavors.

5. Foster Civic Responsibility and Ethical Citizenship:

- Understand and adhere to civic duties, demonstrating responsible behavior in public and private spheres.

- Initiate and participate in civic engagement projects that promote ethical behavior, lawfulness, and active citizenship within local and global communities.

6. Lead by Example in Societal Transformation:

- Implement the principles of Panch Parivartan in daily life, serving as a role model for peers and community members.

- Take leadership roles in social, environmental, and civic projects, influencing positive change in the broader society.

7. Apply Theoretical Knowledge to Practical Action:

- Translate theoretical concepts of Panch Parivartan into practical, measurable actions, applying course principles in community engagement, environmental sustainability, and social responsibility projects.

- Reflect on and improve personal and community behaviors through continuous self-assessment and community feedback mechanisms.

8. Contribute to the Vision of Developed India @2047:

- Actively participate in the nation-building process by embodying and advocating for the transformations necessary to achieve a Developed India by 2047.

- Engage in policy discussions, community programs, and leadership roles that align with the broader vision of a harmonious, sustainable, and value-driven society.

These learning outcomes are designed to not only enhance the students' understanding but also to motivate practical action that can positively impact the community, nation, and the world at large.

UNIT-I: Social Harmony **15 Hours**

Introduction to Social Harmony, Perspectives on Social Unity in India, Exploring historical and cultural foundations of social unity and integration, Common Ancestry, Understanding how all people of India are descendants of common ancestors transcending caste and regional differences, Shared Cultural Heritage, Challenges to Social Harmony, Raising the Debate on Social Harmony, Importance of Social Harmony for Future Existence, Policies for Social Integration, Role of Education in Social Harmony, Community Engagement in building social harmony, Society's Role: grassroots movements, Conflict Resolution, Impact of Social Media on Social Harmony, Promoting Positive Narratives, Social Harmony inclusiveness and diversity, Interfaith Dialogue, Wide Entry of Social Harmony activities in Society, Social Harmony and National Strategy, Social Harmony in a Globalized World, Social Harmony is important for Future National Security, Vision for Social Harmony in Bharat 2047, Social Harmony in Ancient India: Case studies.

UNIT-II: Environment Friendly Lifestyle **15 Hours**

Cultivate an understanding of sustainable living practices, Individuals and communities practice towards environmental protection, Eco-friendly lifestyle, Environmental sustainability in today's world, Climate change, deforestation, and pollution, SDGs framework for personal and community action, Understanding the importance of environmental conservation in India's sustainable development, Key Environmental Challenges, Understanding India's commitment to achieving net zero carbon emissions by 2070 and strategies to meet this target, Reduction of carbon footprint, Alternatives to single-use plastics, Urja Swaraj, Energy-saving techniques, Zero-waste lifestyle, Water conservation, Techniques such as rainwater harvesting, Household water use, and greywater recycling, Reducing food waste, Eco-Friendly Transportation and Green Mobility, Advocating for green mobility options such as cycling, walking, public transport etc. Green technologies and grassroots movements, Comprehensive Model of Environmental Protection for Panch Tatva (क्षिति, जल, पावक, गगन, समीरा), Public Participation in Environmental Conservation, Environment-Vision for Bharat@2047.

UNIT-III: Selfhood **15 Hours**

Concept of Selfhood in the context of Bharat's cultural and spiritual identity, examining its relevance for both personal growth and national development, Principles of Swa (selfhood) and how they guide the nation's path towards self-reliance, cultural preservation, and global leadership, Historical, Philosophical, and Practical dimensions of selfhood, Integrate these values of Swa into personal and professional lives, The Concept of Swa and Its Historical Importance, Principles of Swa-trayi: Swadharm, Swadeshi, and Swaraj, Swadharm (inherent righteous duty):

The responsibility to uphold Dharma for a just and harmonious society, Swadeshi (self-sufficiency): Promoting reliance on indigenous resources, supporting local resources for livelihood, and reducing dependence on foreign goods, Swaraj (self-governance): The pursuit of self-governance rooted in Bharatiya values and traditions, The Dharmic Vision and Civilizational Wisdom, Ethical living, and global well-being, Swa as a Foundation for Personal Development, Emphasizing inner growth, self-reliance, and personal responsibility. Practicing Swa for purposeful life aligned with Bharatiya values, Swa as preserving and promoting Bharat's cultural identity in a globalized world, Swa and National Resurgence, Selfhood Beyond Political Independence, Swadeshi and Indian Knowledge Systems, Case Studies in Selfhood and Swa, Vision for Selfhood in Bharat 2047.

UNIT-IV: Family Awakening and Civic Duty **15 Hours**

Family Awakening:

Introduction to Kutumb Prabodhan (Family Awakening) and its relevance in preserving and reviving traditional family structures, Enhancing family bonds and promoting the joint family system as a foundation for societal well-being, The Importance of the Joint Family System: shared responsibilities, cooperation, and the passing down of values and traditions, Knowing Your Ancestors: Exploring Family Lineage, Writing a Genealogy for Your Family, Documenting family history, tracing lineage, and creating a written record to pass down to future generations, preserving the family's heritage, Discussing Heritage within the Family: strengthen family bonds, foster respect for traditions, and promote a sense of belonging and pride in one's cultural roots, The Role of Elders in Preserving Family Heritage: understanding the family's historical legacy and moral foundations, Using AI Technology to Print Portraits of Ancestors, Joint Family Tradition and Ancestral Glories, Challenges of Modern Technology in Family Life, Promoting Weekly Family Time and Heritage Discussions: share stories, and strengthen emotional bonds between different generations, Guidance for Newly Married Couples: Family and Heritage, Making Plans to Take Care of Each Other in the Family, Learning Through Observation: Families as Learning Labs, Case Studies in Successful Family Awakening, Promoting Intergenerational Respect and Communication, Village vs. Urban Family Dynamics, Encouraging Weekly Family Time, Inter-family Interaction and Learning, Addressing the Challenges of Urban Family Life.

Civic Duties

Introduction to Civic Duties (Nagarik Kartavya), Importance of Civic Duties (Nagarik Kartavya) in building a prosperous society, Fulfilling one's duties as a responsible citizen, Civic Rights vs. Civic Duties, The Role of Civic Duties in Nation Building, Upholding the Rule of Law, Responsibilities of Youth and Citizens, Importance of Public Etiquette in a Civilized Society such as maintaining cleanliness, respecting others, and following societal norms, Civic Engagement and Social Development, Encouraging Community Services, Promoting Sustainable Practices through Civic Duty, Strategies for Raising Civic Awareness among Youth, Civic Duties and Social Welfare, Civic Responsibility in a Globalized World, Case Studies in Civic

Engagement, Challenges to Civic Engagement and Strategies to Overcome Them, Inspirational strategies to be responsible citizens who contribute positively to national and social development, Visionary roadmap for Civic Responsibility leading up to Bharat 2047.

Text/ Reference book:

1. Narendra Modi, Social Harmony Prabhat Prakashan Pvt. Ltd.
2. Mohan Bhagwat, Yashasvi Bharat, Prabhat Prakashan
3. Ramdutt Chakradhar, Arunoday Yuva Present and Future, Archana Prakashan Bhopal
4. Ramdutt Chakradhar, Nation Building and Youth, Archana Prakashan Bhopal
5. Sunil Ambekar, Swarnim Bharat Ke Disha-Sootra, Prabhat Prakashan
6. Satish Kumar, Bharat @ 2047 ke panch stambh, Swadeshi Swavlamban Nyas
7. J. Nandakumar, Swa: Rashtriya Svatva ke liye Sangharsh: Ateeth, Vartman aur Bhavishya, Indus Scrolls Press.
8. J. Nandakumar, Lok, Beyond Folk-Reimagining Bharat's Future Based on National Selfhood, Indus Scrolls Press.
9. Shri Atul Kothari, Shiksha Mein Bhartiyata : Ek Vimarsh, Prabhat Prakashan Pvt. Ltd.
10. Dilip Kelkar, Kutumb Sudrudhikaran, Bhishma Prakashan
11. Anil Sathe, KUTUMB PRABODHAN, Generic
12. Akshit Choudhary, Kutumb darshan, Notion Press
13. Khyati Pathak & Anupam Manur & Pranay Kotasthane, We, The Citizens: Strengthening the Indian Republic, Penguin Random House India
14. के. सिद्धार्थ, पारिस्थितिकी, पर्यावरण एवं संरक्षण, Kitab Mahal
15. अव्यग्र प्रताप सिंह , विपिन वि. कांबळे, पर्यावरण संरक्षण, Poetic Essence Publications
16. Dr. A.P.J. Abdul Kalam, Ignited Minds (R/J): Unleashing the power within India, Penguin Random House India

*****

Name of Course and code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Ability Enhancement Course (AEC) AEC-GE	General English	02	00	00	10	10	30	50

This syllabus will be taken from the University basket.

*****

Name of course and code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Skill Enhancement Course (SEC)- SEC-204	Programming with Python	02	01	00	20	20	60	100

Course Description: This course provides a comprehensive introduction to the Python programming language, equipping students with the fundamental knowledge and practical skills necessary to develop Python applications. Students will explore Python's core concepts, syntax, and data structures, progressing to more advanced topics such as object-oriented programming, file handling, and exception management. Through hands-on exercises and real-world examples, students will gain proficiency in writing clean, efficient, and reusable Python code.

Course Learning Outcomes: Upon successful completion of this course, students will be able to:

- Understand the fundamentals of Python: Define scripting languages, articulate Python's features, applications, and limitations, and compare Python to similar languages.
- Utilise core Python components: Configure the Python environment, write and execute Python programs, and demonstrate understanding of identifiers, variables, keywords, data types, operators, control flow, and functions.
- Manipulate Python data structures: Perform operations on sequence and non-sequence data types, including strings, lists, tuples, sets, and dictionaries, with practical programming examples.
- Implement file operations and modules: Open, close, and manipulate text files, import and utilize modules, and effectively manage exceptions in Python programs.
- Apply object-oriented programming principles: Comprehend and implement classes, objects, inheritance, and polymorphism in Python, and differentiate between abstract classes and interfaces.

UNIT- I: Fundamentals of Python: 11 Hours

Introduction to the language, What is Scripting language, Introduction of Python, Features of Python, flavors of Python, Similar languages, Versions of Python, and limitations of Python. Python applications.

UNIT-II: Components of Python Program: 12 Hours

Installation of Python, Execution of Python program, PVM, Comparisons between C and Python, Comparisons between Java and Python, Identifiers, Variables, Keywords, and Data types in Python, Control Flow and Functions, Operator, Operator precedence.

UNIT-III: String, List, Set, and Dictionary: 11 Hours

Strings, Lists, and Their operations with programming Examples and Tuples and Their operations with programming Examples; Sets- Their functions with programming Examples; and Dictionary- Their procedures with Programming Examples.

UNIT-IV: Files, Modules, and exceptions: 11 Hours

Files, Types in files in Python, Opening a file, Closing a file, working with text files. Modules, Importing a module, Importing an entire Module, and under an alias. exceptions- What is an exception, and what happens when an exception occurs?

Text/ Reference book:

1. Rao, R. N. Core Python Programming (2nd ed.). Dreamtech Press, 2020.
2. Brown, M. C. Python: The complete reference (16th ed.). McGraw Hill, 2023.
3. Deitel, P. J., & Deitel, H. M.. Intro to Python. Pearson, 2023.
4. Charles Dierbach, "Introduction to Computer Science Using Python", 1st Edition, Wiley India Pvt Ltd. ISBN-13: 978-8126556014.
5. Wesley J Chun, "Core Python Applications Programming", 3rd Edition, Pearson Education India, 2015. ISBN-13: 978-9332555365.

*****

Name of Course	Name of course and code	Course Title	MM	L	T	P	C
Value Added	VAC-EEDM VAC-TAC VAC-PA	1.Environmental Education and Disaster Management 2.Tribal Art and Culture 3.Performing Arts	100	2+2	0	0	4

This syllabus will be taken from the University basket.

****

BCA - Second Year, III-Semester

Name of Course and code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Discipline Specific: Major 1 CSE MT-301	Fundamentals of Data Structures	03	01	00	20	20	60	100

Course Description:: The course on Fundamentals of Data Structures aims to equip students with a fundamental understanding of basic data structures and their implementations. Students will learn to understand importance of data structures in context of writing efficient programs. The course emphasizes the mastery to develop skills to apply appropriate data structures in problem solving.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Understand the basics of the data structure.
- Implement linear data structure such as stacks, queues, linked lists and their applications.

- Implement basic operations on binary trees.
- Demonstrate the representation and traversal techniques of graphs and their applications

UNIT-I: Introduction: 10 Hours

Elementary data organization, Data Structure definition, Data type vs. data structure, Categories of data structures, Data structure operations, Applications of data structures. Arrays: Arrays definition, representing arrays in memory, various operations (traversal, insertion, deletion), Multidimensional arrays, Sequential allocation, Address calculation.

UNIT-II: Linked List: 10 Hours

Introduction, Array vs. Linked list, Representation and Implementation of Singly Linked Lists, Traversing and Searching of Linked List, Overflow and Underflow, Insertion and deletion to/from Linked Lists, Insertion and deletion Algorithms, Circular Link-list, Doubly Link-list and Application.

UNIT-III: Stack and Queue: 15 Hours

Introduction & Definition, Application of Stack, Various Representation of Stack, Operation on stack (Push and Pop) Hierarchy of Operation, Representation of Arithmetic Expression (Infix, Postfix, Prefix), Evaluation of postfix expressions and their conversions. Queue: Introduction, Applications of Queue, Various Representations of Queue, Operation on queue. Concept of Deque, Priority Queues, Circular Queue.

UNIT-IV: Searching and Sorting: 10 Hours

Definition, Type of Searching (Binary Search, Linear Search). Sorting: Definition, Definition of Sorting, Comparison of Sorting Method, Bubble Sort, Insertion Sort, Selection Sort, Merge sort.

UNIT-V: Introduction of Tree and Graph: 15 Hours

Definition of Trees, Binary Tree, Type of Binary Tree, Operation on Binary Tree, Traversal of Binary Tree, Binary Search Tree (BST), Expression Trees, Memory Representation of Binary Tree. Graph: Definition of Graph, Terminology & Representations of Graphs, Memory Representations of Graphs, Traversal graph BFS and DFS, Spanning Trees, Minimum Cost Spanning Trees (Kruskal & Prims algorithm).

Text/Reference Books:

1. A. S. Tenenbaum, “Data Structures using C & C++”, Prentice-Hall of India Pvt. Ltd., New Delhi, 2000.
2. S. Lipschutz, Data Structures Mc-Graw Hill International Editions, 1986.
3. Leiserson, Charles E, et al. Introduction To Algorithms. India, MIT Press, 2001.
4. R. Kruse et. al, “Data Structures and Program Design in C”, Pearson EducationAsia, Delhi-2002

5. Horowitz and Sahani, "Fundamentals of data Structures", Galgotia Publication Pvt.Ltd.,New Delhi, 2000.

*****

Name of course and code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Discipline Specific: Major 2 CSE MT-302	Operating System	03	01	00	20	20	60	100

Course Description: To understand the fundamental concepts and functionalities of operating systems. To learn about process management, synchronisation, and scheduling. To explore memory management techniques and file systems. To gain practical experience with operating system commands and utilities.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Describe the basic concepts and components of operating systems.
- Implement process management and synchronization techniques.
- Analyze and apply memory management strategies.
- Understand and use file systems and I/O systems effectively.

UNIT-I: Introduction to Operating Systems: 10 Hours

Definition and Functions of Operating Systems, Operating System Architecture, Types of Operating Systems: Batch, Multiprogramming, Time-Sharing, Real-Time, Distributed, and Network Operating Systems, System Calls and APIs, Operating System Structures: Monolithic, Microkernel, Layered.

UNIT-II: Process Management: 15 Hours

Process Concept: Process vs. Thread, Process Scheduling, Cooperating Processes, Process State Model, Process Control Block (PCB), Scheduling: CPU Scheduling Algorithms (FCFS, SJF, Round Robin, Priority), Inter-Process Communication (IPC): Shared Memory, Message Passing, Process Synchronization: Critical Section Problem, Semaphores, Monitors, Deadlock: Deadlock and its Prevention, Deadlock Avoidance, Recovery from Deadlock.

UNIT-III: Memory Management: 15 Hours

Memory Hierarchy, Address Binding: Logical vs. Physical Addressing, Memory Allocation: Contiguous Allocation, Paging, Segmentation, Swapping, Virtual Memory: Demand Paging and Segmentation with Paging, Page Replacement Algorithms (LRU, FIFO, Optimal), Allocation of Frames, Thrashing.

UNIT-IV: Device and Storage Management: **10 Hours**

Device Management: Techniques for Device Management, Dedicated Device, Shared Device, Buffering, Multiple-paths, Storage management: Mass-Storage Structure, Disk Structure, Disk Attachment, Disk Fragmentation, Disk Scheduling, Disk Scheduling Algorithms: FCFS, SSTF, SCAN, C-SCAN, RAID Structure.

UNIT-V: File System Implementation: **10 Hours**

File system interface: File Concept, Access Methods, Directory Structure, File System Structure, Allocation Methods, and Free-Space Management. System Protection: Goals, Principles, Domain of Protection, Access Matrix, Access Control.

Text/Reference Books:

1. Abraham Silberschatz, Greg Gagne, and Peter B. Galvin, "Operating System Concepts, "Tenth Edition, Wiley, 2018.
2. Modern Operating Systems by Andrew S. Tanenbaum, Fourth Edition, Pearson, 2014.
3. Operating Systems: Internals and Design Principles by William Stallings, 7th Edition, Prentice Hall, 2011
4. Deitel & Deitel, Operating systems, 3rd edition, Pearson Education, India, 2008.

*****

Name of course and code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Minor-CSE MI-303	Logic Circuit Design	04	00	00	20	20	60	100

Course Description:

To provide a foundational understanding of digital systems, including the principles of Boolean algebra, logic gates, and circuit design. The course aims to equip students with the skills to analyze, design, and implement combinational and sequential logic circuits.

Course Learning Outcomes:

- Understand the fundamentals of Boolean algebra and apply it to simplify logic expressions.
- Analyze and design combinational logic circuits using logic gates.
- Design and implement sequential logic circuits, including flip-flops, counters, and registers.
- Use hardware description languages (HDLs) for modeling and simulating digital circuits.

- Apply digital logic concepts to solve real-world problems and implement digital systems.

UNIT-I Design of Digital Combinational Circuits: 10 Hours

K-Map Simplifications methods overview, Type of Combinational Circuits, Design of Half Adder and Full adder using K-Map, Two variable, Three Variable and Four Variable Boolean Equation Design, Binary to Gray Code Conversion Using K-Map and Draw Minimize Logic Circuit, and Gray to Binary, Design Seven Segment Display Decoder using K-Map, BCD code to Ex-3 code converter, 84-2-1 code to Ex-3 Code Converter using K-Map, Design of one code to another code conversion using K-Map, Design of Parity code Checker, Design of two varibale code comparator using K-Map, Design of different code converter for various applications.

UNIT-II Design of Digital Sequential Circuits: 10 Hours

Introduction to Flip-Flops, Define Sequential Circuits, Types of Sequential Circuits, Introduction to Counter, Types of Counters, Synchronous Counters and Asynchronous Counters, Design of various type of Counters using K-Map, Register and type of Register, Applications of Register, Uses of Flip-Flop, Counters and Register for Design of Control Unit.

UNIT-III Design of ALU and Introduction to Microprocessor: 10 Hours

Block daigram of ALU, Working pattern of ALU, Definition of Microprocessor, Type of Microprocessor, Microprocessor Family, PIN daigram of Intel 8085 Microprocessor, 40-pin functions, Architectire of Intel 8085 MP, Functions of Intel 8085, Interrrupt, Register organization, Control Unit Organization, From Intel 8080 to Intel i9 processor family specifications.

UNIT-IV Memory And Programmable Logic: 10 Hours

Introduction, Random access memory, memory decoding, error detection and correction, read only memory, programmable logic array, programmable array logic, sequential programmable devices.

UNIT-V Gate Level Minimization 10 Hours

The k-map method, four-variable map, five-variable map, product of sums simplification, don't-care conditions, NAND and NOR implementation, determination and selection of Prime Implicants, Essential and Non essential prime Implicants.

Text/Reference Books:

1. M. Morris Mano, Michael D. Ciletti, Digital Design, 4th edition, Pearson Education Inc, India, 2008.
2. Digital Principles and Design, Tata McGraw Hill, India Roth, 2004.
3. Fundamentals of Logic Design, 5th Edition, Thomson, India.
4. Digital Design-Principle& practice, 3rd edition by John F. Wakerley, Pears
5. A. K. Maini, "Digital Electronics: Principles, Devices And Applications, Wiley, 2007.

*****

Name of Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Multi Disciplinary Major MDC-FCS-300	Innovative Entrepreneurial Solutions	02	01	00	20	20	60	100

Course Description:

"Innovative Entrepreneurial Solutions" is a Multidisciplinary course designed to empower students with the tools and strategies needed to build innovative, sustainable, and impactful businesses across diverse sectors. This course focuses on four critical areas: Wellness Entrepreneurship (targeting underserved populations), Youth Entrepreneurship (harnessing the potential of the next generation), Agricultural Entrepreneurship (empowering farmers and promoting food security), and Women Empowerment Entrepreneurship (advancing gender equality through business).

Students will engage in a comprehensive exploration of industries such as wellness, tribal art and craft, sports and toys, food processing, dairy, ecofriendly products, and traditional cosmetics and fashion. Emphasizing innovation and sustainability, the course provides both theoretical knowledge and practical applications, enabling students to develop entrepreneurial solutions that address realworld challenges. Key areas of focus include the integration of cultural preservation with modern business practices, the development of sustainable products, and the promotion of social and economic empowerment. By the end of the course, students will be equipped to launch and manage enterprises that not only drive economic growth but also contribute to the betterment of society, particularly in underserved and marginalized communities. The course aims to cultivate a new generation of entrepreneurs who are capable of creating meaningful change through innovative solutions, ensuring that their ventures are both profitable and socially responsible. GYAN Entrepreneurship is a comprehensive course designed to equip students with the knowledge and skills needed to create successful and sustainable businesses in various sectors, with a focus on social and economic empowerment. The course covers four key areas: Wellness Entrepreneurship (G for Garib), Youth Entrepreneurship (Y for Yuwa), Agricultural Entrepreneurship (A for Annadata), and Women Empowerment Entrepreneurship (N for Narishakti). Through a blend of theoretical insights and practical applications, students will explore diverse industries, including wellness, tribal art and craft, sports and toys, food processing, dairy, cow dungbased products, and traditional cosmetics and fashion. The course emphasizes sustainable practices, innovation, and cultural preservation while fostering economic development in underserved communities.

Course Learning Outcomes:

- Master Wellness Entrepreneurship: Develop and manage wellness enterprises, including Ayurvedic products and ecofriendly yoga equipment.
- Foster Tribal and Youth Enterprises: Launch and sustain businesses in tribal crafts, sports gear, and traditional toys, preserving cultural heritage.
- Lead Food and Dairy Ventures: Establish and operate food processing and dairy businesses, focusing on innovation and quality.
- Innovate in Sustainable Products: Create milletbased foods and ecofriendly cow dung products, advancing sustainable agriculture.
- Produce Traditional Beauty and Fashion Items: Manufacture and market traditional cosmetics and handcrafted fashion accessories using modern techniques.
- Empower Communities through Entrepreneurship: Drive social impact by empowering marginalized groups through culturally sensitive and sustainable business models.

UNIT-I: Wellness Entrepreneurship: 15 Hours

Wellness centre business scopes, Equipment required to establishment of Wellness centre, DPR for Wellness centre, Services and Product Categories, Panchakarma Kits, Product Categories and Services; Herbal Extracts: Essential Oils, Tinctures, Herbal Concentrates, Herbal Lep and Powders, Herbal Supplements, Yoga Equipment and service and Product Categories, Herbal and Ayurvedic Spa Services and Products, Infusion Process. Ayush Entrepreneurship: Ayurvedic Cosmetics, Manufacturing Line: Emulsion Techniques, Active Ingredient Integration, Stabilization and Preservation, Packaging.

UNIT-II: Tribal Entrepreneurship: 15 Hours

Traditional Handicrafts: Product Categories: Tribal Jewelry, Pottery, Metal Art products, Beadwork, Handstringing beads; metal casting and shaping for traditional jewelry; patina application for aged effects. Wooden Art products, Clay Preparation, Woven Textiles, Weaving Techniques: Warp and weft techniques on traditional handlooms; natural dyeing processes using plantbased dyes; handembroidery for added texture, Bamboo Products and Manufacturing Line, Wooden Artifacts and Manufacturing Line: Hand Carving and Decorative Techniques: Inlay work, painting, and burning for added artistic value; use of natural pigments and traditional patterns. Sports and Toy Products Youth Entrepreneurship: Traditional Toys and Product Manufacturing Line, Woodturning, Clay Molding, Educational Toys: Puzzles: Jigsaw puzzles, tangrams, and 3D puzzles, Construction Sets: Wooden or plastic blocks, interlocking pieces, Board Games: Traditional and educational board games and Injection molding for plastic toys. Sports Equipment and Product: Cricket Bats: Willow and bamboo bats, handfinished for optimal weight and balance. Hockey Sticks: Wooden and composite sticks, shaped for flexibility and strength. Gym Equipment: Dumbbells, barbells, kettlebells crafted from iron, steel, and composite materials.

UNIT-III: Food Processing and Dairy Entrepreneurship: 15 Hours

Dairy and Millet Dairy Products: Cheese, Yogurt, Butter and Ghee, Flavored Milk: Chocolate, vanilla, almond, and fruitflavored milk. Manufacturing Line: Milk Collection and Testing,

Pasteurization, Fermentation, Churning and Clarification, Homogenization, Dairy and Allied Dairy Products: Whey Protein, Lactose Free Milk, Packaging. Millet Based Products: Millet Flour, Millet Snacks, Traditional Millet Based Foods and ready to eat products, Manufacturing Line: Cleaning and Sorting, Milling, Extrusion, Baking and Frying, Quality Control: Testing for moisture content, shelf life, and nutritional value, Food Processing products and DPRs, Cow Dung-Based Products: Agarbatti (Incense Sticks), Hawan Samagri (Sacred Ritual Materials), Biodegradable Gamla (Plant Pots), Diyas (Lamps), Dung Logs & Cakes, Vermicompost, Herbal Mosquito Repellent, Bio CNG, CBG into cylinders or direct pipeline distribution; ensuring safety standards, Organic Bio Fertilizers.

UNIT-IV: Women Entrepreneurship **15 Hours**

Traditional Cosmetic Products & Bridal business, Skincare and cosmetic Products: Creams, Face Packs, Antiaging creams, Oils, Manufacturing Line: Herb Preparation, Cream Formulation, Oil Blending and Packaging, Handcrafted Fashion Items.

Text Reference Books:

1. Dr. Vasant Lad, Ayurveda: The Science of Self-Healing, Lotus Press
2. P. J. Fellows, Food Processing Technology: Principles and Practice, Woodhead Publishing
3. Shrikant Prasad Srivastava, Indian Handicrafts, National Book Trust
4. R. K. Robinson, Modern Dairy Technology: Advances in Milk Products, Springer
5. H. Panda, Herbal Cosmetics Handbook, Asia Pacific Business Press Inc.
6. Charles W. Fetrow and Juan R. Avila, The Complete Guide to Herbal Medicines, Pocket Books
7. Aditi Ranjan and M.P. Ranjan, Handmade in India: A Geographic Encyclopedia of Indian Handicrafts, Abbeville Press
8. Y.H. Hui, Dairy Science and Technology Handbook, Wiley-Interscience
9. Kate Fletcher and Lynda Grose, Fashion and Sustainability: Design for Change, Laurence King Publishing
10. Suresh Kumar, Advances in Food Processing Technology, CRC Press

*****

Name of Course and code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Ability Enhancement Course (AEC) AEC-CS	Communication Skill	02	00	00	10	10	30	50

This syllabus will be taken from the University basket.

Name of Course and code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Skill Enhancement Course (SEC): SEC-305	Computer Graphics	02	01	0	20	20	60	100

Course Description: This subject helps students develop problem-solving, communication and research skills in the context of computer graphics, including computer representation, and manipulation and display of pictorial information. It also helps students enhance their skills to design and implement three-dimensional (3D) computer images, such as those used in animated films, virtual reality (VR), data visualization and computer games.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Understand the basics of the computer graphics.
- Describe the major computer graphics applications.
- Explain the operations of graphics hardware systems.
- Explain the viewing pipeline of generating a raster-scan image.
- Understand the concept of 2D and 3D transformation.

UNIT-I: Basics of Computer Graphics: **10 Hours**

Introduction, what is computer Graphics?, Basic elements of Computer graphics, Applications of Computer Graphics. Graphics Hardware, Video Display Devices, Architecture of Raster and Random scan display devices, Input devices, Hard-copy devices, Graphics software.

UNIT-II: Graphics Algorithms and Devices: **10 Hours**

Graphics Monitors, Input Devices, Points and Lines; Line Drawing Algorithms, Mid-Point Circle and Ellipse Algorithms; Scan Line Polygon Fill Algorithm, Boundary-Fill and Flood- Fill.

UNIT-III: 2D Graphics Transforms and Viewing: **10 Hours**

2-D Geometrical Transforms and Viewing: Translation, Scaling, Rotation, Reflection and Shear Transformations; Matrix Representations and Homogeneous Coordinates; Composite Transforms, Transformations Between Coordinate Systems, Viewing Pipeline, Viewing Coordinate Reference Frame, Window to View-Port Coordinate Transformation, Viewing Functions, Line and Polygon Clipping Algorithms.

UNIT-IV: 3D Graphics and Transformations: **15 Hours**

3-D Object Representation, Geometric Transformations and Viewing: Polygon Surfaces, Quadric Surfaces, Spline Representation, Bezier and B-Spline Curves; Bezier and B-Spline Surfaces;

Illumination Models, Polygon Rendering Methods, Viewing Pipeline and Coordinates; General Projection Transforms and Clipping.

Text/Reference Books:

1. Computer Graphics C Version, Donald Hearn and M Pauline Baker, Pearson Education, 2nd edition, 2006
2. Introduction to Computer Graphics, J.D. Foley, A.V. Dam, Addison-Wesley Publishing Company, 2nd edition, 1994.
3. Computer Graphics (Schaums Outline Series), R.A. Plastock et.al., TMH, 2nd edition, 2006
4. Computer Graphics, J.D. Foley, Pearson Education, 2nd edition, 2004
5. J. D. Foley, A. Van Dam, S. K. Feiner and J. F. Hughes, Computer Graphics - Principles and Practice, Second Edition in C, Pearson Education, 2003.

*****

BCA: Second Year, IV-Semester

Name of Course and code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Discipline Specific: Major 1: CSE MT-401	Core Java Programming	03	01	00	20	20	60	100

Course Description: The objective of this course is to introduce students to the fundamental concepts and techniques of Java programming. Students will learn object-oriented programming (OOP) principles, develop problem-solving skills, and gain hands-on experience in building Java applications. The course aims to equip students with the knowledge and skills necessary to develop robust, efficient, and maintainable Java software.

Course Learning Outcomes:

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

- Understand Java Basics: Demonstrate an understanding of basic Java syntax, data types, and control structures.
- Implement Object-Oriented Concepts: Apply object-oriented programming principles such as encapsulation, inheritance, and polymorphism to create well-structured Java applications.
- Develop Java Applications: Write, compile, and execute Java programs that solve real-world problems using classes, objects, methods, and arrays.
- Handle Exceptions: Implement error handling and exception management in Java to build robust and fault-tolerant applications.

- Utilize Java Libraries: Effectively use standard Java libraries and APIs, including those for data structures, file I/O, and networking.
- Implement GUI Applications: Design and develop graphical user interfaces (GUIs) using Java's AWT and Swing libraries, Understand Multithreading and work with Java Collections.

UNIT-I Introduction to Java Programming: 12 Hours

Introduction, Programming language Types and paradigms, Computer Programming Hierarchy, How Computer Architecture Affects a Language? Why Java? Flavors of Java, Java Designing Goal, Role of Java Programmer in Industry, Features of Java Language, JVM –The heart of Java, Java's Magic Bytecode.

The Java Environment: Installing Java, Java Program Development, Java Source File Structure, Compilation, Executions.

UNIT-II Object Oriented Programming through Java: 12 Hours

Introduction to OOP, Classes and Objects, OOP principles, Encapsulation, Abstraction, Inheritance and Polymorphism, Constructors, initializing reference variables using constructors, Pass by value v/s pass by reference.

UNIT-III Java Inheritance and Polymorphism: 12 Hours

Extending Classes and Inheritance Use and Benefits of Inheritance in OOP, Types of Inheritance in Java, Inheriting Data members and Methods, Role of Constructors in inheritance, Overriding Super Class Methods, Use of "super", Polymorphism in inheritance, Type Compatibility and Conversion Implementing interfaces.

UNIT-IV Java Packages and I/O: 12 Hours

Input/output File handling, Exploring Java IO Byte oriented classes and character-oriented classes. Defining package, importing packages, creating user defined packages, packages in Java. String handling.

UNIT-V Exception handling: 12 Hours

Exceptions Fundamentals of exception, Exception types, using try & catch, multiple catch, nested try, throw, finally, built-in exception, user defined exceptions.

Text/ Reference Books:

1. R.Nageswara Rao, CORE JAVA An Integrated Approach, , Dreamtech Press
2. Naughton, Patric & Schildt, Herbert. Java 2 Complete Reference. TMH. Stephen Potts, Alex Pstrikov
3. Object-Oriented Programming with C++ and Java Debasis Samanta, Prentice Hall India.
4. E. Balagurusamy, Programming with Java, Tata McGraw-Hill Education India, 2014
5. Programming in JAVA, 2nd Ed, Sachin Malhotra & Saurabh Choudhary, Oxford Press

*****

Name of Course and code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Discipline Specific: Major 2 : CSE MT-402	Database Management System	03	01	00	20	20	60	100

Course Description: To develop skills in database design, implementation, and management. To familiarize students with SQL and its practical applications. Construct simple and moderately advanced database queries using Structured Query Language (SQL). Understand and successfully apply logical database design principles, including E-R diagrams and database normalization.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Understand the concepts and principles of database management systems.
- Design and create relational databases using SQL.
- Query and manipulate data using SQL commands.
- Apply normalization techniques to ensure data integrity.

UNIT- I: Introduction to Databases: **10 Hours**

Introduction of File Organization, Database Management system, Characteristics of a Database, Database Administrators, Data, Information, and Knowledge, File system vs DBMS, Types of Database system, Data Dictionary, Advantage and Disadvantages of databases systems, Data Models, Schemas & Instances, DBMS Architecture & Data Independence, Level of Abstraction, Data Languages & Interfaces.

UNIT- II: Relational Database and E-R modeling: **15 Hours**

Relational database concepts: tables, tuples, attributes, keys, etc., Entity-Relationship (ER) modeling: Data modeling using the Entity-Relationship Approach. E-R Modeling: Entity types, entity set, attribute and key, relationships, relation types, roles and structural constraints, weak entities., Codd's rules, Relational Schemas, Constraints: Domain Constraints, Key constraints or Uniqueness Constraints, Entity Integrity constraints, Referential integrity constraints. Relational Algebra and Calculus.

UNIT- III: Data Dependency and Normalization: **10 Hours**

Introduction, Anomalies, Normalization, Purpose of normalization, Data Dependency, Armstrong's Axioms, Function Dependency, Normal Forms (1NF, 2NF, 3NF, BCNF), Denormalization and its implications.

UNIT- IV: Database Language: **15 Hours**

What is SQL, Database Languages: DDL, DML, DCL, TCL, Filtering, Sorting, and Aggregating Data, JOINs (INNER, LEFT, RIGHT, FULL), Subqueries and Nested Queries, Stored Procedures and Triggers, views and indexes. Introduction to PL SQL.

UNIT- V: Transaction management and Concurrency control: 10 Hours

Transaction management: ACID properties, serializability and concurrency control, Lock based concurrency control (2PL, Deadlocks), Time stamping methods, optimistic methods, database recovery management. Deadlock: Detection, Avoidance, and Recovery.

Text/Reference Books:

1. Silberschatz, A., Korth, H.F., Sudarshan, S., Database System Concepts, McGraw-Hill International Edition, 2006 (5 th Edition)
2. Elmasri, R., Navathe, S.B., Fundamentals of Database Systems, Fourth Edition, Pearson Education,
3. Desai, B.C., An Introduction to Database Systems, Galgotia Publications.
4. Date, C.J., An Introduction to Database Systems, Pearson Education, 7 th Edition
5. Garcia-Molina, H., Ullman, J.D., Widom, J., Database Systems: The Complete Book, Pearson Education, 2002.

*****

Name of Course and code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Discipline Specific: Major 3 : CSE MT-403	Design and Analysis of Algorithms	03	01	00	20	20	60	100

Course Description: The objective of this course is to provide students with a solid foundation in algorithm analysis and design techniques. The course aims to develop students' skills in solving computational problems, analyzing algorithm complexity, and designing efficient algorithms.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Understand the fundamentals of algorithm analysis and design.
- Apply algorithmic techniques to solve computational problems.
- Design and implement efficient algorithms for real-world scenarios.
- Analyze the time and space complexity of algorithms

UNIT-I : Basics of algorithms: 15 Hours

Introduction to Algorithms and problem-solving techniques, pseudo code for expressing algorithms, Average, Best and worst case analysis, Asymptotic notation, time and space complexity, recurrences, probabilistic analysis, disjoint set operations, union and find algorithms,

Algorithm design paradigms, Sorting Algorithms and analysis: Bubble sort, Selection sort, Insertion sort, Sorting in linear time : Bucket sort, and Radix sort.

UNIT-II: Divide And Conquer: 15 Hours

Introduction, Recurrence and different methods to solve recurrence, Multiplying large Integers Problem, Problem Solving using divide and conquer algorithm - Binary Search, Max-Min problem, Sorting (Merge Sort, Quick Sort), Matrix Multiplication, Exponential. Greedy Method: General Characteristics of greedy algorithms, Problem solving using Greedy Algorithm, Huffman code.

UNIT-III : Graphs and Dynamic Programming: 10 Hours

An introduction using graphs and games, Undirected Graph, Directed Graph, Traversing Graphs, Depth First Search, Breadth First Search, Topological sort, Connected components, Topological sorting, Single Source Shortest paths, Minimum Spanning trees (Kruskal's algorithm, Prim's algorithm). Problem Solving using Dynamic programming, Knapsack problem, All Points Shortest path, Matrix chain multiplication, Longest Common Subsequence.

UNIT-IV: Backtracking and Branch and Bound: 10 Hours

General method, Applications- n-queen problem, Sum of subsets problem, Graph coloring and Hamiltonian cycles. Branch and Bound: General method, applications - traveling salesperson problem, 0/1 knapsack problem- LC branch and bound solution, FIFO branch and bound solution.

UNIT-V: NP-Hard And NP-Complete Problems: 10 Hours

Basic concepts, non-deterministic algorithms, NP-hard and NP-complete classes, the P vs. NP problem, Cook's theorem, Travelling Salesman problem, Hamiltonian problem.

Text/Reference Books:

1. "Introduction to the Design and Analysis of Algorithms" by Anany Levitin, 3/e, 2017.
2. "Algorithm Design Manual" by Steven S. Skiena, 2010.
3. AnanyLevitin. Introduction to the design and Analysis of Algorithms. Dorling Kindersley (India) Pvt.Ltd. Second edition, 2008.
4. Gav PAI, Data Structures and Algorithms, Tata McGraw Hill, Jan 2008.
5. Donald E. Knuth. The art of Computer Programming,Volume 1: Fundamental Algorithms.Addison Wesley, 1997.

*****

Name of Course and code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Discipline Specific: Major 4 CSE MP-404	Core Java Programming Laboratory	00	00	02	00	00	50	50

Course Description:

The Core Java Programming Lab course is designed to provide hands-on experience in developing Java applications. Students will learn the fundamentals of Java programming, including object-oriented principles, data structures, and basic algorithms. The lab exercises will cover essential topics such as control structures, classes and objects, inheritance, exception handling, file I/O, and basic GUI programming. This course aims to strengthen the understanding of Java concepts through practical implementation, preparing students to develop robust and efficient Java applications.

Course Learning Outcomes:

By the end of the Core Java Programming Lab, students will be able to:

- Write and execute basic Java programs using control structures and data types.
- Implement object-oriented programming concepts such as classes, inheritance, polymorphism, and encapsulation in Java.
- Handle exceptions and perform file I/O operations in Java.
- Develop simple Java-based GUI applications.
- Apply core Java concepts to solve real-world programming problems.

Laboratory Exercises:

1. Write a program to find the average and sum of the N numbers Using Command line argument.
2. Write a program to Demonstrate Type Casting.
3. Write a program to find the number of arguments provide at runtime.
4. Write a program to Test the Prime number.
5. Write a program to calculate the Simple Interest and Input by the user.
6. Write a program to create a Simple class to find out the Area and perimeter of rectangle and box using super and this keyword.
7. Write a program to find G.C.D of the number.
8. Write a program to design a class account using the inheritance and static that show all function of bank (withrowal, deposite).
9. Write a program to find the factorial of a given number using Recursion.
10. Write a program to design a class using abstract Methods and Classes.
11. Write a program to design a String class that perform String Method(Equal,Reverse the string,change case).

12. Write a program to handle the Exception using try and multiple catch block.
13. Write a program that Implement the Nested try Statements.
14. Write a program to Create a package that access the member of external class as well as same package.
15. Write a program that import the user define package and access the Member variable of classes that Contained by Package.
16. Write a program that show the partial implementation of Interface.
17. Write a program to Handle the user defined Exception using throw keyword.
18. Write a program to create a thread that Implement the Runnable interface.
19. Write a program to Implement Interthread communication.
20. Write a program to create a class component that show controls and event handling on that controls.(math calc).

****

Name of course and code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Minor 1 CSE MI-405	Discrete Mathematics	02	00	00	10	10	30	50

Course Description: The course on Discrete Mathematics aims to introduce concepts of mathematical logic for analyzing propositions and proving theorems. Use sets for solving applied problems, and use the properties of set operations algebraically.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Write an argument using logical notation and determine if the argument is or is not valid.
- Determine when a function is 1-1 and "onto".
- Demonstrate the ability to write and evaluate a proof or outline the basic structure of and give examples of each proof technique described.
- Understand the basic principles of sets and operations in sets.
- Understand the Posets and Hasse diagram.

UNIT-I: Set Theory and Functions: **10 Hours**

Set theory, Principle of mathematical induction, Relation: Type and composition of relations, Pictorial representation of relations, Closures of relations, Equivalence relations, Partial ordering relation. Function: Types, Composition of function, Recursively defined function.

UNIT-II: Algebraic Structures: **10 Hours**

Group, Monoid, Semi group, Abelian group, Properties of group, Subgroup, Cyclic group, Cosets, Permutation groups, Homomorphism, Isomorphism and Automorphism of groups.

UNIT-III: Propositional Logic: 10 Hours

Proposition, Tautologies, Contradictions, Algebra of Proposition, Logical implication, Logical equivalence, Normal forms, Predicates and quantifiers. Lattices: Introduction, Ordered set, Posets, Hasse Diagram, Hasse diagram of partially ordered set, Consistent enumeration, Isomorphic ordered set, Well ordered set, Properties of lattices, Bounded lattices, Distributive lattices, and Complemented lattices.

Text/Reference Books:

1. Liptschutz, Seymour, "Discrete Mathematics", TMH
2. Trembley, J.P & R. Manohar, "Discrete Mathematical Structure with Application to Computer Science", TMH
3. David Liben-Nowell, Discrete Mathematics for Computer Science, Wiley publication, July 2017.
4. C.L.Liu, "Elements of Discrete Mathematics", McGraw Hill"
5. K. H. Rosen, Discrete Mathematics and its applications, Tata McGraw-Hill, 6th Ed., 2007.

*****

Name of course and code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
(Minor 2) Vocational CSE MI-406	Database Management System(DBMS) Laboratory	00	00	02	00	00	50	50

Course Description: This course provides a comprehensive overview of database system design and implementation. Students will explore the fundamental concepts, techniques, and challenges involved in building and managing robust and efficient database systems

Course Learning Outcomes: On completion of this course, the student will be able to:

- Demonstrate an understanding of the relational data model.
- Transform an information model into a relational database schema and to use a data definition language and/or utilities to implement the schema using a DBMS.
- Formulate, using relational algebra, solutions to a broad range of query problems.
- Formulate, using SQL, solutions to a broad range of query and data update problems.

List of Exercises:

1. Design a Database and create required tables. For e.g. Bank, College Database
2. Create the student/employee Table and construct the following requires for the database:
    a. Apply the constraints like Primary Key , Foreign key, NOT NULL to the tables.
    b. Find out name of all students.

   c. Retrieve the list of name and the city of all students.
   d. List of all students/employee who stay in city ―BOMBAY or city DELHI.
   e. List of all students /employee who are located in ―MADRAS.
3. Write a sql statement for implementing ALTER,UPDATE and DELETE
4. Write the queries to implement the joins
5. Write the query for implementing the following functions: MAX(),MIN(),AVG(),COUNT()
6. Write the query to implement the concept of Intergrity constrains
7. Write the query to create the views
8. Perform the queries for triggers
9. Perform the following operation for demonstrating the insertion , updation and deletion using the referential integrity constraints
10. Write the query for creating the users and their role.
11. Create table with attributes emp. No., emp. Name, Designation, Salary, and Department no. Construct for following queries:
    a. Display complete information of all the employees working as a manager.
    b. Display name of all the employees working as a clerk.
    c. Suppose DA for manager is 75% of salary then display name of all managers.
    d. Select names and designation whose salary is greater then 15000.
    e. Apply key constraints as Primary Key, Foreign Key etc as per requirement.
    f. Between operation- list of all Employee Name & DOJ (date of joining) to join the Company in 2010
12. AND/OR operation- make a table that have an employee Perform AND/ORoperation.
13. Group by function- BCA 85 Create the table for facilities having faculty-id, dept. no., designation name and group by similar dept.no. Facilities by using count function.
14. Create a table for emp. Using following data:- emp. name, emp age, emp salary, emp city & display the emp salary in ascending and descending order.
15. Max-Min function- create a table for students having similar attributes s_name, S_marks, s_id, s_sec&remark.
    a. Find the maximum marks obtained by students.
    b. Find the minimum marks obtained by students.
    c. Sum of all students marks using sum function.
    d. Find the average of marks using avg function.
16. Hands on PL-SQL with Laboratory exercises: 2 programs will be given at the lab.

Text/Reference Books:

1. Database system concepts, design and applications, S.K. Singh, Pearson Education, New Delhi.
2. Sql/PL/SQL, Batross, Ivan BPB, 2010.

*****

Name of course and code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Ability Enhancement Course (AEC) AEC-ICT	Information and Communication Technology	02	00	00	10	10	30	50

Course Description: To provide students with an understanding of the fundamental concepts of ICT and its applications in communication systems. The course aims to equip students with the skills to effectively utilize ICT tools and technologies for efficient information management, communication, and problem-solving in various domains.

Course Learning Outcomes:
Upon successful completion of the course, students will be able to:

- Demonstrate a clear understanding of ICT concepts and their role in modern communication systems.
- Apply ICT tools to enhance communication efficiency in personal, academic, and professional contexts.
- Analyze and solve problems related to information management and communication using appropriate ICT solutions.
- Understand the ethical and societal implications of ICT in communication.

UNIT-I: Fundamentals of Internet: 10 Hours
What is Internet?, Internet applications, Internet Addressing- Entering a Web Site Address, URL-Components of URL, Searching the Internet , Browser-Types of Browsers, Introduction to Social Networking- Twitter, Facebook, Skype, Tumblr, LinkedIn, Youtube, WhatsAap.

UNIT-II: Email Basics and G-Suite: 10 Hours
Basics of electronic mail; Getting an email account; Sending and receiving emails; Accessing sent emails; Using Emails; Document collaboration; Mail management.
G-Suite: Google drive, Google documents, Google Spreadsheets, Google Slides and Google forms.

UNIT-III: Internet Security and Digital Initiatives: 10 Hours
Overview of Internet Security, Email threats and secure Emails, Viruses and antivirus softwares, firewalls, Cryptography, Digital signatures, Copyright issues.
What are GOI digital initiatives in higher education ? (SWAYAM, SWAYAM Prabha, National Digital Library (NDL), e-Yantra, FOSSE, Spoken Tutorials, Virtual Labs, NPTEL)

Text / References Books:

1. Fundamentals of Information and Communication Technology (ICT), Prasun Barua, Amazon Digital Services LLC - Kdp, 2023
2. Information And Communication Technology (ICT), T. Verma, Notion Press, 2021
3. Information and Communication Technology (ICT), 2019, Dr. Sanjay Kumar Khokhar,
4. Information and Communication Technology (ICT), 2021, Dr. Desai Sikandar Siddi Nawab
5. Information and Communication Technology (ICT), Ibrahim Ahmad, 2007,

*****

BCA - Third Year, V-Semester

Name of Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Discipline Specific: Major 1: CSE MT-501	.NET Programming	03	01	00	20	20	60	100

Course Description: To learn about the architecture and environment of.NET Technologies and to examine their features. To become proficient in C# programming to create console applications as well as window applications. To become proficient with C# libraries and exception-handling methods. Acquire knowledge of ADO.NET and advanced C# features. Develop an understanding of .NET assemblies and properties.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Know about the.NET Framework, its runtime environment, and Visual Studio's application development IDE.
- Utility of object-oriented concepts while creating applications.
- Create standalone console and window form programs to demonstrate C# language constructs.
- Acquire knowledge of ADO.NET database concepts and use them to build distributed data-driven applications.
- Create, record, and troubleshoot ASP.NET web forms that incorporate server and validation controls and implementation of ASP.NET web services.

UNIT-I: .NET Framework Overview: **10 Hours**

Overview of .NET Framework: Introduction,.NET Compatible Languages, .NET Framework Architecture, Common Language Runtime(CLR), Common Type System(CTS), Common

Language Specification(CLS), Microsoft Intermediate language(MSIL), Just-in-Time Compilation, .NET Framework Classes, OOPs Concepts.

UNIT-II: Introduction to C#: **10 Hours**

Structure of C#, System. Console Class, System. Object Class, Data Types, Identifiers, Variables, Constants, Literals, Arrays and Strings, Classes and Objects, Inheritance, Interfaces, Polymorphism, Properties, Delegates and Events, Indexes, Boxing and Unboxing.

UNIT-III: C# Using Libraries: **10 Hours**

Namespace-System, System Collections, Input Output, Managing Console I/O Operations, Window Forms, Reflection, Error and Exception, Versioning, Multi-Threading, Unsafe Mode.

UNIT-IV: Advanced Features Using C#: **15 Hours**

Windows Services, Web Services, ASP.NET Web Form Controls, C# in Web Application, ADO.NET, Distributed Application in C#, Graphical Device Interface with C#, Localization and Globalization.

UNIT-V: .NET Assemblies and Attributes: **15 Hours**

Net Assemblies Features, Structure of Assemblies, Components of Assembly, Calling Assemblies, Private and Shared Assemblies, Built-In Attributes and Custom Attributes, Introduction about Generic.

Text/Reference Books:

1. Jeff Jeffrey Richter, "Applied Microsoft .Net Framework Programming", Microsoft Press.
2. Karli Watson ,Jacob Vibe Hammer, "Beginning Visual C# 2012 Programming", Wiley India.
3. Fergal Grimes, "Microsoft .Net for Programmers", SPD.
4. Christian Nagel, Jay Glynn, Morgan Skinner, "Professional C#", Wiley India.
5. Tony Baer, Jan D. Narkiewicz, Kent Tegels, "Understanding the .Net Framework", Wrox.
6. Balagurusamy, "Programming with C#", TMH.
7. Shibi Panikkar and Kumar Sanjeev, "C# with .NET Framework", Firewall Media.

*****

Name of course and code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Discipline Specific: Major 2 CSE MT-502	Software Engineering	03	01	00	20	20	60	100

Course Description: This course introduces the concepts and methods required for the construction of large software intensive systems. It aims to set these techniques in an appropriate engineering and management context. The course aims is to develop a broad understanding of

the discipline of software engineering and management of software systems. This will provide he detailed knowledge regarding the concepts of OOPS, various testing techniques, cost analysis and software reliability and quality assurance.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Basic knowledge of various processes to be followed in the software development life-cycle models.
- Implement communication, modeling, construction and deployment practices in software development.
- Analyze & design the software models using the concepts of OOPs.
- Explain the concepts of various software testing methods & be able to apply appropriate testing approaches for development of software.
- Explain the project management & different types of metrics used in software development and risk and cost benefit analysis is implemented.
- Apply the concepts of project management & planning.
- Apply the concept of software reliability and quality assurance is implemented.

UNIT-I: Software Engineering Fundamentals: 10 Hours

Introduction, Software Perspective, Nature of Software, Types of Software, Software Dependability, Software Characteristics, Product Characteristics, Software Engineering Perspective, Achievements and Challenges, Purpose of Software Engineering, A Generic View, Need and Significance, Criticism, Principle of Software Engineering, Software Engineering Approach, Process Development, Development Life Cycle, Project Management, Project Planning, Project Organizing and Staffing, Project Directing, Project Controlling, Quality Management, Configuration Management

UNIT-II: Software Process: 10 Hours

Introduction, Software Process Characteristics, Process Improvement, Process Modeling, Process Modeling Description, Software Life Cycle, Software Life Cycle Models: Waterfall Model, Build-And-Fix Model, The Spiral Model, Incremental Model, Prototyping Model, RAD Model, Win-Win Spiral Model, Formal System Development Model, V-Model, Agile Model

UNIT-III: Software Requirement: 10 Hours

Introduction, Specifying Software Requirement, Need and Importance, Goals and Objectives, Requirement Analysis, Requirement Analysis Process, Software Requirement Elicitation, Elicitation Techniques, SMART Requirement, Requirement Specification Checklist, Types of Requirements, Requirement Testing, Quality Software Requirement

UNIT-IV: Software Design, Coding and Testing: 15 Hours

Introduction, Software Design Viewpoint, Design Characteristics, Good Design, Design Principles, Ben Shneiderman's Design Principle, IBM's Design Principles for Tomorrow, Design Checklist, Software Design Process, Preliminary Design, Detailed System Design. Object

Oriented Design Approach, **Software Coding:** Programming Practice, Programming Style. **-Software Testing:** Testing Fundamentals, Top Down and Bottom Up Approaches, Functional Testing, Structural Testing, Test Cases and Test Criteria, Software Testing Strategies, Testing Levels, Unit Testing, Integration Testing and System Testing, Alpha and Beta Testing, Test Plan, Test Case Specification, Test Case Execution and Analysis.

UNIT-V: Software Maintenance: 15 Hours

Software as an Evolutionary Entity, Need for Maintenance, Categories of Maintenance, Cost of Maintenance, Software Configuration Management Activities**,** Constructive Cost Models (COCOMO), Introduction to software Project Management, Project managers roles and responsibilities, Project planning**,** Software Reliability and Quality Assurance: Software Reliability, Reliability Issues, Musa's Model, Reliability Measurement Process, Software Quality, Software Quality Assurance, Software Quality Standards, SEI Capability Maturity Model, Comparison between ISO and CMM Model, CASE (Computer Aided Software Engineering): Various CASE Tools and their Usefulness, Role of Data Dictionary in CASE Tools, Architecture of CASE Environment, CASE support in Software life cycle.

Text/Reference Books:

1. Pankaj Jalote, "Software Engineering", Narosa Publications.
2. Rajib Mall, "Fundamental of Software Engineering", PHI.
3. R A Khan, A Agrawal, Software Engineering: A Practitioners Approach, Narosa Publication
4. Roger S. Pressman, "Software Engineering: A practitioner's Approach", McGraw Hill International Edition.
5. Sommerville, "Software Engineering", 7th edition, Pearson education.
6. K.K. Agarwal and Yogesh Singh, "Software Engineering", New Age International Publishers.
7. James F. Peters, Witold Pedrycz, "Software Engineering, an Engineering approach", John Wiley.
8. Shely Cashman Rosenblatt, "Systems Analysis and Design", Thomson Publications.
9. Waman S Jawadekar, "Software Engineering principles and practice", TMH

*****

Name of course and code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Discipline Specific: Major 3 CSE MT-503	Computer Networks	03	01	0	20	20	60	100

Course Description: To understand the fundamental concepts of computer networking and their practical applications. To gain hands-on experience with network configuration and

management. To analyze and troubleshoot networking issues effectively. To explore current trends and future directions in network technologies.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Explain the importance of data communications and the Internet in supporting and daily activities.
- Explain how communication works in data networks and the Internet.
- Recognize the different internetworking devices and their functions.
- Explain the role of protocols in networking.

UNIT-I: Introduction to Networking and Physical Layer: 10 Hours

Definition and Importance of Networking, Network Types: LAN, WAN, MAN, Network Models: OSI and TCP/IP, Network Topologies and Components, Overview of Network Devices: Routers, Switches, Hubs, Modems, The Physical layer: Data and Signals, Transmission impairment, Data rate limits, Performance, Transmission media: Introduction, Guided Media, Unguided Media, Switching, Multiplexing, ISDN.

UNIT-II: Data Link Layer: 15 Hours

Data Link Layer Concepts: Introduction, Link layer addressing, Error detection and Correction: Cyclic codes, Checksum, Forward error correction, Data link layer protocols, HDLC, Point to Point Protocol, Sliding Window Protocol, Media Access control: Random Access, Controlled Access, Aloha Protocols, Carrier Sense, Multiple Access control, CSMA with collision detection, IEEE Standards, Channelization, Connecting devices and virtual LANs: Connecting Devices.

UNIT-III: Network Layer: 10 Hours

IP Addressing and Subnetting, Routing Fundamentals: Static vs. Dynamic Routing, Routing Protocols: RIP, OSPF, BGP, Address Resolution Protocol (ARP), The network layer in the Internet: IPV4 Addresses, IPV6, Internet Control protocol.

UNIT-IV: Transport Layer: 15 Hours

TCP vs. UDP: Comparison and Use Cases**,** TCP Connection Establishment and Termination (Three-Way Handshake)**,** Flow Control and Congestion Control**,** Error Handling and Reliability. Session and Transport interaction, Synchronization Points, Session Protocol Data Unit. Presentation Layer: Translation, Encryption/ Decryption, Authentication, Data compression.

UNIT-V Application Layer: 10 Hours

Network security, DES, RSA Algorithm, Common Protocols: HTTP, FTP, SMTP, DNS, Cryptography and Compression Techniques Network Applications and Services, Network Management protocols.

Text/Reference Books:

1. Data communications and networking, Behrouz A. Forouzan, Mc Graw Hill Education, 5th edition, 2012.
2. Computer Networks, Andrew S. Tanenbaum, Wetherall, Pearson, 5th edition, 2010.
3. Andrew S.Tanenbaum, ―Computer Networks, 4th Edition, Prentice Hall of India publishing Pvt. Ltd.
4. Kurose, Ross (2010), Computer Networking: A top down approach, Pearson Education, India.
5. Data and Computer Communications, Tenth Edition, William Stalling

*****

Name of Course and **Code**	**Course Title**	**L**	**T**	**P**	**Sessional**		**ESM**	**Total**
					MSE	**IA**		
Discipline Specific: Major 4- CSE MP-504	.NET Programming Laboratory	00	00	02	00	00	50	50

Course Description: To learn and comprehend the various C# statements. Recognize and apply various graphical tools for creating webpage. To learn Database connectivity using C# code to create a dynamic website. To become familiar with the fundamental development of applications tools.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Understand the bsics
- Learn the fundamentals of C# programming as well as the many graphics tools and how to apply them.
- Learn the fundamentals of Object-oriented programming concepts and their practical demonstration.
- Database connectivity with MS-Access, Sql Server and Oracle.

List of Programs:

1. Write a C# program based on various data types and type casting including boxing and unboxing.
2. Write a C# program based on conditional statements and loops.
    a. Write a C# program to print Fibonacci series without using recursion and using recursion.
    b. Write a C# program to check prime number.
    c. Write a C# program to print factorial of a number.
    d. Write a C# program to print sum of digits.
    e. Write a C# program to swap two numbers without using third variable.
    f. Write a C# program to convert decimal number to binary.
    g. Write a C# program to print number triangle.
    h. Write a C# program to convert number in characters.

i. Write a C# program to print various shapes.

3. Write a C# program based on array.
4. C# program to demonstrate the class and object creation using this function including call by value, call by reference and call by out.
5. Write C# program based on constructor and sealed class.
6. Write C# program to demonstrate the example of single inheritance.
7. Write C# to demonstrate the example of multilevel inheritance.
8. Write C# to demonstrate hierarchical inheritance, multilevel inheritance with method overriding.
9. Write C# to demonstrate the simple interface and structure.
10. Write C# to demonstrate method overloading based on the number of arguments.
11. Write C# to demonstrate method overloading based on types of arguments.
12. Write C# to demonstrate method overloading based on the order of arguments.
13. Demonstrate working with Forms, various controls and events.
14. Write C# program for database connectivity using ADO.NET.
15. Write C# program for window services.

Text/Reference Books:

1. Jeff Jeffrey Richter, "Applied Microsoft .Net Framework Programming", Microsoft Press.
2. Karli Watson ,Jacob Vibe Hammer, "Beginning Visual C# 2012 Programming", Wiley India.
3. Fergal Grimes, "Microsoft .Net for Programmers", SPD.
4. Christian Nagel, Jay Glynn, Morgan Skinner, "Professional C#", Wiley India.
5. Tony Baer, Jan D. Narkiewicz, Kent Tegels, "Understanding the .Net Framework", Wrox.

*****

Name of Course and **Code**	**Course Title**	**L**	**T**	**P**	**Sessional**		**ESM**	**Total**
					MSE	**IA**		
Minor 1 CSE MI-505	Computer System and Architecture	02	00	00	10	10	30	50

Course Description: The primary objective of the Computer Organization course is to provide students with a comprehensive understanding of the internal structure and functioning of computer systems. This includes learning about the design and operation of key components such as processors, memory, and input/output systems. The course aims to equip students with the foundational knowledge necessary to understand how hardware and software interact, and how computational tasks are executed at the machine level.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Understand and explain the architecture and organization of modern computer systems.
- Analyze the performance of various components of a computer system, including CPU, memory, and I/O devices.
- Design basic digital circuits and understand how they are used to build computer components.
- Apply knowledge of assembly language to write and debug programs that interact directly with hardware.
- Evaluate different computer architectures and make informed decisions about system design trade-offs.

UNIT-I Basics of Computer Organization: 10 Hours

Introduction to Computer Organization, Block Daigram of CPU, Define Instruction, Type of Instruction set, Instruction Set Organization, Instruction Set for Intel 8085 and other Microprocessors, Instruction Codes, Computer Registers for Instructions, Computer Instructions, Timing and Control, Instruction Cycle, Machine code and executable file, Memory-Reference Instructions, Input-Output, Interrupt. Type of Interrrupts (Hardware and Software Interrupt), System Call.

UNIT-II Programming the Basic Computer: 10 Hours

Machine Language, Assembly Language, Assembler, Program Loops, Subroutines, Input-Output Programming. **Microprogrammed Control:** Control Memory, Address Sequencing, Design of Control Unit.**Central Processing Unit:** General Register Organization, Stack Organization, Instruction Formats, Addressing Modes, RISC Computer, CISC Computer. Pipeline, Type of pipeline, Instrction pipeline, Arithmetic Pipeline, Design of Pipeline in Microprocessor.

UNIT-III: Input-Output Organization: 10 Hours

Peripheral Devices, Input-Output Interface, Asynchronous Data Transfer, Modes of Transfer, Priority Interrupt, DMA, Serial Communication. Characteristics of Multiprocessors, Interconnection Structures, Inter-processor Arbitration, Inter-processor Communication and Synchronization, Cache Coherence, Multicore Processors.

Text/Reference Books:

1. Patterson, D. A., & Hennessy, J. L.. Computer Organization and Design: The Hardware/Software Interface. Morgan Kaufmann.
2. Stallings, W.. Computer Organization and Architecture: Designing for Performance. Pearson.
3. Hamacher, V. C., Vranesic, Z. G., & Zaky, S. G.. Computer Organization and Embedded Systems. McGraw-Hill Education.
4. Null, L., & Lobur, J.. The Essentials of Computer Organization and Architecture. Jones & Bartlett Learning.
5. Harris, D. M., & Harris, S. L.. Digital Design and Computer Architecture. Morgan Kaufmann.

6. Tanenbaum, A. S., & Austin, T.. Structured Computer Organization. Pearson.
7. Mano, M. M.. Computer System Architecture. Pearson.

*****

Name of course and code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
(Minor 2) Vocational CSE MI-506	Software Engineering Laboratory	00	00	02	00	00	50	50

Course Description: Apply the Software Development Life Cycle (SDLC) to software projects. Conduct requirements analysis and develop use cases. Design software architecture using UML diagrams. Knowledge of project documentation.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Learn SA/SD analysis of various project
- Able to know how to develop software project documentation using various techniques based on software engineering

List of Programs:

- Perform testing mechanism on application and generation of test cases.
    1. Program for Configuration Management.
    2. Perform SA/SD for any software application as per decided by course coordinator.
    3. Design and development of test cases for testing.
    4. Perform Cost/Benefit Analysis.
    5. Identify scenarios & develop UML Use case and Class Diagram for the project.
    6. Draw DFD (upto 2 levels) and prepare Data Dictionary for the project.
    7. Develop Activity / State Transition diagram for the project.
    8. Develop Sequence and Collaboration diagram for the project.

Finally a Mini Project may be planned and carried out the semester to understand the importance of various concepts of Software Engineering and Project Management. The student may take up a Project in his/her intended area of specialization or in any other functional area of Computer Science. Ideally the Project should exhibit a cross -functional orientation.

The student shall submit the complete project and written structured report based on work done during this period. Project may be research project–based or may be an operational assignment involving working by the student on a given task / assignment / project /a real world problem. The report should be well documented and supported by –

- Summary

- Outline of the problem/task undertaken
- Research methodology & data analysis (in case of research projects)
- Relevant activity charts, tables, graphs, diagrams
- Learning of the student through the project

It should reflect the nature and quantum of work undertaken by the student. The completion of the Project shall be certified by the Faculty Guide.

Text/Reference Books:

1. Pankaj Jalote, "Software Engineering", Narosa Publications.
2. Rajib Mall, "Fundamental of Software Engineering", PHI.
3. R A Khan, A Agrawal, Software Engineering: A Practitioners Approach, Narosa Publication
4. Roger S. Pressman, "Software Engineering: A practitioner's Approach", McGraw Hill International Edition.
5. Sommerville, "Software Engineering", 7th edition, Pearson education.
6. K.K. Agarwal and Yogesh Singh, "Software Engineering", New Age International Publishers.
7. James F. Peters, Witold Pedrycz, "Software Engineering, an Engineering approach", John Wiley.
8. Shely Cashman Rosenblatt, "Systems Analysis and Design", Thomson Publications.
9. Waman S Jawadekar, "Software Engineering principles and practice", TMH

*****

Name of Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Internship CSE I-507	Internship	00	00	02	00	00	50	50

Internship: Students are responsible to choose Internship program online/offline to learn any specific field of knowledge and provide certificate from the agency/society/institute/center for the fulfillment of the course/part of work. Another option is also provided for those students how have not chosen such kind of program is responsible to complete the assigned task provided by Department under any faculty.

This work will be provided by concerned faculty member and student has to complete the work and submit report to Department and evaluation will be based on such documents and reports.

*****

BCA - Third Year, VI-Semester

Name of Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Discipline Specific: Major 1: CSE MT-601	Fundamentals of Cyber Security	03	01	00	20	20	60	100

Course Description: This course provides a foundational understanding of cybersecurity principles, threats, and mitigation strategies in today's digital landscape. Students will explore the evolving threat landscape, common cybercrimes, and legal frameworks. The course emphasizes practical skills in social media security, e-commerce protection, and securing digital devices. Through real-world case studies and hands-on exercises, students will develop the knowledge and skills to identify vulnerabilities, implement security measures, and respond effectively to cyber threats.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Learn the foundations of Cyber security and the threat landscape.
- To equip students with the technical knowledge and skills needed to protect and defend
- against cyber threats.
- To develop skills in students that can help them plan, implement, and monitor cyber
- Security mechanisms to ensure the protection of information technology assets.

UNIT-I Introduction to Cybersecurity: 12 Hours

Defining Cyberspace and Overview of Computer and Web-technology, Architecture of cyberspace, Communication and web technology, Internet, World wide web, Advent of internet, Internet infrastructure for data transfer and governance, Internet society, Regulation of cyberspace, Concept of cyber security, Issues and challenges of cyber security.

UNIT- II Cyber crime and Cyber law: 12 Hours

Classification of cyber crimes, Common cyber crimes- cyber crime targeting computers and mobiles, cyber crime against women and children, financial frauds, social engineering attacks, malware and ransomware attacks, zero day and zero click attacks, Cybercriminals modus-operandi , Reporting of cyber crimes, Remedial and mitigation measures, Legal perspective of cyber crime, IT Act 2000 and its amendments, Cyber crime and offences, Organisations dealing with Cyber crime and Cyber security in India, Case studies.

UNIT-III Social Media Overview and Security: 12 Hours

Introduction to Social networks. Types of Social media, Social media platforms, Social media monitoring, Hashtag, Viral content, Social media marketing, Social media privacy, Challenges, opportunities and pitfalls in online social network, Security issues related to social media, Flagging and reporting of inappropriate content, Laws regarding posting of inappropriate content, Best practices for the use of Social media, Case studies.

UNIT- IV E - Commerce and Digital Payments: 12 Hours

Definition of E- Commerce, Main components of E-Commerce, Elements of E-Commerce security, E-Commerce threats, E-Commerce security best practices, Introduction to digital payments, Components of digital payment and stake holders, Modes of digital payments- Banking Cards, Unified Payment Interface (UPI), e-Wallets, Unstructured Supplementary, Service Data (USSD), Aadhar enabled payments, Digital payments related common frauds and preventive measures. RBI guidelines on digital payments and customer protection in unauthorised banking transactions. Relevant provisions of Payament Settlement Act,2007.

UNIT- V Cyber Security Essentials: 12 Hours

Digital Devices Security , Tools and Technologies for Cyber Security End Point device and Mobile phone security, Password policy, Security patch management, Data backup, Downloading and management of third party software, Device security policy, Cyber Security best practices, Significance of host firewall and Ant-virus, Management of host firewall and Anti-virus, Wi-Fi security, Configuration of basic security policy and permissions.

Text/Reference Books:

1. Cyber Crime Impact in the New Millennium, by R. C Mishra , Auther Press. Edition 2010.
2. Cyber Security Understanding Cyber Crimes, Computer Forensics and Legal Perspectives by Sumit Belapure and Nina Godbole, Wiley India Pvt. Ltd. (First Edition, 2011)
3. Security in the Digital Age: Social Media Security Threats and Vulnerabilities by Henry A. Oliver, Create Space Independent Publishing Platform. (Pearson , 13th November, 2001)
4. Electronic Commerce by Elias M. Awad, Prentice Hall of India Pvt Ltd.
5. Cyber Laws: Intellectual Property & E-Commerce Security by Kumar K, Dominant Publishers.
6. Network Security Bible, Eric Cole, Ronald Krutz, James W. Conley, 2nd Edition, Wiley India Pvt. Ltd.
7. Fundamentals of Network Security by E. Maiwald, McGraw Hill

*****

Name of course and code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Discipline Specific: Major 2 CSE MT-602	Data Warehouse and Data Mining	03	01	00	20	20	60	100

Course Description: To understand data warehouse concepts, architecture, business analysis and tools. To understand data pre-processing and data visualization techniques. To study algorithms for finding hidden and interesting patterns in data. To understand and apply various classification and clustering techniques using tools.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Design a Data warehouse system and perform business analysis with OLAP tools.
- Apply suitable pre-processing and visualization techniques for data analysis
- Apply frequent pattern and association rule mining techniques for data analysis
- Apply appropriate classification and clustering techniques for data Analysis.

UNIT-I: Data Warehousing, Business Analysis and On-Line Analytical Processing (OLAP): 10 Hours

Basic Concepts, Data Warehousing Components, Building a Data Warehouse, Database Architectures for Parallel Processing, Parallel DBMS Vendors, Multidimensional Data Model, Data Warehouse Schemas for Decision Support, Concept Hierarchies -Characteristics of OLAP Systems, Typical OLAP Operations, OLAP and OLTP.

UNIT-II: Introduction to Data Mining Systems: 10 Hours

Knowledge Discovery Process, Data Mining Techniques, Issues and applications, Data Objects and attribute types, Statistical description of data, Data Preprocessing Cleaning, Integration, Reduction, Transformation and discretization, Data Visualization, Data similarity and dissimilarity measures.

UNIT-III: Frequent Pattern Analysis: 10 Hours

Mining Frequent Patterns, Associations and Correlations, Mining Methods, Pattern Evaluation Method, Pattern Mining in Multilevel, Multi Dimensional Space, Constraint Based Frequent Pattern Mining, Classification using Frequent Patterns

UNIT-IV: Classification And Clustering: 10 Hours

Decision Tree Induction, Bayesian Classification, Rule Based Classification, Classification by Back Propagation, Support Vector Machines, Lazy Learners, Model Evaluation and Selection-Techniques to improve Classification Accuracy. Clustering Techniques, Cluster Analysis-Partitioning Methods, Hierarchical Methods, Density Based Methods, Grid Based Methods, Evaluation of clustering, Clustering high dimensional data, Clustering with constraints, Outlier analysis outlier detection methods.

UNIT-V : Weka Tool: 10 Hours

Datasets, Introduction, Iris plants database, Breast cancer database, Auto imports database Introduction to WEKA, The Explorer, Getting started, Exploring the explorer, Learning

algorithms, Clustering algorithms, Association rule learners. Web Data Mining, Data Visualization, Applications of Data Mining

Text/Reference Books:

1. Jiawei Han and Micheline Kamber, ―Data Mining Concepts and Techniques, Third Edition, Elsevier, 2012.
2. Arun K. Pujari, "Data Mining Techniques" Universities Press
3. Humphries, Michael W. Hawkins, Michelle C. Dy, "Data Warehousing: Architecture and Implementation", Pearson
4. Alex Berson and Stephen J.Smith, ―Data Warehousing, Data Mining & OLAP‖, Tata McGraw – Hill Edition, 35th Reprint 2016.
5. K.P. Soman, Shyam Diwakar and V. Ajay, ―Insight into Data Mining Theory and Practice, Eastern Economy Edition, Prentice Hall of India, 2006.
6. Ian H.Witten and Eibe Frank, ―Data Mining: Practical Machine Learning Tools and Techniques, Elsevier, Second Edition.

*****

Name of course and code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Discipline Specific: Major 3 CSE MT-603	Artificial Intelligence	03	01	00	20	20	60	100

Course Description:

The Artificial Intelligence course provides an in-depth exploration of the foundational concepts, methodologies, and applications of AI. Students will begin by understanding the definition, scope, and historical development of AI, leading to an exploration of key AI techniques such as symbolic AI, machine learning, and neural networks. The course progresses into the crucial area of knowledge representation, covering logical reasoning, semantic networks, and ontologies, equipping students with the tools to represent and manipulate knowledge effectively. The design and implementation of expert systems form a core component, offering hands-on experience in building AI systems for real-world applications. The final unit focuses on AI planning and natural language processing (NLP), introducing students to the intricacies of automated planning, language models, and machine translation. This course prepares students to apply AI techniques in solving complex problems and designing intelligent systems in various domains.

Course Learning Outcomes: Here are five course learning outcomes for the provided "Introduction to Artificial Intelligence" syllabus, focusing on practical application and understanding:

- Analyze real-world problems and determine the suitability of applying AI techniques for their solution. Students will be able to assess problem characteristics, identify potential AI approaches, and justify their choices based on the strengths and limitations of each technique.
- Implement and evaluate fundamental AI algorithms for problem-solving, including search strategies, knowledge representation, and basic machine learning techniques. Students will gain hands-on experience with core AI algorithms, understanding their implementation details and analyzing their performance on different problem instances.
- Design and develop simple AI agents that can interact with environments, make decisions, and achieve specific goals. Students will apply their knowledge of agent architectures, planning techniques, and search algorithms to create intelligent agents capable of solving problems within defined environments.
- Explain the principles of natural language processing and demonstrate their application in tasks such as text analysis, language understanding, and machine translation. Students will gain a foundational understanding of NLP techniques, enabling them to analyze text, extract meaning, and explore applications like chatbot development or sentiment analysis.
- Evaluate the architecture and functionality of expert systems, understanding their strengths, limitations, and ethical considerations in various domains. Students will be able to analyze existing expert systems, identify their components, and critically assess their effectiveness, potential biases, and impact on decision-making processes.

UNIT-I: Introduction to Artificial Intelligence: 12 Hours

Definition and Scope of AI: Understanding what Artificial Intelligence is and its applications in various fields, History and Evolution of AI: Key milestones, from early developments to modern advancements; AI Techniques and Approaches: Symbolic AI, Machine Learning, and Neural Networks.; Agents and Environments: Concept of intelligent agents, types of agents, and the relationship between agents and their environments;

UNIT-II: Problem Solving and Search Strategie: 10 Hours

Problem Solving in AI: Introduction to statespace search, uninformed search strategies (e.g., BFS, DFS), and informed search strategies (e.g., A, Greedy search); Introduction to Machine Learning: Basics of supervised, unsupervised, and reinforcement learning. Problem Formulation, Uninformed Search Techniques- DFS, BFS, Hill Climbing, A*, Searching And-Or Graphs, Constrained Satisfaction Problems, Adversarial Search: Games, Minimax Algorithm, Alpha Beta pruning.

UNIT-III: Knowledge Representation: 10 Hours

Knowledge Representation in AI: Importance and types of knowledge (declarative, procedural); Logical Representation: Propositional logic, firstorder logic, and inference rules; Semantic Networks: Concepts, nodes, and relationships in semantic networks; Frames and Scripts: Structure of knowledge in frames, and the role of scripts in AI; Ontologies: Role of ontologies in

representing knowledge and structuring information; Reasoning Methods: Deductive reasoning, inductive reasoning, and abductive reasoning.

UNIT-IV: Planning and Natural Language Processing: 10 Hours

Introduction to AI Planning: Definition and types of planning (strategic, tactical, operational); Planning Algorithms: STRIPS, PartialOrder Planning, and Hierarchical Planning; Automated Planning and Scheduling: Use of AI in automating complex tasks and scheduling; Introduction to Natural Language Processing (NLP): Overview of NLP, applications, and challenges; Language Models: Ngrams, hidden Markov models, and neural language models; Syntax and Semantics in NLP: Parsing techniques, semantic analysis, and disambiguation; Speech Recognition and Generation: Fundamentals of speech recognition systems and natural language generation; Machine Translation: Techniques for translating text from one language to another using AI.

UNIT-V: Expert System Design: 10 Hours

Introduction to Expert Systems: Definition, components, and applications of expert systems; Architecture of Expert Systems: Knowledge base, inference engine, and user interface; Knowledge Acquisition: Methods for gathering and encoding expert knowledge; Inference Mechanisms: Forward chaining, backward chaining, and hybrid approaches; Case Studies of Expert Systems: Examples from various domains like medical diagnosis, finance, and engineering; Design and Development: Steps involved in designing an expert system, from problem identification to deployment; Challenges and Limitations: Common challenges in developing and maintaining expert systems.

Text/Reference Books:

1. Elain Rich and Kevin Knight. *Artificial Intelligence*, 3rd Edition.
2. Thareja, R.. Artificial Intelligence: *Beyond Classical AI.* Pearson.
3. Nils J. Nilsson. *Artificial Intelligence: A New Synthesis*.
4. Peter Flach. *Machine Learning: The Art and Science of Algorithms that Make Sense of Data*.
5. S. Russel and P. Norvig, "Artificial Intelligence – A Modern Approach", Second Edition, Pearson Education

*****

Name of course and code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Discipline Specific: Major 4 CSE MP-604	Project Work	00	00	04	00	50	50	100

Dissertation/Research Work/Project: Dissertation/research work/project copied from other students will be considered to have used unfair means. If two dissertations are found identical by

more than 40%, then zero marks will be awarded to both of them. In such a case the dissertation will have to be resubmitted on the new topic.

*****

Name of course and code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Minor 1 CSE MI-605	Basics to Quantum Computing	02	00	00	10	10	30	50

Course Description: Quantum computing is fast emerging as one the key disruptive technologies of our times. It is a fundamentally new computing paradigm that has the potential to efficiently solve certain challenging problems which cannot be solved efficiently in a classical setting. IBM has made significant investment in this technology and is recognized as a leader in this space. This course will provide introduction to Quantum Computation, starting with basic concepts such as superposition and entanglement, to discussing the quantum circuit model of computation and basic Quantum algorithms that demonstrate the power of computing with quantum bits. We will also introduce the idea of quantum error correction to mitigate the effects of noise in today's quantum devices. We will have full hands-on sessions for each concept taught using Qiskit, a pythonic way of programming and the IBM Circuit Composer.

UNIT- I: Introduction to Essential Linear Algebra and Quantum Computation: **10 Hours**
Introduction to Essential Linear Algebra, Complex Numbers, Basic Physics for Quantum Computing: The Journey to Quantum, Quantum Physics Essentials, Basic Atomic Structure, Hilbert Spaces, Uncertainty, Quantum States, Entanglement: Data Structures, Algorithms, Computational Complexity, Coding Theory

UNIT- II: State of a Quantum System, Quantum Circuit: **10 Hours**
Logic Gates, Computer Architecture, Fundamental Computer Science for Quantum Computing, Quantum bits, Bloch sphere representation of a qubit, Multiple qubits, Hilber Space, Probabilities and measurements, Entanglement, Density operators and correlation, Basics of quantum mechanics, Design of quantum circuits.

UNIT-III: Quantum Cryptography: **10 Hours**
Introduction to Quantum Key Distribution: The principles of secure communication using quantum properties, The BB84 protocol: a prominent example of QKD, Security analysis of QKD against eavesdropping.

Text/Reference Books:

1. Phillip Kaye, Raymond Laflamme et. al., An introduction to Quantum Computing, Oxford University press, 2007.

2. Chris Bernhardt, Quantum Computing for Everyone, The MIT Press,Cambridge, 2020
3. David McMahon-Quantum Computing Explained-Wiley-Interscience , IEEE Computer Society (2008) Quantum Computation and Quantum Information, M. A. Nielsen &I.Chuang, Cambridge University Press (2013).
4. Quantum Computing A Gentle Introduction , Eleanor G. Rieffel and Wolfgang H. Polak MIT press (2014)

*****

Name of Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
(Minor 2) Vocational: CSE MI-606	Artificial Intelligence Laboratory	00	00	02	00	00	50	50

Course Description: This AI lab introduces fundamental concepts of logic programming using Prolog. Students will gain hands-on experience implementing various algorithms and problem-solving techniques.

Software Required: SWI-Prolog (or any compatible Prolog interpreter)

List of Experiments:

1. Factorial of a Number: Write a Prolog program to calculate the factorial of a given number.
2. Maximum of Two Numbers: Write a Prolog program to find the maximum of two given numbers.
3. Even or Odd Number: Write a Prolog program to check if a given number is even or odd.
4. Addition and Subtraction: Write a Prolog program to perform addition or subtraction of two numbers.
5. Gender Identification: Write a Prolog program to check the gender of a name from a given list.
6. Sentence Verification: Write a Prolog program to verify the grammatical correctness of a simple sentence.
7. Family Relationship Tree: Write a Prolog program to represent and query a family relationship tree.
8. Traveling Salesman Problem: Implement a Prolog program to solve the Traveling Salesman Problem using a suitable algorithm.
9. Circle Calculations: Write a Prolog program to calculate the area and circumference of a circle given its radius.

10. Greatest Common Divisor: Write a Prolog program to find the GCD of two numbers.
11. List Length: Write a Prolog program to find the length of a given list.
12. Leap Year Check: Write a Prolog program to determine if a given year is a leap year.
13. Reverse a List: Write a Prolog program to print the reverse of a given list.
14. Add Element to List: Write a Prolog program to add an element to the beginning or end of a given list.

*****

BCA - Forth Year, VII-Semester

Name of course and code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Discipline Specific: Major 1 CSE MT-701	Advanced Programming	03	01	00	20	20	60	100

Course Description:: To enable the students to learn about advanced features of C, C++ & UNIX programming.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Design an algorithmic solution for a given problem and translate it into a program.
- Use the appropriate control statements to solve the given problem.
- Implement different Operations on arrays and use functions to solve the given problem.
- Understand pointers, structures and unions
- Understanding of ADT with C++ classes
- Proper knowledge of OOP concepts in C++
- Basic knowledge of Unix and Shell programming

UNIT-I: C Language Fundamentals: 15 Hours

Algorithm and its characteristics, flowchart, Algorithm involving Decisions and Loops, Developing Algorithms and Sketching Flowcharts for various problems, Pseudo code, top down & bottom-up approaches of program design. Introduction and Features of "C" language, Structure of "C" program, Identifiers and Keywords, Constants, Variables, Scope of variables, Typedef, Type Conversion, Arithmetic Operators, Statements, Expressions, Operators, Precedence of operators, Library Functions, Input/Output Statements, Compound statements and block, Control structures, Decision making and Branching, Decision making & looping.

UNIT-II: Function and Structured Data Types in C: 10 Hours

Need of "C" function, User Defined and Library Functions, Prototype of Function, Call by Value, Nesting of Functions, Recursion, Array, Multidimensional Array declaration and their

applications, Array as Function Argument, String, Basic functions dedicated to String Manipulation, Pointer variable and its importance, Pointer Arithmetic, passing parameters by reference, pointer to pointer, Declaration and Defining Structure, Accessing Structure members, nesting of structures, Array of Structures, Structure as Function Argument, declaration of unions, pointer to structure & unions.

UNIT-III: Principal of Object-Oriented Programming: 10 Hours

Procedural Vs. Object-Oriented Programming, OOPS paradigms, basic concept of OOP: abstraction, encapsulation, modularity, data hierarchy through inheritance, polymorphism and typing, Applications and Benefits of OOP, Basic components of a C++ Program and program structure, Compiling and Executing C++ Program, Build and execute a C program in C++, Write equivalent programs in C++, Constructors and Their Types, Destructors, Inline Functions, Passing Objects as Function Parameters

UNIT-IV: OOP Concepts in C++: 10 Hours

Friend Functions, Overloading of Functions, Friend class, Dynamic allocation operator new and delete, Types of polymorphism, Constructor overloading, Operator overloading, Template, Types of inheritance, Protected derivation of class, resolving ambiguity, Pointer to object, this pointer, Virtual class, virtual function, Implementing Polymorphism in C++, Need of Exception handling, try, catch and throws keywords, defining namespace, benefit of namespace.

UNIT-V: UNIX Concepts & Shell Programming: 15 Hours

Introduction to UNIX, UNIX System Organization (The Kernel and the Shell), Files and Directories, Library Functions and System Calls, Editors (vi and ed), UNIX Shell Programming: Types of Shells, Shell Meta Characters, Shell Variables, Shell Scripts, Shell Commands, UNIX Environment, UNIX System Administration: File System, Mounting and Unmounting File System, System Booting, Shutting Down, Handling User Account, Backup, Recovery, Security.

Text/Reference Books:

1. E. Balaguruswamy, Programming in ANSI C 5th Edition McGraw-Hill
2. Yashvant P. Kanetkar, Let Us C", BPB Publications.
3. Programming: Principles and Practice Using C++ by Bjarne Stroustrup
4. R. Venugopal, Rajkumar, and T. Ravishanker "Mastering C++", TMH, 1997.
5. The C Programming Language by Brian W. Kernighan and Dennis M. Ritchie.
6. Encyclopedia C by Robert A. Radcliffe, BPB Publications.
7. Object Oriented Programming in C++ by Robert Lafore, Techmedia Publication.
8. Object Oriented Programming in C++, R Rajaram, New Age International Publishers.
9. Parata, "Advanced UNIX Programming guide", BPB, 1985, India, New Delhi.
10. Yashwant Kanitkar, "UNIX Shell Programming", BPB, 2003, India, New Delhi.
11. Vijay Mukhi,"The 'C' Odyssey UNIX-the open boundless C", BPB, 2004, India, New Delhi.

12. Sumitabh Das, "UNIX Concepts and applications", TMH, 2017, USA, New York.

*****

Name of Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Discipline Specific: Major 2 CSE MT-702	Machine Learning and its application	03	01	00	20	20	60	100

Course Description: This course provides a comprehensive introduction to the principles and practices of data science, equipping students with the knowledge and skills to extract meaningful insights from data. Students will explore the entire data science lifecycle, from data collection and preparation to exploratory data analysis, data visualization, and the application of statistical methods. The course also covers essential database technologies for data science, including SQL and NoSQL databases, enabling students to effectively manage and query structured and unstructured data.

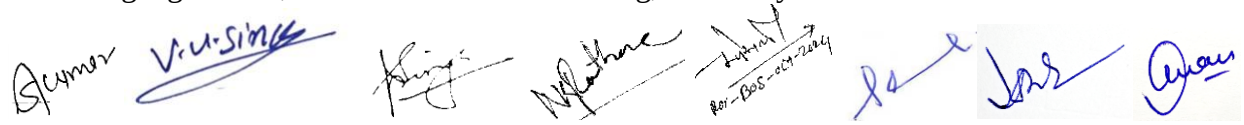
Course Learning Outcomes:

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

- Articulate the importance of data science and its role in various domains.
- Understand the data science process and the key steps involved in a typical data science project.
- Perform exploratory data analysis using statistical measures, data visualization techniques, and appropriate tools.
- Write SQL queries to retrieve, filter, and aggregate data from relational databases.
- Work with NoSQL databases, including document databases like MongoDB, to handle unstructured data.
- Apply data science methodologies to solve real-world problems and communicate findings effectively.

UNIT-I: Introduction to Machine Learning: 10 Hours

Introduction, What is Human Learning, Types of Human Learning, Learning under expert guidance, Learning guided by knowledge gained from experts, Learning by self, What is Machine Learning?, How do machines learn?, Well-posed learning problem, Types of Machine Learning, Supervised learning, Unsupervised learning, Reinforcement learning, Comparison – supervised, unsupervised, and reinforcement learning, Problems Not To Be Solved Using Machine Learning, Applications of Machine Learning, Banking and finance, Insurance, Healthcare, State-of-The-Art Languages/Tools In Machine Learning, Python, R, Matlab, SAS, Other languages/tools, Issues in Machine Learning, Summary



UNIT-II: Preparing to Model: **[1]10 Hours**

Introduction, Machine Learning Activities, Basic Types of Data in Machine Learning, Exploring Structure of Data, Exploring numerical data, Plotting and exploring numerical data, Exploring categorical data, Exploring relationship between variables, Data Quality and Remediation, Data quality, Data remediation, Data Pre-Processing, Dimensionality reduction, Feature subset selection, Summary

UNIT-III: Modeling and Evaluation: **10 Hours**

Introduction, Selecting a Model, Predictive models, Descriptive models, Training a Model (for Supervised Learning), Holdout method, K-fold Cross-validation method, Bootstrap sampling, Lazy vs. Eager learner, Model Representation and Interpretability, Underfitting, Overfitting, Bias – variance trade-off, Evaluating Performance of a Model, Supervised learning – classification, Supervised learning – regression, Unsupervised learning – clustering, Improving Performance of a Model, Summary

UNIT-IV: Supervised Learning-Classification: **15 Hours**

Introduction, Example of Supervised Learning, Classification Model, Classification Learning Steps, Common Classification Algorithms, k-Nearest Neighbour (kNN), Decision tree, Random Forest model, Support vector machines, Summary, Supervised Learning- Regression**:** Introduction, Example of Regression, Common Regression Algorithms, Simple linear regression, Multiple linear regression, Assumptions in Regression Analysis, Main Problems in Regression Analysis, Improving Accuracy of the Linear Regression Model, Polynomial Regression Model, Logistic Regression, Maximum Likelihood Estimation, Summary

UNIT-V : Unsupervised Learning: **15 Hours**

Introduction, Unsupervised vs Supervised Learning, Application of Unsupervised Learning, Clustering, Clustering as a machine learning task, Different types of clustering techniques, Partitioning methods, K-Medoids: a representative object-based technique, Hierarchical clustering, Density-based methods – DBSCAN, Finding Pattern using Association Rule, Definition of common terms, Association rule, The apriori algorithm for association rule learning, Build the apriori principle rules, Summary

Text/ Reference Books:

1. Tom M. Mitchell- Machine Learning- McGraw Hill Education, International Edition
2. Bishop, Christopher. Neural Networks for Pattern Recognition. New York, NY: Oxford University Press
3. Ethem Alpaydin, “Introduction to Machine Learning (Adaptive Computation and Machine Learning)”, The MIT Press

1

4. Machine Learning by Subramanian Chandramouli, Saikat Dutt, Amit Kumar Das, 2018, Pearson Education India
5. Christopher M. Bishop Pattern Recognition and Machine Learning- Springer, 2nd edition

*****

Name of course and code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Discipline Specific: Major 3 CSE MT-703	Data Structures and Algorithms	03	01	00	20	20	60	100

Course Description: To learn the advanced concepts of data structure and algorithms and its implementation. The course has the main ingredients required for a computer science graduate and has all the necessary topics for assessment of advanced data structures and algorithms.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Basic ability to analyze algorithms and to determine algorithm correctness and time efficiency class.
- Master a variety of advanced abstract data type (ADT) and data structures and their implementations.
- Ability to apply and implement learned algorithm design techniques and data structures to solve problems.

UNIT-I: Fundamental of Algorithm: 15 Hours

Algorithms, Performance analysis- time complexity and space complexity, Asymptotic Notation- Big Oh, Omega and Theta notations, Complexity Analysis Examples. Data structures-Linear and non linear data structures, ADT concept, Linear List ADT, Array representation, Linked representation, singly linked lists -insertion, deletion, search operations, doubly linked lists- insertion, deletion operations, circular lists. Representation of single, two dimensional arrays, Sparse matrices and their representation.

UNIT-II: Stack and Queue concepts 10 Hours

Stack and queue ADTs, array and linked list representations, infix to postfix conversion using stack, implementation of recursion, Circular queue-insertion and deletion, Dequeue ADT, array and linked list representations, Priority queue ADT, implementation using Heaps, Insertion into a Max Heap, Deletion from a Max Heap.

UNIT-III: Searching and Sorting 10 Hours

Searching -Linear and binary search methods, Hashing-Hash functions, Collision Resolution methods-Open Addressing, Chaining, Rehashing, double hashing. Sorting –Bubble sort, Insertion sort, Quick sort, Merge sort, Heap sort, Radix sort, comparison of sorting methods.

UNIT-IV: Trees: 15 Hours

Ordinary and Binary trees terminology, Properties of Binary trees, Binary tree ADT, representations, traversals, Threaded binary trees. Graphs- Graphs terminology, Graph ADT, representations, graph traversals/search methods-DFS and BFS, Applications of Graphs- Minimum cost spanning tree using Kruskal's and Prim's algorithm, Shortest-path Algorithms: Dijkstra's and Floyd's algorithm, Topological sort.

UNIT-V: Binary and Balanced Trees: 10 Hours

Binary search tree-Binary search tree ADT, insertion, deletion and searching operations, Balanced search trees, AVL trees-Definition and examples only, Red Black trees – Definition and examples only, B-Trees-definition, insertion and searching operations.

Text/Reference Books:

1. Thomas Coremen, "Introduction to Algorithms", Third edition, Prentice Hall of India (2009).
2. Kleinberg J., Tardos E., "Algorithm Design", 1st Edition, Pearson, 2012.
3. Motwani R., Raghavan P., "Randomized Algorithms", Cambridge University Press, 1995.
4. Vazirani, Vijay V., "Approximation Algorithms", Springer, 2001.
5. Fundamentals of Data structures in C, 2nd Edition, E.Horowitz, S.Sahni and Susan AndersonFreed, Universities Press.
6. Data structures A Programming Approach with C, D.S.Kushwaha and A.K.Misra, PHI.

*****

Name of Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Discipline Specific: Major 4 CSE MP-704	Machine Learning Using Python Laboratory	00	00	02	00	00	50	50

List of Programs:

1. Familiarizing with Anaconda and Jupyter for importing modules and dependencies for ML
2. Familiarization with NumPy, Panda and Matplotlib by Loading Dataset in Python.
3. Implement and demonstrate the FIND-S algorithm for finding the most specific hypothesis based on a given set of training data samples. Read the training data from a .CSV file.

4. For a given set of training data examples stored in a .CSV file, implement and demonstrate the Candidate-Elimination algorithm to output a description of the set of all hypotheses consistent with the training examples.
5. Write a program to demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.
6. Build an Artificial Neural Network by implementing the Backpropagation algorithm and test the same using appropriate data sets.
7. Write a program to implement the naïve Bayesian classifier for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering few test data sets.
8. Assuming a set of documents that need to be classified, use the naïve Bayesian Classifier model to perform this task.
9. Built-in python classes/API can be used to write the program. Calculate the accuracy, precision, and recall for your data set.
10. Write a program to construct a Bayesian network considering medical data. Use this model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set. You can use Python ML library classes/API.
11. Apply EM algorithm to cluster a set of data stored in a .CSV file. Use the same data set for clustering using k-Means algorithm. Compare the results of these two algorithms and comment on the quality of clustering. You can add Python ML library classes/API in the program.

*****

Name of course and code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Discipline Specific: Major 5 CSE MP-705	Data Structures and Algorithms Laboratory	00	00	02	00	00	50	50

Course Description: To provide the knowledge of basic data structures and their implementations. To understand importance of data structures in context of writing efficient programs. Students can develop skills to apply appropriate data structures in problem solving.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Choose a suitable Data Structures for an application
- Develop ability to implement different Sorting and Search methods
- Have knowledge onData Structures basic operations like insert, delete, search, update and traversal
- Design and develop programs using various data structures

List of Programs

1. Write a program to initialize the element into array and display.
2. Write a program to swap the first element with last, second to second last and so on (reversing elements) by using an array.
3. Write a program to display the sum of all the elements of the array.
4. Write a program to search an element into an array by using Linear Search.
5. Write a program to search an element into an array by using Binary Search.
6. Write a program to implement bubble sort.
7. Write a program to implement selection sort.
8. Write a program to implement quick sort.
9. Write a program to implement insertion sort.
10. Write a program to search an element in the 2-dimensional array by using linear search.
11. Write a program to merge two sorted arrays into one sorted array.
12. Write a program to perform the following operation in matrix.
    a. Addition b. Subtraction c. Transpose
13. Write a program to find multiplication of two matrices.
14. Write a program to print following matrix
    a. Upper diagonal b. Lower diagonal
15. Write a program to print sum of diagonal elements.
16. Write a program to implement stack using array to perform:
    a. PUSH b. POP c. Display
17. Write a program to convert infix to postfix by using stack.
18. Write a program to implement factorial by using a recursion function.
19. Write a program to print reverses of a given number by using stack.
20. Write a program to implement a queue using an array.
    a. Insert b. Delete c. Display
21. Write a program to implement a circular queue using an array.
22. Write a program to implement link list with following operations:
    a. Insert b. Display c. Delete
23. Write a program to implement stack using a link list.
24. Write a program to implement a queue using a link list.
25. Write a program to implement a circular queue using a link list.
26. Write a program to concatenate two link lists.
27. Write a program to add two polynomials with the help of a link list.
28. Write a program to implement a circular queue using a link list.
29. Write a program to reverse the link list.
30. Write a program to implement the following operations on Binary Search Tree
    a. Insert b. Delete c. Search d. Display

Text/Reference Books:

1. "Data Structures using C", ISRD group Second Edition, TMH
2. "Data Structures through C", Yashavant Kanetkar, BPB Publications

3. "Data Structures Using C" Balagurusamy E. TMH
4. Fundamentals of Data structures in C, 2nd Edition, E.Horowitz, S.Sahni and Susan AndersonFreed, Universities Press.
5. Data structures A Programming Approach with C, D.S.Kushwaha and A.K.Misra, PHI.

*****

Name of Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Minor 1 CSE MI - 706	Fundamentals of Big Data Analytics	02	00	00	10	10	30	50

Course Description: This course provides a comprehensive introduction to the world of big data, equipping students with the knowledge and skills to tackle the challenges and harness the power of massive datasets. Students will delve into the core concepts of big data, explore distributed file systems and the Hadoop ecosystem, and gain hands-on experience with essential big data technologies. The course emphasizes practical skills in processing, analyzing, and interpreting large datasets, covering the entire data analytics process and highlighting the essential skills for success in this rapidly evolving field.

Course Learning Outcomes:
Upon successful completion of this course, students will be able to:

- Understand the characteristics, importance, and applications of big data.
- Utilize Hadoop and its ecosystem tools for big data processing and analysis.
- Apply data analytics techniques to extract meaningful insights from large datasets.

UNIT- I: **Introduction to Big Data and Big Data Analytics:** **07 Hours**
Introduction, Big Data, Distributed file system, Big Data and its importance, Four Vs Drivers for Big data, Big data analytics, Big data applications, Algorithms using map reduce.

UNIT- II : **Hadoop and applied concepts:** **08 Hours**
Introduction, Hadoop, Definition of Hadoop, Big Data, Apache Hadoop, Hadoop Eco System, Moving Data in and out of Hadoop, Map Reduce, Understanding Inputs and Outputs of MapReduce, Data Serialization, Architecture of Hadoop. Introduction to Hdfs, Hive And Hiveql, Hbase and Spark.

UNIT-III: Mastering Big Data Technologies: **15 Hours**
HDFS, Hive, HBase, and ZooKeeper: Introduction, Overview of HDFS-Installation, Shell, Java API, Hive-Architecture-Installation, Comparison with Traditional Database, HiveQL-Querying Data, Sorting, Aggregating, Map Reduce Scripts, Joins, Sub queries, Concepts of HBase,

Advanced Usage, Schema Design, Advance Indexing, PIG5- Zookeeper, How it helps in monitoring a cluster? Uses of HBase in Zookeeper, How to Build Applications with Zookeeper?, Distinguish between HDFS and HBase.

Text/Reference Books:

1. Hadoop: The Definitive Guide by Tom White
2. Big Data: Principles and Best Practices of Scalable Realtime Data Systems by Nathan Marz and James Warren
3. HBase: The Definitive Guide by Lars George
4. Learning Spark: Lightning-Fast Big Data Analysis by Holden Karau, Andy Konwinski, Patrick Wendell, and Matei Zaharia
5. Michael Berthold, David J. Hand, "Intelligent Data Analysis", Springer, 2007

*****

Name of Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Minor 2: Vocational CSE MI - 707	Introduction to Data Science	02	00	00	10	10	30	50

Course Description: This course provides a comprehensive introduction to the principles and practices of data science, equipping students with the knowledge and skills to extract meaningful insights from data. Students will explore the entire data science lifecycle, from data collection and preparation to exploratory data analysis, data visualization, and the application of statistical methods. The course also covers essential database technologies for data science, including SQL and NoSQL databases, enabling students to effectively manage and query structured and unstructured data.

Course Learning Outcomes:

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

- Articulate the importance of data science and its role in various domains.
- Understand the data science process and the key steps involved in a typical data science project.
- Perform exploratory data analysis using statistical measures, data visualization techniques, and appropriate tools.
- Write SQL queries to retrieve, filter, and aggregate data from relational databases.
- Work with NoSQL databases, including document databases like MongoDB, to handle unstructured data.
- Apply data science methodologies to solve real-world problems and communicate findings effectively.

UNIT-I: INTRODUCTION: **10 Hours**

Need for Data Science – Data Science Process – Business Intelligence and Data Science – Prerequisites for a Data Scientist. Exploratory Data Analysis - Statistical measures, Basic tools (plots, graphs, and summary statistics) of EDA, Data Analytics Life-cycle, Preparing Data, Data Visualization, Uni-variant, Multi variant Analysis

UNIT-II: DATABASES FOR DATA SCIENCE: **10 Hours**

Structured Query Language (SQL): Data Munging, Filtering, Joins, Aggregation, Window Functions, Ordered Data, No-SQL, Document Databases, Wide-column Databases and Graphical Databases. Unstructured data: MongoDB, JSON.

UNIT-III: DATA SCIENCE METHODOLOGY: **10 Hours**

Analytics for Data Science – Examples of Data Analytics – Data Analytics Lifecycle: Data Discovery, Data Preparation, Model Planning, Model Building.

Text/ Reference Books:

1. Avrim Blum, John Hopcroft, Ravindran Kannan, "Foundations of Data Science", Cambridge University Press, 1st Edition, 2020.
2. Sanjeev Wagh, Manisha Bhende, Anuradha Thakare, 'Fundamentals of Data Science, CRC Press, 1st Edition, 2022.
3. Joel Grus, "Data Science from Scratch: First Principles with Python", O'Reilly Media, 1st Edition, 2015.
4. Daimi, Kevin, Ed. Hamid R. Arabnia, Principles of Data Science, Springer, 2020.
5. Vijay Kotu, Bala Deshpande, Data Science: Concepts and Practices, Morgan Kaufmann Publishers, Second edition, 2019

*****

Four-Year UG Degree (Honors)

BCA - Fourth Year, VIII-Semester- (Honors)

Name of course and code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Discipline Specific: Major 1 CSE MT-801	Fundamentals of IoT and Cloud Computing	03	01	00	20	20	60	100

Course Description: This course provides a comprehensive introduction to the interconnected worlds of the Internet of Things and Cloud Computing. Students will gain a deep understanding

of the fundamental concepts, architectures, technologies, and applications driving these transformative fields. The course explores the interplay between IoT devices, data acquisition, cloud-based processing, and the development of innovative solutions across various domains.

Course Learning Outcomes:

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

- Understand the fundamental concepts, characteristics, and architectures of the Internet of Things.
- Identify and analyze various communication models and technologies used in IoT systems.
- Explore real-world applications of IoT in domains such as energy, agriculture, healthcare, manufacturing, and smart cities.
- Explain the evolution, characteristics, benefits, and challenges of Cloud Computing.
- Differentiate between various cloud deployment models (public, private, community, hybrid) and service models.
- Describe the architecture of cloud computing systems, including virtualization techniques and their role in resource management.

UNIT-I : Introduction to Internet of Things: 10 Hours

Internet of Things Concepts, Characteristics of IoT ,Physical and Logical design of IoT, Functional blocks of IoT, Communication models, Smart Object and Smart Environments

UNIT-II Technologies and Application: 10 Hours

Domains of IoT, Internet of Things Framework, Communication technology infrastructure, Architecture and reference models, Application Domains: Energy, Agriculture, Health Care, Manufacturing, Smart cities.

UNIT- III: Introduction, Traditional Computing: 10 Hours

History, and Evolution of Cloud Computing, Cluster Vs. Grid Vs. Cloud Computing, Characteristics of Cloud Computing, Benefits of Cloud Computing, Cloud Computing Applications, and Challenges of Cloud Computing. Overview of Edge Computing and Its Role in IoT and Batch Processing, Privacy Concerns in IoT and GDPR Compliance.

UNIT-IV: **Cloud deployment models, and architecture:** **15 Hours**

Introduction, Cloud Deployment Models-Public Cloud, Private Cloud, Community Cloud, and Hybrid Cloud; Choosing the Appropriate Deployment Model-Suitability of Public Cloud, Suitability of Private Cloud, Suitability of Community Cloud, Suitability of Hybrid Cloud, Comparative analysis of cloud deployment models, Service Delivery Models, Infrastructure As a Service (IaaS), Platform As a Service(PaaS), Software As a Service (SaaS), Other Services (Security Management, Identity Management, Storage, Database, Back-up, etc.), Cloud Architecture, Layers and Anatomy of the Cloud and Network Connectivity in Cloud computing.

UNIT-V: **Resource Virtualization:** **15 Hours**

What is virtualization, virtualizing physical computing resources, understanding abstraction, business benefits of virtualization, machine or server level virtualization, exploring hypervisor or virtual machine monitor, exploring hypervisor or virtual machine monitor, operating system level virtualization: removal of the hypervisor, major server virtualization products and vendors, some other types of virtualizations, advantages of virtualization, virtualization security recommendations, Cloud Deployment Models: Public, Private, Hybrid, and Community Clouds.

Text / Reference Books :

1. CathyO'Neil,RachelSchutt, Doing DataScience, Straight Talkfrom The Frontline. O'Reilly, Internet of Things (A Hands-on-Approach) , Vijay Madisetti , Arshdeep Bahga, University Press, First Edition, 2013.
2. Avesand,S.,Karnouskos,S.,Boyle,D.,Mulligan,C.,Tsiatsis,V.,Holler,J. Internet of Things. Netherlands: Elsevier Science, 2014.
3. Irizarry, Rafael A. Introduction to Data Science: Data Analysis and Prediction Algorithms with R. United States: CRC Press, 2019.
4. Jan Holler, Vlasios Tsiatsis, Catherine Mulligan, Stefan Avesand, Stamatis Karnouskos, David Boyle, "FromMachine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence", 1st Edition, AcademicPress, 2014.
5. Cloud Computing : A Practical Approach by Anthony T. Velte Toby J. Velte, Robert Elsenpeter, by TheMcGraw-Hill, 2010.

*****

Name of course and code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Discipline Specific: Major 2 CSE MT-802	Advanced Java Programming	03	01	00	20	20	60	100

Course Description: The objective of the Advanced Java Programming course is to deepen students' understanding of Java programming by exploring advanced concepts such as multithreading, networking, database connectivity, and web technologies. The course aims to equip students with the skills needed to develop complex, scalable, and high-performance Java applications.

Course Learning Outcomes:

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

- Implement multithreading and concurrency mechanisms in Java applications.
- Develop and integrate Java-based network applications using sockets and RMI.
- Utilize JDBC for database connectivity and perform CRUD operations.
- Design and develop web applications using Java servlets, JSP, and frameworks.
- Apply best practices in Java programming to create efficient and maintainable code.

UNIT-I J2EE and Swing Overview: 12 Hours

Introduction to J2EE, Difference among J2SE, J2EE, and J2ME

Swing: The Origins of Swing-Swing Features-The MVC Connection-Components and Containers-The Swingpackages-JLabel-JTextfield-TheSwingButtons-JButton-JToggleButton-CheckBoxes-RadioButtons-Jlist-Jmenu-JcomboBox-JTable-JScrollPane-JTabbedPane.

UNIT-II Comprehensive JDBC Guide: 12 Hours

Interacting with Database: Introduction to JDBC, Essential JDBC classes, connecting to database, inserting data in database, Retrieving data from database, deleting data in database, updating data in database, store image in the database, to retrieve image from database, to store file in database, retrieve file from database. Database Programming using JDBC, JDBC Drivers & Architecture, CURD operation Using JDBC, Connecting to non-conventional Databases.

UNIT-III Java Server Pages: 12 Hours

Introduction, Architecture of JSP, Life Cycle of JSP, Scripting elements (Scriplets, JSP Declarations, JSP Expression), Directive Elements (page, include, taglib), JSP Actions (include, setproperty, getproperty, forward, text), Implicit objects (request, response, out, page, Exception), including HTML in JSP

UNIT-IV Java Server Technologies: 12 Hours

Servlet Web Application Basics, Architecture and challenges of Web Application, Introduction to servlet, Servlet life cycle, Developing and Deploying Servlets, Exploring Deployment, Descriptor (web.xml), Handling Request and Response.

UNIT-V: Java Beans: 12 Hours

Introduction to Java Beans-Advantages of Java Beans-Using Bound and Constraint Properties-Persistence-Java Beans API-A Bean Example

Struts2: Introduction toStruts2, Struts2Architecture, Actions, Interceptors and Type Conversion, Value Stack/OGNL, UI Component Tags and Results, Validation Framework, Database Access

Text/Reference Books:

1. Patrick Naughton, “COMPLETE REFERENCE: JAVA2”, Tata McGraw-Hill, 2003

2. Schildt Herbert,The Complete Reference Java Seventh Edition 2011
3. Core Servlets and Java Server pages volume1: Core Technologies By Marty Hall and Pearson.
4. Java 6 Programming, Black Book, Dreamtech
5. Java Server Programming, Java EE6 (J2EE 1.6), Black Book, Dreamtech
6. Advanced Java Technology, By M.T. Savaliya, Dreamtech

*****

Name of Course and **Code**	**Course Title**	**L**	**T**	**P**	**Sessional**		**ESM**	**Total**
					MSE	**IA**		
Discipline Specific: Major 3 CSE MT-803	Advanced Database Management System	03	01	00	20	20	60	100

Course Description:
To develop skills in database design, implementation, and management. To familiarize students with SQL and its practical applications. Construct simple and moderately advanced database queries using, Structured Query Language (SQL). Understand and successfully apply logical database design principles, including E-R diagrams and database normalization.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Understand the concepts and principles of database management systems.
- Design and create relational databases using SQL.
- Query and manipulate data using SQL commands.
- Apply normalization techniques to ensure data integrity.

UNIT-I: Introduction to Databases: **12 Hours**

An overview of Database Management system, Characteristics of a Database, Database Administrators, Data, Information, and Knowledge, File system vs DBMS, Types of Database system, Data Dictionary, Advantage and Disadvantages of databases systems, Data Models, Schemas & Instances, DBMS Architecture & Data Independence, Level of Abstraction, Data Languages & Interfaces. Database Security: Authentication, Authorization and access control, DAC, MAC and RBAC models, Intrusion detection, SQL injection.

UNIT-II: Entity-Relationship Model : **12 Hours**

Data modeling using the Entity-Relationship Approach: E-R Model concepts, Entity types, entity set, attribute and key, relationships, relation types, roles and structural constraints, weak entities. Conventional Data Models & Systems, Mapping constraints, Generalization, Aggregation, reduction of an ER diagram to tables, extended ER model, relationship of higher degree.

UNIT-III: Relational Model and Relational Algebra: 12 Hours

Relational Model concepts, Relational Algebra: Terminology, Operators – Write, Operator – Retrieval, Select, Project, Union, Intersection, Difference, Cartesian Product, Join, Natural Join, Outer Join. RELATIONAL MODEL: Relationship Algebra Selection and Projection, Set Operations, Renaming, Joints, Division, Storage strategy: Indices, B-trees, hashing.

UNIT-IV: Relational Database Design: 12 Hours

Function Dependencies & Normalizations for Relational Database, Functional Dependencies, Normal Forms based on primary keys (INF, 2NF, 3NF & BCNF). Transaction management: ACID properties, serializability, Concurrency Control Techniques, Locking Techniques: Concept of concurrency control, Multivalued dependency and Join dependency. Concurrency Control: Lock Based Protocols; Time Stamped Based Protocols, Deadlock Handling, Recovery Techniques, Recovery Concepts, Database Backup and Recovery from catastrophic failures.

UNIT-V: Constraints, Views and SQL: 12 Hours

What is constraints, types of constrains, Integrity constraints, Key constraints or Uniqueness Constraints, Referential integrity constraints, Views: Introduction to views, data independence, security, updates on views, comparison between tables and views. SQL: What is SQL, Database Languages: DDL, DML, DCL, TCL, Filtering, Sorting, and Aggregating Data, JOINs (INNER, LEFT, RIGHT, FULL), Subqueries and Nested Queries, Introduction to PL SQL, Stored Procedures and Triggers, cursor, indexes.

Text/Reference Books:

1. Silberschatz, A., Korth, H.F., Sudarshan, S., Database System Concepts, McGraw-Hill International Edition, 2006 (5 th Edition)
2. Elmasri, R., Navathe, S.B., Fundamentals of Database Systems, Fourth Edition, Pearson Education,
3. Desai, B.C., An Introduction to Database Systems, Galgotia Publications,
4. Date, C.J., An Introduction to Database Systems, Pearson Education, 7 th Edition
5. Garcia-Molina, H., Ullman, J.D., Widom, J., Database Systems: The Complete Book, Pearson Education, 2002

*****

Name of course and code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Discipline Specific: Major 4 CSE MT-804	Theory of Computation	03	01	00	20	20	60	100

Course Description: Formal language and automata theory is an exciting, theoretical branch of computer science. Through automata, computer scientists are able to understand how machines compute functions and solve problems and more importantly, what it means for a function to be defined as computable or for a question to be described as decidable. The course deals with the concept of computability and mathematical models, such as finite automata, grammars and Turing machines, and the relations between these models.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Students understand that, how finite automata, stacking machines, context-free grammars and Turing machines work as well as solve problems.
- To understand the conversion of a non-deterministic finite automaton to a deterministic one, conversion of a finite automaton into a regular expression and vice versa, and minimization of a deterministic finite automaton;
- To understand the use of Chomsky's language hierarchy including the terms regular language, context-free language, Turing decidable language and Turing acceptable language.

UNIT-I: Finite Automata and Equivalence **12 Hours**

Introduction and Finite Automata: Motivation for studying theory of computation, Alphabets, Strings, Languages, Finite Automata (FA). Acceptance of strings, and languages, Deterministic Finite Automata (DFA) and Non Deterministic Finite Automata (NFA), Transition diagrams and Language recognizers. Conversions and Equivalence: Equivalence between NFA with and without e- transitions, NFA to DFA conversion. Equivalence between two FSMs. Minimization of Automata; Finite Automata with output – Moore and Mealy machines.

UNIT-II: Formal Language: **12 Hours**

Definitions of a grammar, derivations and the language generated by a grammar, Chomsky classification of language. Regular Expressions; Regular Languages: Definition of Regular Expressions, FA and Regular Expressions, Regular Languages, Conversion from RE to FA and FA to RE. Pumping lemma for regular languages, Pumping Lemma for regular set, Closure properties of regular languages

UNIT-III: Grammar: **12 Hours**

Chomsky classification of grammar. Context Free Grammars and Languages (CFG), Language generated by a CFG, Leftmost. Rightmost derivations, Derivation trees. Ambiguity in grammars and languages, Simplification of: Context Free Grammars. Chomsky normal form (CNF), Greibach normal form (GNF), Pumping Lemma for Context Free Languages.

UNIT-IV: Push Down Automata: **12 Hours**

Definition and languages acceptable by PDA. Instantaneous description, Automata on Infinite Words: Polynomial Time Reductions and Cook-Levin Theorem, Büchi Automata and ω-Regular

Languages, PDA computation, Equivalence and Conversion of CFG's and PDA's, Deterministic PDA.

UNIT-V: Turing Theory: 12 Hours

Turing Machines, Time Complexity of Turing Machines and the Classes P and NP, definition. Model. Language acceptability by Turing Machine, instantaneous description, design of TM, Variations of TM: Multi-tape TMs, Non Deterministic TM, The Church-Turing thesis. Undesirability Definitions of recursively enumerable and recursive languages, Universal Turing machine.

Text/Reference Books:

1. J. Hop croft, J. D. Ullman, R Introduction to Automata Theory. Languages and Computation, 3rd Ed., Pearson
2. Daniel I.A. Cohen, Introduction, to Computer Theory, John Wiley; Sons.
3. N. Chandrasekhar K.L.P. Mishra. Theory of Computer Science. Automata Languages Computations;, PHI publications.
4. Michael Sipser, Introduction to the Theory of Computation, Books/Cole Thomson Learning, 2001.
5. Theory of Computation by V Sarthi.
6. Formal Languages and automata theory by C K Nagpal.

*****

Name of Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Minor 1 CSE MP-805	Advanced Java Programming Laboratory	00	00	02	00	00	50	50

List of Porgrams:

1. Write a Java Program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green. When a radio button is selected, the light is turned on, and only one light can be on at a time No light is on when the program starts.
2. Write a Java Program to create multiple frames, which create a Frame2 with a 'back 'button, such that when a user click 'back' button, Frame 2 is closed and we see the Frame1 only?
3. Write a Java Program to create a frame using swing in which create a push button with a label and image. When the button is clicked an image is displayed in the Frame?
4. Write a Java Program to create a student table, which includes name, roll no, branch and age or DOB?

5. Write a Java Program to create a tabbed pane with two tabs. In the first tab sheet, display some push buttons with names of Branches. In second tab sheet, display checkboxes with names of subjects.
6. Write a java program to create a menu with several menu items by implementing JMenu.
7. Write a java program to create a combo box with some name of some places. The user can select any one name from the list and the selected country name is displayed in the frame? (Use JComboBox)
8. Write a java program to select multiple places and displayed in Frame using JList?
9. Write a java program to create a bean that performs conversion of American dollar to Indian rupee.
10. Write a java program to create a bean that counts the number of button clicks?
11. Installation of Apache Tomcat webserver.
12. Write a java Program to create a simple servlet and run it using tomcat server.
13. Write a Java Program to create a JSP page to display a simple message along with the current Date?
14. Write a Java Program to create a JSP page to display the random number?
15. Create a struts 2 application.

*****

Name of Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
(Minor 2) Vocational CSE MI-806	Advanced DBMS Laboratory	00	00	02	00	00	50	50

Course Description: To understand the different issues involved in the design and implementation of a database system. To study the physical and logical database designs, database modeling, and relational models. To understand and use SQL to query, update, and manage a database. To develop an understanding of essential DBMS concepts such as: transaction processing, integrity, concurrency, and recovery in databases. To design and build a simple database system and demonstrate competence with the fundamental tasks involved with modeling, designing, and implementing a DBMS.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Demonstrate an understanding of the relational data model.
- Performing PL/SQL programming using concept of Cursor Management, Error Handling, Triggers.
- Apply various normalization techniques.
- Formulate, using SQL, solutions to a broad range of query and data update problems.

- To develop appropriate Databases to a given problem that integrates ethical, social, legal, and economic concerns

List of Programs:

1. Implementation of DDL commands of SQL with suitable examples
2. Implementation of different types of operators in SQL:
3. Arithmetic Operators, Logical Operators, Comparison Operator, Special Operator, Set Operation
4. Practice on transaction processing.
5. Practice on functional dependencies.
6. Implement used defined procedures and functions using PL/SQL blocks.
7. Implement PL/SQL programmes using exception handling.
8. Implementation of different types of constraints.
9. Implement PL/SQL programmes using control structure.
10. Write the query for creating the users and their role.
11. Implementation of different types of function with suitable examples:
12. Number function, Aggregate Function, Character Function, Conversion Function, Date Function
13. Implementation the following:
    • Group By & having clause
    • Order by clause
    • Indexing
14. Implementation of different types of Joins: inner, outer, equi, natural join
15. Implementation the following:
    • Sub queries
    • Views
16. Study & Implementation of Database Backup & Recovery commands.
17. Create table EMPLOYEES with following fields (Employee_Id, First_Name, Last_Name, Email, Phone_Number, Hire_Date, Job_Id, Salary, Commission_Pct, Manager_Id, Department_Id) and write SQL command to get the following:
    a. Find out the employee id, names, salaries of all the employees
    b. List out the employees who works under Manager 100.
    c. Find the names of the employees who have a salary greater than or equal to 4800
    d. List out the employees whose last name is 'AUSTIN'
    e. Find the names of the employees who works in departments 60,70 and 80
    f. Display the unique Manager_Id.
18. Create Client_master with the following fields(ClientNO, Name, Address, City, State, bal_due) and write SQL command to get the following
    a. Insert five records
    b. Find the names of clients whose bal_due> 5000 .
    c. Change the bal_due of ClientNO " C123" to Rs. 5100
    d. Change the name of Client_master to Client12 .

e. Display the bal_due heading as "BALANCE".

19. Study & Implementation of Rollback, Commit, Savepoint.
20. Perform the queries for triggers
21. Perform the following operation for demonstrating the insertion , updation and deletion using the referential integrity constraints
22. Make a table that have an employee Perform AND/OR, and between operation.
23. Create the table for facilities having faculty-id, dept. no., designation name and group by similar dept.no. Facilities by using count function.
24. Create a table for emp. Using following data:- emp. name, emp age, emp salary, emp city & display the emp salary in ascending and descending order.

Text/Reference Books:

1. Database system concepts, design and applications, S.K. Singh, Pearson Education, New Delhi.
2. Sql/PL/SQL, Batross, Ivan BPB.

*****

BCA - Forth Year, VIII-Semester- (Honours with Research)

Name of Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Research Specific: Major 1 CSER MT--801	Research Design and Methodology	03	01	00	20	20	60	100

Course Description: This course provides students with the fundamental knowledge and skills to understand, design, and critically evaluate research in their field.

Course Learning Outcomes:

- Define research and explain its significance in advancing knowledge and informing decision-making across various disciplines.
- Compare and contrast different research designs (exploratory, descriptive, experimental) and select the most appropriate design to address a specific research question.
- Develop a sampling plan, including choosing a sampling method and determining sample size, that aligns with the research question and ensures representativeness.
- Analyze data using appropriate descriptive and inferential statistical techniques to answer research questions and test hypotheses.

- Interpret research findings accurately, draw valid conclusions, and communicate research results effectively in written and oral formats.

UNIT- I: Foundations of Research Basics: 12 Hours

Foundations of Research What is research? Objectives, motivations, types, The scientific method and its limitations. Key concepts: Theory, empiricism, variables, hypotheses.

UNIT-II: Research Design and Ethics: 12 Hours

Research Design, Purpose and importance of research design, Major types: Exploratory, descriptive, experimental, Qualitative vs. quantitative approaches. Ethical considerations in research design.

UNIT III: Sampling and Data Collection 12 Hours

Sampling and Data Collection, Defining the population and sample, Probability vs. non-probability sampling techniques, Determining an appropriate sample size, Common data collection methods: Surveys, interviews, observations, Longitudinal and Cross-Sectional Designs.

UNIT IV: Data Analysis and Methods 12 Hours

Data Analysis, Preparing data for analysis: Cleaning, coding, Descriptive statistics: Summarizing and visualizing data, Mixed Methods Research: Integrating Qualitative and Quantitative Approaches,Basic inferential statistics: Testing hypotheses, relationships, Using statistical software (e.g., SPSS, R).

UNIT V: Data Interpretation and Communication: 12 Hours

Interpreting and Communicating Findings, Drawing valid conclusions from data,Identifying limitations of the research, Ensuring Validity and Reliability in Data Collection, Ethical considerations in data interpretation, Communicating findings effectively: Reports, presentations.

Text/Reference Books:

1. Cooper, D. R., & Schindler, P. S.. *Business research methods* (9th ed.). McGraw-Hill Education.
2. Bryman, A., & Bell, E.. *Business research methods*. Oxford University Press.
3. Kothari, C. R.. *Research methodology: Methods and techniques* (2nd ed.). New Age International.

*****

Name of Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		

Research Specific: Major 2 CSER MP-802	Dissertation	03	01	00	20	20	60	100

*****

Name of Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Research Specific: Major 3 CSER MT-803	Research Publication	00	00	03	00	00	100	100

*****

Name of Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Research Specific: Major 4 CSER MT 804	Viva-Voce	02	01	00	20	20	60	100

*****

Name of Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Minor 1 CSER MI-805	Advance Research Laboratory	00	00	02	00	00	50	50

*****

Name of Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Minor 2 CSER MI-806	Seminar Presentation	00	00	02	10	10	30	50

*****

National Education Policy-2020, w.e.f. July-2024
Academic Program Handbook: Curriculum and Syllabus
Master of Computer Applications (M.C.A.)
Semester I to IV

Indira Gandhi National Tribal University (IGNTU), Amarkantak, M.P. (A Central University)

Department of Computer Science
Faculty of Computer Science
Amarkantak, Madhya Pradesh-484887, Bharat (India)

Master of Computer Applications (M.C.A.)
(To be implemented from Academic Year: 2024-24 onwards under NEP 2020)

D. Name of the Academic Programme: Master of Computer Applications

Preamble:

This syllabus is the extension of the existing syllabus which is currently being taught to M.C. A. of IGNTU for the last few years, but modified as per need of IT and R & D Industry and to be implemented in academia from the academic year 2020 onwards. However, there are few changes incorporated in the existing syllabus. It is believed that the proposed qualitative change in the way M.C.A. is taught, will offer a more enriched learning experience. It aims to provide technology-oriented students with the knowledge and ability to develop creative solutions, and better understand the effects of future developments of computer systems and technology on people and society. The syllabus is about developing skills to learn new technology, grasping the concepts and issues behind its use and the use of computers.

7. Program Educational Objectives:

- To train the post graduates to acquire in depth knowledge of fundamental concepts and programming skills for holistic development of the pupils.
- To prepare the post graduates for productive careers in the software industry, corporate sector, Government Organizations, and R & D.
- To prepare post graduates to acquire excellent computing ability so that they can analyze, design and create Solutions for real time problems.
- To apply the current tools and techniques to create systems for solving Industry oriented problems.
- To prepare post graduates to gain multidisciplinary knowledge through real time case studies, projects and industry internships to meet the industry needs.

8. Pedagogical Teaching and Learning Methods:

Pedagogical Teaching and Learning Methods:

We would prioritize the integration of modernized and scientifically designed teaching pedagogy. It's essential that any innovative teaching approach be complemented by an equally effective learning process, ensuring that both evolve together. Below are the key techniques we would implement:

- **Crossover Teaching and Learning:** This approach involves relating classroom concepts to real-life examples. For instance, while explaining a theoretical concept, we would draw parallels to practical scenarios that students can easily relate to, thereby enhancing their understanding.
- **Context-Based Teaching:** When discussing theoretical concepts, we use fictitious examples to simplify and clarify ideas. We often draw inspiration from examples cited by renowned professors from institutions like Stanford and MIT to ensure the examples are both relevant and educational.
- **Adaptive Teaching:** We tailor our teaching methods to the varying learning capacities of students. By categorizing students based on their grasp of the material, we develop specialized strategies to address the needs of each group, ensuring that all students have the opportunity to succeed.
- **Incidental Learning:** This involves teaching concepts in unexpected contexts. For example, if we find ourselves in a traffic jam, we might take that opportunity to explain the concept of deadlocks, making learning spontaneous and contextually relevant.
- **Computational Thinking:** To equip students with the skills to tackle complex problems, we emphasize computational thinking. This approach encourages students to break down larger problems into smaller, manageable parts, which can then be solved systematically.
- **Comparative Teaching:** Given the diverse student body, we support the use of comparative teaching, where students' strengths and weaknesses are highlighted through peer comparison. This method not only promotes self-awareness but also motivates students to improve by learning from their peers.
- **Academic Exchange Hub:** We believe in fostering academic growth through exchange programs. By leveraging our contacts at prestigious institutions such as IIT`s, NIT`s, IIIT`s, Thapar University and BITS Pilani, we aim to create a network for the exchange of ideas. This will enhance both faculty and student quality over time through continuous collaboration.

Scheme of Examination:

- Mid Semester Examination (MSE) : 10+10= 20 Marks
- Internal Assessment (IA) : 15+5(Attendance) Marks
- End Semester Examination (ESE) : 60 Marks

9. Program Outcomes:

- Apply knowledge of computing fundamentals, computing specialization, Programming Skills, and domain knowledge to conceptualize computing models.

- Problem analysis Identify, formulate, research literature, and solve complex computing problems reaching substantiated conclusions using fundamental principles of computing sciences, and relevant domain disciplines.
- The 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.
- Innovation and entrepreneurship identify a timely opportunity and use innovation to pursue that opportunity to create value and wealth for the betterment of the individual and society at large.

E. MCA Curriculum

10. About the course: In the fastest growing era of Information Technology, almost every year new technology, techniques, programming languages are coming on the market. To be at par with vibrant changes, this MCA two years/MCA one year duration degree program curriculum has been designed. This is a very challenging task to design curriculum with absorbing rapid changes in technologies, and accordingly identify the core and elective subjects. In designing this curriculum, we have accommodated core courses and elective courses covering vast domains of Information technologies under new education policy (NEP 2020). This curriculum also has a strong laboratory and project orientation in which the use of new tools is used. Most courses will have an associated laboratory and it is expected that they will be equipped with the latest software tools.

11. Structure of the Curriculum: The MCA programme is designed based on more emphasis on the application part of computer science. The primary emphasis in MCA is on designing information systems for various organizations. There is high demand for MCA candidates in the development of application software, since the inception of the MCA courses in India, this is observed in industries, banking, insurance sectors etc., having high demand for MCA graduates. In the era of digital India, abundant domains come up with a huge demand of MCA graduates. So, major thrust is on giving the students a sound background in computing, business functioning and mathematics relevant to cutting edge information technologies such as refers to technological devices, techniques or achievements that employ the most current and high-level IT developments. A strong laboratory component is added as part of the curriculum. The laboratories, besides supplementing the theory course should also expose the student to the use of the latest software tools. Most of the courses are designed based on the Industry demands with all latest programming languages which are in high demand across the globe.

12. MCA Course Prerequisites and Period:

As per the NEP 2020, MCA 2 Year and MCA 1 Year program is designed.

MCA Two year (4 semesters) course: Candidates having B.C.A./B.Sc. (Computer Science/IT/ Mathematics or equivalent)/ B.Voc. (Software Development/IT) degree from a recognized Indian or foreign university (as per the AIU foreign equivalence list) having secured a minimum of 50% in aggregate in case of General and OBC categories and 45% in case of candidates belonging to SC/ST/PWD categories are eligible to apply.

MCA One year (2 semesters) course: Candidate is eligible if he/ she passed UG 4 year degree in Computer Science/Computer Application.

- **Medium of Instruction and Examination:**
  The medium of instruction as well as examination will be English only.*
  *As per NEP directive and guideline will be amended and stay infomed to students.

End Semester Examination: 60 Marks

The END SEMESTER Paper shall be of 60 marks of 3 hours duration. The pattern of Questions asked shall be as per the format provided by CoE.

Note: A student shall be eligible to appear in the End Semester Examination of course if he/she appeared in Mid Semester Examination and Internal Assessment and fulfills the requirement attendance, failing which he/she will not be permitted to appear in the End Semester Examination of the respective course.

13. Dissertation/Research Work/Project/ Internship/Seminar Evaluation:
The dissertation/Research Work/Project work is evaluated based on the following heads as per the requirements of evaluation criteria mentioned in curriculum framework:

1. Mid 1- Presentation & evaluation of Synopsis
2. Mid 2- Presentation & evaluation of Progress of work
3. End Semester:
   a. Evaluation of Dissertation
   b. Presentation and Viva

14. Dissertation/Research Work/Project: Dissertation/research work/project copied from other students will be considered to have used unfair means. If two dissertations are found identical by more than 40%, then zero marks will be awarded to both of them. In such a case the dissertation will have to be resubmitted on the new topic.

15. Committee for Evaluation:
The evaluation of components I and II is carried out by the course coordinator assigned by the Head of the Department.

Components III will be evaluated by a committee consisting of the Chairman of BoS or his/her nominee and an external examiner/internal examiner invited from other University/ Industry/ Society/ Community/Other Department or the same Department within the University.

16. Seminar: The purpose of a seminar is to enable students to improve their knowledge and understanding of a topic by engaging with key issues. Evaluation of the seminar will be based on presentation and viva voce.

17. Credit and Teaching Hours:
The credit and teaching hours shall be distributed as under:
Theory
1- Credit = 15 hours / per semester
2- Credit = 30 hours / per semester
3- Credit = 45 hours / per semester
4- Credit = 60 hours / per semester
Practical
1- Credit = 30 hours / per semester
2- Credit = 60 hours / per semester
Tutorial
1- Credit = 15 hours / per semester

18. Strategies: Keeping IT & Computer Science Syllabus (BCA-MCA) Current/ Relevant: IGNT University & Department of Computer Science:

- **Curriculum Review Cycles:** Most institutions have established procedures for periodic curriculum review and revision. Faculty should be aware of these cycles and actively participate in them.
- **Industry Collaboration:** Institutions often form advisory boards or engage with industry partners to gather input on curriculum needs and emerging trends.
- **Academic Freedom:** Faculty members generally have a degree of academic freedom in designing their courses. However, this freedom is balanced with the institution's responsibility to offer relevant and high-quality programs.
- **Staying Current:** Faculty members have a professional obligation to stay abreast of advancements in their field. This includes attending conferences, reading research, participating in professional development, and engaging with industry practitioners.
- **Flexibility and Adaptation:** Incorporating new technologies, tools, and concepts into the curriculum requires flexibility and a willingness to adapt teaching methods.
- **Transparency and Communication:** Clearly communicate any syllabus changes to students, HoD, BoS Chairman and Dean, explaining the rationale behind the updates and how they connect to industry needs.

- **Accreditation Standards:** Educational institutions often need to meet accreditation standards, which may include requirements for curriculum relevance and industry alignment.
- **Intellectual Property:** When incorporating external materials or resources into the curriculum, faculty should be mindful of copyright and intellectual property rights.

In essence, ensuring that IT and Computer Science syllabus remain current and relevant is an ongoing process that requires collaboration, communication, and a commitment to staying informed about industry trends. Faculty members play a crucial role in this process, working within the framework of UGC/ IGNTU guidelines.

*****

F. (M.C.A.) - Semester Wise Curriculum Framework Under NEP 2020
Master of Computer Applications
Two-Year Master Program

MCA FIRST SEMESTER							
Name of Course	Name of course and code	Course Title	MM	L	T	P	C
Discipline Specific: Major 1*	MCA-101	Overview Programming Concepts	100	3	1	0	4
Discipline Specific: Major 2*	MCA -102	Machine Learning	100	3	1	0	4
Discipline Specific: Major 3*	MCA -103	Advanced Data Structures and Algorithms	100	3	1	0	4
Discipline Specific: Major 4 **	MCA -104	Digital Logic and Computer Design	100	3	1	0	4
Discipline Specific: Major 5	MCA -105	Machine Learning Laboratory	50	0	0	2	2
Discipline Specific: Major 6	MCA -106	Advanced Data Structures and Algorithms Laboratory	50	0	0	2	2
		Total Credits	20				

* Pick subjects from Departmental Basket Pools

** Out of two subjects - students should opt one of the subjects from above for their study.

*****

MCA SECOND SEMESTER							
Name of Course	Name of course and code	Course Title	MM	L	T	P	C
Discipline Specific: Major 1*	MCA -201	Advanced Java Programming	100	3	1	0	4
Discipline Specific: Major 2*	MCA -202	Advanced Database Management System	100	3	1	0	4
Discipline Specific: Major 3*	MCA -203	Formal Languages and Automata Theory	100	3	1	0	4
Discipline Specific: Major 4 **	MCA -204	IoT and Cloud Computing	100	3	1	0	4
Minor 1	MCA -205	Advanced Java Programming Laboratory	50	0	0	2	2
(Minor 2) Vocational	MCA -206	Advanced RDBMS Laboratory	50	2	0	0	2
		Total Credits					20

* Pick subjects from Departmental Basket Pools

*****

MCA THIRD SEMESTER							
Name of Course	Name of course and code	Course Title	MM	L	T	P	C
Discipline Specific: Major 1	MCA-301	DSE-1 (I/II/III/IV)	100	3	1	0	4
Discipline Specific: Major 2	MCA-302	DSE-2 (I/II/III/IV)	100	3	1	0	4
Discipline Specific: Major 3	MCA-303	DSE-3 (I/II/III/IV)	100	3	1	0	4

Discipline Specific: Major 4	MCA-304	Seminar/ Poster Presentation/ Group Discussion	100	0	0	4	4
Discipline Specific: Major 5	MCA-305	Project Work-I	100	0	0	4	4
		Total Credits					20

*****

MCA FOURTH SEMESTER							
Name of Course	Name of course and code	Course Title	MM	L	T	P	C
Discipline Specific: Major 1	MCA-401	Project Work-II	500	0	0	20	20
		1. Project Synopsis	100	4	0	0	4
		2. Presentation of work	100	0	0	4	4
		3. Project Prototype	100	0	0	4	4
		4. Project Dissertation	100	4	0	0	4
		5. Viva-Voce	100	0	0	4	4
		Total Credits					20

*****

MCA - First Year, I-Semester

Name of Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Discipline Specific: Major 1 MCA-101	Overview Programming Concepts	03	01	00	20	20	60	100

Course Description: To enable the students to learn about advanced features of C, C++, Unix and Android Programming.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Design an algorithmic solution for a given problem and translate it into a program.
- Use the appropriate control statements to solve the given problem.
- Implement different Operations on arrays and use functions to solve the given problem.
- Understand pointers, structures and unions
- Understanding of ADT with C++ classes
- Proper knowledge of OOP concepts in C++
- Basic knowledge of Unix and Shell programming
- Basic knowledge of Android programming

UNIT-I: C Language Fundamentals: 15 Hours

About "C" language, Structure of "C" program, Identifiers and Keywords, Constants, Variables, Scope of variables, Operators, Library Functions, Input/Output Statements, Compound statements and block, Control structures, Decision making and Branching, Decision making & looping, Functions, Prototype of Function, Call by Value, Nesting of Functions, Recursion, Array, Multidimensional Array, String, Basic functions dedicated to String Manipulation, Pointer variable and its importance, Pointer Arithmetic, passing parameters by reference, pointer to pointer, Declaration and Defining Structure, & Union, Enumeration.

UNIT-II: Principal of Object-Oriented Programming: 10 Hours

Procedural Vs. Object-Oriented Programming, OOPS paradigms, basic concept of OOP: abstraction, encapsulation, modularity, data hierarchy through inheritance, polymorphism and typing, Applications and Benefits of OOP, Basic components of a C++ Program and program structure, Constructors & Destructors, Functions, Polymorphism, Constructor overloading, Operator overloading, Inheritance, Exception handling.

UNIT-III: UNIX Concepts & Shell Programming: 15 Hours

Introduction to UNIX, UNIX System Organization (The Kernel and the Shell), Files and Directories, Library Functions and System Calls, Editors (vi and ed), UNIX Shell Programming: Types of Shells, Shell Meta Characters, Shell Variables, Shell Scripts, Shell Commands, UNIX Environment, UNIX System Administration: File System, Mounting and Unmounting File System, System Booting, Shutting Down, Handling User Account, Backup, Recovery, Security.

UNIT: IV: Introduction to Mobile Applications: **10 Hours**

Characteristics of Mobile Applications, An Overview of the Android Platform: Introducing Android, Setting Up Your Android Development Environment, Writing Your First Android Application, Mastering the Android Development Tools.

UNIT V: Android Application Basics: 10 Hours

Understanding the Anatomy of an Android Application, Defining Your Application Using the Android Manifest File, Managing Application Resources, Using Android Preferences, Working with Files and Directories, Using Content Providers, Designing Compatible Applications.

Text/Reference Books:

1. E. Balaguruswamy, Programming in ANSI C 5th Edition McGraw-Hill
2. Yashvant P. Kanetkar, Let Us C", BPB Publications,
3. Programming: Principles and Practice Using C++ by Bjarne Stroustrup
4. Yashwant Kanitkar, "UNIX Shell Programming", BPB, 2003, India, New Delhi.
5. Sumitabh Das, "UNIX Concepts and applications", TMH, 2017, USA, New York.
6. Lauren Darcey and Shane Conder, "Android Wireless Application Development", Pearson Education, 2nd ed. (2011)
7. Clifton, "Android User Interface Design: Turning Ideas and Sketches into Beautifully Designed Apps", Addison-Wesley Professional, 2013.

*****

Name of Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Discipline Specific: Major 2 MCA-102	Machine Learning	03	01	00	20	20	60	100

Course Description: This course provides a comprehensive introduction to the principles and practices of data science, equipping students with the knowledge and skills to extract meaningful insights from data. Students will explore the entire data science lifecycle, from data collection and preparation to exploratory data analysis, data visualization, and the application of statistical methods. The course also covers essential database technologies for data science, including SQL and NoSQL databases, enabling students to effectively manage and query structured and unstructured data.

Course Learning Outcomes:

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

- Articulate the importance of data science and its role in various domains.
- Understand the data science process and the key steps involved in a typical data science project.
- Perform exploratory data analysis using statistical measures, data visualization techniques, and appropriate tools.
- Write SQL queries to retrieve, filter, and aggregate data from relational databases.
- Work with NoSQL databases, including document databases like MongoDB, to handle unstructured data.

- Apply data science methodologies to solve real-world problems and communicate findings effectively.

UNIT-I: Introduction to Machine Learning: 10 Hours

Introduction, What is Human Learning, Types of Human Learning, Learning under expert guidance, Learning guided by knowledge gained from experts, Learning by self, What is Machine Learning?, How do machines learn?, Well-posed learning problem, Types of Machine Learning, Supervised learning, Unsupervised learning, Reinforcement learning, Comparison – supervised, unsupervised, and reinforcement learning, Problems Not To Be Solved Using Machine Learning, Applications of Machine Learning, Banking and finance, Insurance, Healthcare, State-of-The-Art Languages/Tools In Machine Learning, Python, R, Matlab, SAS, Other languages/tools, Issues in Machine Learning, Summary

UNIT-II: Preparing to Model: [2]10 Hours

Introduction, Machine Learning Activities, Basic Types of Data in Machine Learning, Exploring Structure of Data, Exploring numerical data, Plotting and exploring numerical data, Exploring categorical data, Exploring relationship between variables, Data Quality and Remediation, Data quality, Data remediation, Data Pre-Processing, Dimensionality reduction, Feature subset selection, Summary

UNIT-III: Modeling and Evaluation: 10 Hours

Introduction, Selecting a Model, Predictive models, Descriptive models, Training a Model (for Supervised Learning), Holdout method, K-fold Cross-validation method, Bootstrap sampling, Lazy vs. Eager learner, Model Representation and Interpretability, Underfitting, Overfitting, Bias – variance trade-off, Evaluating Performance of a Model, Supervised learning – classification, Supervised learning – regression, Unsupervised learning – clustering, Improving Performance of a Model, Summary

UNIT-IV: Supervised Learning-Classification: 15 Hours

Introduction, Example of Supervised Learning, Classification Model, Classification Learning Steps, Common Classification Algorithms, k-Nearest Neighbour (kNN), Decision tree, Random Forest model, Support vector machines, Summary, Supervised Learning- Regression**:** Introduction, Example of Regression, Common Regression Algorithms, Simple linear regression, Multiple linear regression, Assumptions in Regression Analysis, Main Problems in Regression Analysis, Improving Accuracy of the Linear Regression Model, Polynomial Regression Model, Logistic Regression, Maximum Likelihood Estimation, Summary

UNIT-V : Unsupervised Learning: 15 Hours

Introduction, Unsupervised vs Supervised Learning, Application of Unsupervised Learning, Clustering, Clustering as a machine learning task, Different types of clustering techniques,

2

Partitioning methods, K-Medoids: a representative object-based technique, Hierarchical clustering, Density-based methods – DBSCAN, Finding Pattern using Association Rule, Definition of common terms, Association rule, The apriori algorithm for association rule learning, Build the apriori principle rules, Summary

Text/ Reference Books:

6. Tom M. Mitchell- Machine Learning- McGraw Hill Education, International Edition
7. Bishop, Christopher. Neural Networks for Pattern Recognition. New York, NY: Oxford University Press
8. Ethem Alpaydin, "Introduction to Machine Learning (Adaptive Computation and Machine Learning)", The MIT Press
9. Machine Learning by Subramanian Chandramouli, Saikat Dutt, Amit Kumar Das, 2018, Pearson Education India
10. Christopher M. Bishop Pattern Recognition and Machine Learning- Springer, 2nd edition

*****

Name of Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Discipline Specific: Major 5 :MCA-105	Machine Learning Laboratory	00	00	02	00	00	50	50

List of practical Program:

1. Familiarization with NumPy, Panda and Matplotlib by Loading Dataset in Python.
2. Implement and demonstrate the FIND-S algorithm for finding the most specific hypothesis based on a given set of training data samples. Read the training data from a .CSV file.
3. For a given set of training data examples stored in a .CSV file, implement and demonstrate the Candidate-Elimination algorithm to output a description of the set of all hypotheses consistent with the training examples.
4. Write a program to demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.
5. Build an Artificial Neural Network by implementing the Backpropagation algorithm and test the same using appropriate data sets.
6. Write a program to implement the naïve Bayesian classifier for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering few test data sets.
7. Assuming a set of documents that need to be classified, use the naïve Bayesian Classifier model to perform this task. Built-in python classes/API can be used to write the program. Calculate the accuracy, precision, and recall for your data set.

8. Write a program to construct a Bayesian network considering medical data. Use this model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set. You can use Python ML library classes/API.
9. Apply EM algorithm to cluster a set of data stored in a .CSV file. Use the same data set for clustering using k-Means algorithm. Compare the results of these two algorithms and comment on the quality of clustering. You can add Python ML library classes/API in the program.
10. Write a program to implement k-Nearest Neighbour algorithm to classify the iris data set. Print both correct and wrong predictions. Python ML library classes can be used for this problem.

*****

Name of the Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Discipline Specific: Major 3 MCA-103	Advanced Data Structures and Algorithms	03	01	00	20	20	60	100

Course Description: To learn the advanced concepts of data structure and algorithms and its implementation. The course has the main ingredients required for a computer science graduate and has all the necessary topics for assessment of advanced data structures and algorithms.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Basic ability to analyze algorithms and to determine algorithm correctness and time efficiency class.
- Master a variety of advanced abstract data type (ADT) and data structures and their implementations.
- Ability to apply and implement learned algorithm design techniques and data structures to solve problems.

UNIT-I: Algorithms and Data Structures: 10 Hours

Algorithms, Performance analysis- time complexity and space complexity, Asymptotic Notation- Big Oh, Omega and Theta notations, Complexity Analysis Examples. Data structures-Linear and non linear data structures, ADT concept, Linear List ADT, Array representation, Linked representation, singly linked lists -insertion, deletion, search operations, doubly linked lists- insertion, deletion operations, circular lists. Representation of single, two dimensional arrays, Sparse matrices and their representation.

UNIT-II: Stack and Queue: 10 Hours

Stack and Queue ADTs, array and linked list representations, infix to postfix conversion using stack, implementation of recursion, Circular queue-insertion and deletion, Dequeue ADT, array and linked list representations, Priority queue ADT, implementation using Heaps, AVL Trees and Red-Black Trees, Insertion into a Max Heap, Deletion from a Max Heap.

UNIT-III: Searching and Sorting: **15 Hours**

Linear and binary search methods, Hashing-Hash functions, Collision Resolution methods-Open Addressing, Chaining, Rehashing, double hashing. Sorting –Bubble sort, Insertion sort, Quick sort, Merge sort, Heap sort, Radix sort, comparison of sorting methods.

UNIT-IV: Trees and Graphs Overview: **15 Hours**

Trees- Ordinary and Binary trees terminology, Properties of Binary trees, Binary tree ADT, representations, traversals, Threaded binary trees. Segment Trees and Fenwick Trees (Binary Indexed Trees), Graphs- Graphs terminology, Shortest Path Algorithms (Dijkstra's, Bellman-Ford), Graph ADT, representations, graph traversals/search methods-DFS and BFS, Applications of Graphs-Minimum cost spanning tree using Kruskal's and Prim's algorithm, Shortest-path Algorithms: Dijkstra's and Floyd's algorithm, Topological sort.

UNIT-V: Search Trees and Algorithms: **15 Hours**

Search trees- Binary search tree-Binary search tree ADT, Advanced DP Problems (Matrix Chain Multiplication, Egg Dropping), insertion, deletion and searching operations, Balanced search trees, AVL trees-Definition and examples only, Red Black trees – Definition and examples only, B-Trees-definition, insertion and searching operations, Text compression-Huffman coding and decoding, Pattern matching algorithm, Need of approximation algorithms: Introduction to P, NP, NP-Hard and NP Complete.

Text/Reference Books:

1. Thomas Coremen, "Introduction to Algorithms", Third edition, Prentice Hall of India (2009).
2. Kleinberg J., Tardos E., "Algorithm Design", 1st Edition, Pearson, 2012.
3. Motwani R., Raghavan P., "Randomized Algorithms", Cambridge University Press, 1995. 4. Vazirani, Vijay V., "Approximation Algorithms", Springer, 2001.

*****

Name of Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Discipline Specific: Major 6 MCA-106	Advanced Data Structures and Algorithms Laboratory	0	0	2	0	0	50	50

Course Description: To provide the knowledge of basic data structures and their implementations. To understand importance of data structures in context of writing efficient programs. Students can develop skills to apply appropriate data structures in problem solving.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Choose a suitable Data Structures for an application
- Develop ability to implement different Sorting and Search methods
- Have knowledge onData Structures basic operations like insert, delete, search, update and traversal
- Design and develop programs using various data structures

List of Programs

1. Write a program to initialize the element into array and display.
2. Write a program to swap the first element with last, second to second last and so on (reversing elements) by using an array.
3. Write a program to display the sum of all the elements of the array.
4. Write a program to search an element into an array by using Linear Search.
5. Write a program to search an element into an array by using Binary Search.
6. Write a program to implement bubble sort.
7. Write a program to implement selection sort.
8. Write a program to implement quick sort.
9. Write a program to implement insertion sort.
10. Write a program to search an element in the 2-dimensional array by using linear search.
11. Write a program to merge two sorted arrays into one sorted array.
12. Write a program to perform the following operation in matrix.
    a. Addition b. Subtraction c. Transpose
13. Write a program to find multiplication of two matrices.
14. Write a program to print following matrix
    a. Upper diagonal b. Lower diagonal
15. Write a program to print sum of diagonal elements.
16. Write a program to implement stack using array to perform:
    a. PUSH b. POP c. Display
17. Write a program to convert infix to postfix by using stack.
18. Write a program to implement factorial by using a recursion function.
19. Write a program to print reverses of a given number by using stack.
20. Write a program to implement a queue using an array.
    a. Insert b. Delete c. Display
21. Write a program to implement a circular queue using an array.
22. Write a program to implement link list with following operations:
    a. Insert b. Display c. Delete

23. Write a program to implement stack using a link list.
24. Write a program to implement a queue using a link list.
25. Write a program to implement a circular queue using a link list.
26. Write a program to concatenate two link lists.
27. Write a program to add two polynomials with the help of a link list.
28. Write a program to implement a circular queue using a link list.
29. Write a program to reverse the link list.
30. Write a program to implement the following operations on Binary Search Tree
    a. Insert b. Delete c. Search d. Display

Text/Reference Books:

1. "Data Structures using C", ISRD group Second Edition, TMH
2. "Data Structures through C", Yashavant Kanetkar, BPB Publications
3. "Data Structures Using C" , Balagurusamy E. TMH

*****

Name of Course and **Code**	**Course Title**	**L**	**T**	**P**	**Sessional**		**ESM**	**Total**
					MSE	**IA**		
Discipline Specific: Major 4 MCA-104	Digital Logic and Computer Design	03	01	00	20	20	60	100

Course Description:

This course, provides a solid foundation in the principles of digital electronics, focusing on the computer design and analysis of both combinational and sequential digital circuits. Starting with the basics of semiconductor electronics, the course delves into various numbering systems, Boolean algebra, and the fundamental logic gates that form the building blocks of digital systems. Students will explore advanced topics such as Karnaugh Maps, combinational circuit design, and sequential circuits, including flip-flops, counters, and registers. The course emphasizes hands-on learning through the design and optimization of real-world digital circuits, encouraging creativity and problem-solving skills. By the end of the course, students will be equipped with the knowledge and expertise to design and implement efficient digital circuits, paving the way for innovation in computer engineering and related fields.

Course Learning Outcomes:

- Comprehensive understanding of digital electronics fundamentals and semiconductor devices for computer design basics.
- Proficiency in Boolean algebra for circuit simplification and optimization.
- Expertise in designing and implementing combinational circuits.
- Advanced skills in sequential circuit design, including counters and state machines.

- Practical experience in constructing and simulating digital circuits.
- Enhanced problem-solving abilities for innovative digital logic design.
- Strong foundation for advanced studies or professional careers in digital electronics.

UNIT-I Foundations of Digital Electronics: 12 Hours

Semi-Conductors Electronics, P type and N type Semiconductor, Working of Semiconductor Devices, Diode and Transistor Characteristics, Number System, Base of Number System, Types of Number System, Need of Number System, Conversion inbetween number systems, Octal Number System, Hexadecimal Number System, Decimal and Binary Number System, Different Type of Numbering Systems: Decimal, Octal, Binary, Hexadecimal, Conversation from one number system to another number system, Digital Signal, Modulation, Need of Modulation, Type of Modulations. Logic Gate Basics: NOT Gate, AND Gate, OR Gate, Exclusive-OR (XOR) Gate, Truth Tables for Logic Gates, Truth Tables for Combinational Logic. Boolean Algebra and Binary Math: Binary Addition, Binary Subtraction, Binary Complements, One's Complement, Two's Complement.

UNIT-II: Boolean Expressions and Boolean Algebra: 12 Hours

Need for Boolean Expressions, Symbols of Boolean Algebra, Boolean Expressions of Combinational Logic, Laws of Boolean Algebra, Rules of Boolean Algebra: NOT Rule, OR Rules, AND Rules, XOR Rules, DeMorgan's Theorem, Standard Boolean Expression Formats: Sum-of-Products, Converting an SOP Expression to a Truth Table, Converting a Truth Table to an SOP Expression, Product - of-Sums, Converting POS to Truth Table, Converting a Truth Table to a POS Expression, Two variable Boolean Equaction through K-Map, Three variable Boolean Equaction through K-Map, Four variable Boolean Equaction through K-Map and draw of Minimization Boolean Circuit, Various examples of two, three and four variable minimization of Boolean Equations using K-Map.

UNIT-III Design Methodology for Digital Circuits: 12 Hours

Understand the Problem Statement, Combinational vs. Sequential Circuits, Design Methodology: From Specification to Implementation, Standard Combinational Circuits: Encoders, Decoders, Multiplexers, Demultiplexers, Designing and Analyzing Small Combinational Circuits, Define the Truth Table: Inputs and Outputs, Derive Boolean Expressions: Karnaugh Maps (KMaps), Sum of Products (SOP) or Product of Sums (POS), Optimize the Boolean Expression, Draw the Logic Diagram, Connections, Circuit Diagram: Prepare a clean and labeled version of the final circuit diagram, Gate Count Reduction and Circuit drawing with universal gates, Examples of Various Design pattern.

UNIT-IV Design of Digital Combinational Circuits: 12 Hours

Design of Half Adder and Full adder using K-Map, Two variable, Three Variable and Four Variable Boolean Equation Design, Design a 4-bit binary to Gray code converter circuit, Create a circuit to convert a BCD (Binary Coded Decimal) number to its corresponding Gray code (Do not care condition), Design a Gray code to binary code converter circuit (3-bits and 4-bits),

Design a 3-bit binary to excess-3 code converter circuit, Create a circuit that converts a 4-bit binary number to its 2's complement, Implement a circuit to convert a 4-bit binary number to a 7-segment display output, Design a BCD to 7-segment display decoder circuit, Design even parity generator circuit, Implement a odd parity generator circuit, Design a 2-bit binary comparator circuit to compare two 2-bit numbers, Implement magnitude comparator circuit that outputs whether one number is greater than, less than, or equal to another, Design a BCD to excess-3 code converter circuit, Implement a circuit that converts a binary number to its one's complement, 84-2-1 code to Ex-3 Code Converter using K-Map.

UNIT-V Design of Digital Sequential Circuits: 12 Hours

Definition of Flip-Flops and Latches, Uses and Applications of Flip-Flop, Types of Flip-Flop, Design a D flip-flop, Design a T flip-flop using a D flip-flop. Synchronous Counters: Design a 4-bit synchronous binary counter, Create a synchronous mod-5 counter, Design a 3-bit synchronous up-down counter, Implement a synchronous decade (mod-10) counter, Design a 4-bit synchronous ring counter, Create a 4-bit synchronous counter that counts in a specific sequence (e.g., 0, 3, 5, 7, 9), Implement a 3-bit synchronous Gray code counter, Design a 4-bit synchronous BCD counter, Create a synchronous mod-12 counter. Asynchronous Counters (Ripple Counters): Design a 4-bit asynchronous (ripple) binary counter, Create a 3-bit asynchronous down counter, Design a mod-8 asynchronous counter, Design a 4-bit asynchronous up-down counter, What is Registers?, Applications of Registers, Design a 4-bit shift register using flip-flops, Definition of State Machines (FSM) and Sequence Generators, Memory Elements.

Text / Reference Books:

1. M. Moris Mano, Computer Systems Architecture, 3rd Edition, Pearson Education, 2007.
2. William Stallings, Computer Organization and Architecture: Designing for Performance, 10th Edition, Pearson Education, 2016.
3. David A. Patterson and John L. Hennessy Computer Organization and Design: The Hardware/Software Interface, 5th Edition, Elsevier, 2005.
4. Hayes, J.P., Computer Architecture and Organization, McGraw Hill (1998) 3rd ed.
5. Leigh, W.E. and Ali, D.L., System Architecture: software and hardware concepts, South Wester Publishing Co. (2000).

*****

Name of Course and **Code**	**Course Title**	**L**	**T**	**P**	**Sessional**		**ESM**	**Total**
					MSE	**IA**		
Discipline Specific: Major 5 MCA-105	Data Science and Big Data Analytics	03	01	00	20	20	60	100

Course Description: This course provides a comprehensive introduction to the rapidly evolving fields of Data Science and Big Data Analytics. Students will gain a strong foundation in the principles, techniques, and tools used to extract knowledge and insights from large and complex datasets. The course covers key concepts in data science, including machine learning, statistical modeling, and data visualization, along with the challenges and opportunities presented by big data. Students will gain hands-on experience with industry-standard tools like Hadoop, Hive, HBase, and Spark, enabling them to process, analyze, and interpret big data effectively.

Course Learning outcomes:

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

- Understand the fundamental concepts and principles of data science and big data analytics.
- Apply various machine learning techniques, including supervised and unsupervised learning, for prediction and forecasting.
- Utilize statistical modeling and visualization techniques to analyze and interpret data.
- Understand the challenges and opportunities associated with big data, including volume, velocity, variety, and veracity.
- Implement and manage big data solutions using Hadoop ecosystem tools such as HDFS, Hive, HBase, and Spark.
- Design and develop data-driven applications to address real-world problems.

UNIT- I: Introduction to Data Science: 12 Hours

What is the meaning of Data? More insights about Data Science, Types of Data Machine Learning, Supervised and Unsupervised Learning, Predictions and Forecasts, Innovation and Experimentation, Big Errors, Privacy, Theories, Models, Intuition, Causality, Prediction, Correlation, The Very Beginning: Got Math?, Exponentials, Logarithms, and Compounding, Normal Distribution. Vector Algebra, Open Source: Modelling in R, System Commands, loading Data, Root Solving, Regression, Heteroskedasticity, Auto-regressive models, Vector Auto-Regression, Prediction Trees.

UNIT- II: **Introduction to Big Data and Big Data Analytics: 12 Hours**

Introduction, Big Data, Distributed file system, Big Data and its importance, Four VsDrivers for Big data, Big data analytics, Big data applications, Algorithms using map reduce.

UNIT- III : **Hadoop and applied concepts:** **12 Hours**

Introduction, Hadoop, Definition of Hadoop, Big Data, Apache Hadoop, Hadoop Eco System, Moving Data in and out of Hadoop, Map Reduce, Understanding Inputs and Outputs of MapReduce, Data Serialization, Architecture of Hadoop. Introduction to Hdfs, Hive And Hiveql, Hbase and Spark.

UNIT-IV: Mastering Big Data Technologies: **12 Hours**

HDFS, Hive, HBase, and ZooKeeper: Introduction, Overview of HDFS-Installation, Shell, Java API, Hive-Architecture-Installation, Comparison with Traditional Database, HiveQL-Querying Data, Sorting, Aggregating, Map Reduce Scripts, Joins, Sub queries, Concepts of HBase, Advanced Usage, Schema Design, Advance Indexing, PIG5- Zookeeper, How it helps in monitoring a cluster? Uses of HBase in Zookeeper, How to Build Applications with Zookeeper?, Distinguish between HDFS and HBase.

UNIT-V: **Data Analytics Process:** **12 Hours**

There's Great Work Behind the Scenes**:** Introduction: More data, more questions for better answers. We can never say enough: "There is no good wind." , Understanding the basics: Identify what we already know and define the tasks to be accomplished. Which technology to adopt? Understanding data analytics is good, but knowing how to use it is better! (What skills do you need?).

Text/Reference Books:

1. García-Gil, D., Ramírez-Gallego, S., García, S., & Herrera, F.. A comparison on scalability for batch big data processing on Apache Spark and Apache Flink. *Big Data Analytics*, *2*. https://doi.org/10.1186/s41044-016-0020-2
2. Jiwat Ram, C. Z.. The implications of big data analytics on business intelligence: A qualitative study in China. *Procedia Computer Science*, *87*, 221–226.
3. Nazari, E., Shahriari, M. H., & Tabesh, H.. BigData analysis in healthcare: Apache Hadoop, Apache spark and Apache Flink. *Frontiers in Health Informatics*, *8*, 14.
4. Sagiroglu, S., & Sinanc, D.. Big Data: a review. In *2013 International Conference on Collaboration Technologies and Systems*. IEEE.
5. Shabbir, M. Q., & Gardezi, S. B. W.. Application of big data analytics and organizational performance: The mediating role of knowledge management practices. *J Big Data*, *7*, Article 47. https://doi.org/10.1186/s40537-020-00331-4

*****

MCA - First Year, II-Semester

Name of Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		

Discipline Specific: Major 1, MCA-201	Advanced Java Programming	03	01	00	20	20	60	100

Course Description: The objective of the Advanced Java Programming course is to deepen students' understanding of Java programming by exploring advanced concepts such as multithreading, networking, database connectivity, and web technologies. The course aims to equip students with the skills needed to develop complex, scalable, and high-performance Java applications.

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

- Implement multithreading and concurrency mechanisms in Java applications.
- Develop and integrate Java-based network applications using sockets and RMI.
- Utilize JDBC for database connectivity and perform CRUD operations.
- Design and develop web applications using Java servlets, JSP, and frameworks.
- Apply best practices in Java programming to create efficient and maintainable code.

UNIT-I J2EE and Swing Overview: 12 Hours

Introduction to J2EE, Difference among J2SE, J2EE, and J2ME, Overview of Collections API (List, Set, Map, Queue) Swing: The Origins of Swing-Swing Features-The MVC Connection-Components and Containers-TheSwingpackages-JLabel-JTextfield-TheSwingButtons-JButton-JToggleButton-CheckBoxes-RadioButtons-Jlist-Jmenu-JcomboBox-JTable-JScrollPane-JTabbedPane.

UNIT-II Interacting with Database: 12 Hours

Introduction to JDBC, Essential JDBC classes, connecting to database, inserting data in database, Retrieving data from database, Generics and Type Safety, Iterators and the For-Each Loop, deleting data in database, updating data in database, store image in the database, to retrieve image from database, to store file in database, retrieve file from database. Database Programming using JDBC, JDBC Drivers & Architecture, CURD operation Using JDBC, Connecting to non-conventional Databases.

UNIT-III Java Server Pages: 12 Hours

Introduction, Architecture of JSP, Life Cycle of JSP, Scripting elements (Scriplets, JSP Declarations, JSP Expression), Directive Elements (page, include, taglib), JSP Actions (include, setproperty, getproperty, forward, text), Implicit objects (request, response, out, Navigable Maps and Sets, page, Exception), including HTML in JSP.

UNIT-IV Java Server Technologies: 12 Hours

Servlet Web Application Basics, Architecture and challenges of Web Application, Introduction to servlet, Servlet life cycle, Developing and Deploying Servlets, Exploring Deployment, Descriptor (web.xml), Handling Request and Response.

UNIT-V Java Beans and Hibernate: 12 Hours

Introduction to Java Beans-Advantages of Java Beans-Using Bound and Constraint Properties-Persistence-Java Beans API-A Bean Example, Deadlock, Starvation, and Livelock

Hibernate: Introduction to Hibernate, Hibernate Architecture, Understanding Object Persistence, Hibernate Basics, Collections and custom types, Querying Persistent Objects, Hibernate Query Language (HQL).

Text/Reference Books:

1. Patrick Naughton, "COMPLETE REFERENCE: JAVA2", Tata McGraw-Hill, 2003
2. Schildt Herbert,The Complete Reference Java Seventh Edition 2011
3. Core Servlets and Java Server pages volume1: Core Technologies By Marty Hall and Pearson.
4. Java 6 Programming, Black Book, Dreamtech
5. Java Server Programming, Java EE6 (J2EE 1.6), Black Book, Dreamtech
6. Advanced Java Technology, By M.T. Savaliya, Dreamtech

*****

Name of Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Discipline Specific: Major - 2 MCA-202	Advanced Database Management System	03	01	00	20	20	60	100

Course Description: To develop skills in database design, implementation, and management. To familiarize students with SQL and its practical applications. Construct simple and moderately advanced database queries using, Structured Query Language (SQL). Understand and successfully apply logical database design principles, including E-R diagrams and database normalization.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Understand the concepts and principles of database management systems.
- Design and create relational databases using SQL.

- Understand and apply transaction processing concepts and convert schedules to serializable schedules.
- Apply normalization techniques to ensure data integrity.
- Illustrate different concurrency control mechanism to preserve data consistency in a multiuser environment.

UNIT-I: Introduction to Databases: 12 Hours

Introduction of File Organization, Database Management system, Characteristics of a Database, Database Administrators, Data, Information, and Knowledge, File system vs DBMS, Types of Database system, Data Dictionary, Advantage and Disadvantages of databases systems, Data Models, Schemas & Instances, DBMS Architecture: DBMS standardization, Global, Local, External, and Internal Schemas, Architectural models for Distributed DBMS. Data Independence, Level of Abstraction, Data Languages & Interfaces. Query Processing Overview, Query Optimization. Object oriented and object relational databases, Logical databases, Web databases, Distributed databases.

UNIT-II: Relational Database and E-R Modeling: 12 Hours

Relational database concepts: tables, tuples, attributes, keys, etc., Entity-Relationship (ER) modeling: Data modeling using the Entity-Relationship Approach. E-R Modeling: Entity types, entity set, attribute and key, relationships, relation types, roles and structural constraints, weak entities., Generalization, Aggregation, reduction of ER diagram to table, extended ER model, relationship of higher degree, Codd's rules, Relational Schemas, Introduction to UML, Constraints: Domain Constraints, Key constraints or Uniqueness Constraints, Entity Integrity constraints, Referential integrity constraints.

UNIT-III: Relational Algebra and Calculus: 12 Hours

Relational algebra: introduction, Selection and projection, set operations, renaming, Joins, Division, syntax, semantics. Operators, grouping and ungrouping, relational comparison. Calculus: Tuple relational calculus, Query Execution Plans and Analysis, Advanced Indexing Techniques (Clustered, Covering Indexes), Domain relational Calculus, calculus vs algebra, computational capabilities. Query processing and optimization: Evaluation of relational algebra expressions, Query equivalence, Join strategies, Query optimization algorithms. Storage strategy: Indices, B-trees, hashing.

UNIT-IV: Normalization and SQL: 12 Hours

Introduction, Need of Normalization, Purpose of normalization, Data Dependency, Armstrong's Axioms, Function Dependency, Normal Forms (1NF, 2NF, 3NF, BCNF), Denormalization and its implications, Open source and Commercial DBMS - MYSQL, ORACLE, DB2, SQL server. SQL: What is SQL, Database Languages: DDL, DML, DCL, TCL, operators, Clause (order by, group by, Having), Filtering, Sorting, and Aggregating Data, JOINs (INNER, LEFT, RIGHT, FULL), Subqueries and Nested Queries, Stored Procedures and Triggers, indexes. Introduction to PL SQL.

UNIT-V: Transaction management and Concurrency control: 12 Hours

Hashing Techniques and Applications in Databases, Transaction management: ACID properties, serializability and concurrency control, Lock based concurrency control (2PL, Deadlocks), Time stamping methods, optimistic methods, database recovery management. Database Security: Authentication, Authorization and access control, DAC, MAC and RBAC models, Intrusion detection, SQL injection. Deadlock Handling, Recovery Techniques, Recovery Concepts, Database Backup and Recovery from catastrophic failures.

Text/Reference Books:

1. Silberschatz, A., Korth, H.F., Sudarshan, S., Database System Concepts, McGraw-Hill International Edition, 2006 (5 th Edition)
2. Elmasri, R., Navathe, S.B., Fundamentals of Database Systems, Fourth Edition, Pearson Education,
3. Desai, B.C., An Introduction to Database Systems, Galgotia Publications,
4. Date, C.J., An Introduction to Database Systems, Pearson Education, 7 th Edition
5. Garcia-Molina, H., Ullman, J.D., Widom, J., Database Systems: The Complete Book, Pearson Education, 2002.

*****

Name of the Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Discipline Specific: Major - 3 MCA-203	Formal Language and Automata Theory	03	01	00	20	20	60	100

Course Description: Formal language and automata theory is an exciting, theoretical branch of computer science. Through automata, computer scientists are able to understand how machines compute functions and solve problems and more importantly, what it means for a function to be defined as computable or for a question to be described as decidable. The course deals with the concept of computability and mathematical models, such as finite automata, grammars and Turing machines, and the relations between these models.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Students understand that, how finite automata, stacking machines, context-free grammars and Turing machines work as well as solve problems.
- To understand the conversion of a non-deterministic finite automaton to a deterministic one, conversion of a finite automaton into a regular expression and vice versa, and minimization of a deterministic finite automaton;
- To understand the use of Chomsky's language hierarchy including the terms regular language, context-free language, Turing decidable language and Turing acceptable language.

UNIT-I: Fundamental of Finite Automata: 12 Hours

Introduction and Finite Automata: Motivation for studying theory of computation, Alphabets, Strings, Languages, Finite Automata (FA). Acceptance of strings, and languages, Deterministic Finite Automata (DFA) and Non Deterministic Finite Automata (NFA), Transition diagrams and Language recognizers. Conversions and Equivalence: Equivalence between NFA with and without e- transitions, NFA to DFA conversion. Equivalence between two FSMs. Minimization of Automata; Finite Automata with output – Moore and Mealy machines.

UNIT-II: Formal Language: 12 Hours

Definitions of a grammar, Parse Trees and Ambiguity in CFGs, derivations and the language generated by a grammar, Chomsky classification of language. Regular Expressions; Regular Languages: Definition of Regular Expressions, FA and Regular Expressions, Regular Languages, Pumping Lemma for Context-Free Languages, Definition and Types of Pushdown Automata (Deterministic and Non-Deterministic), Conversion from RE to FA and FA to RE. Pumping lemma for regular languages, Pumping Lemma for regular set, Closure properties of regular languages

UNIT-III: Grammar: 12 Hours

Chomsky classification of grammar. Context Free Grammars and Languages (CFG), Language generated by a CFG, Leftmost. Bottom-Up Parsing (LR Parsing, SLR, LALR), Rightmost derivations, Derivation trees. Ambiguity in grammars and languages, Simplification of: Context Free Grammars. Chomsky normal form (CNF), Greibach normal form (GNF), Pumping Lemma for Context Free Languages.

UNIT-IV: Push Down Automata: 12 Hours

Definition and languages acceptable by PDA. Instantaneous description, PDA computation, Equivalence & conversion of CFG's and PDA's, Deterministic PDA.

UNIT-V: Turing Theory: 12 Hours

Turing Machines, definition. Model. Language acceptability by Turing Machine, instantaneous description, design of TM, Variations of TM: Multi-tape TMs, Non Deterministic TM, The Church-Turing thesis. Error Detection and Recovery in Parsing, Undesirability Definitions of recursively enumerable and recursive languages, Universal Turing machine.

Text/Reference Books:

1. J. Hop croft, J. D. Ullman, R Introduction to Automata Theory. Languages and Computation, 3rd Ed., Pearson
2. Daniel I.A. Cohen, Introduction, to Computer Theory, John Wiley & Sons.
3. N. Chandrasekhar ; K.L.P. Mishra.Theory of Computer Science. Automata Languages Computation, PHI publications.
4. Peter Dehning, Jack B. Dennis, "Machines, Languages and Computation", Second Edition, PrenticeHall, 1978
5. Harry R. Lewis, Christos H. Papadimitriou, "Elements of the theory of computation", Second Edition, PrenticeHall, 1998

*****

Name of the Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Discipline Specific: Major 4 MCA-204	IoT and Cloud Computing	03	01	00	20	20	60	100

Course Description: This course provides an introduction to the concepts, technologies, and applications of the Internet of Things (IoT) and Cloud Computing. It explores how IoT devices collect and transmit data, and how cloud computing platforms store, process, and analyze this data. The course covers IoT architecture, protocols, and security issues, as well as cloud service models (IaaS, PaaS, SaaS), cloud deployment models, and the integration of IoT with cloud platforms. Practical case studies and hands-on projects are included to reinforce the understanding of real-world applications.

Course Learning Outcomes:

At the end of this course, students will be able to:

- Understand IoT Architectures and Protocols: Explain the key components, architectures, and communication protocols used in IoT systems.
- Implement IoT Solutions: Design and implement basic IoT systems using sensors, actuators, and microcontrollers.
- Analyze Cloud Service Models: Understand the various cloud service models (IaaS, PaaS, SaaS) and deployment models (public, private, hybrid).
- Integrate IoT with Cloud Platforms: Demonstrate how to connect and manage IoT devices using cloud platforms for data storage, processing, and analytics.
- Evaluate Security Challenges: Identify and address security and privacy issues in IoT and cloud computing environments.

- Apply IoT and Cloud in Real-world Applications: Develop and deploy IoT solutions leveraging cloud computing for practical use cases in industries such as healthcare, smart cities, and agriculture.

UNIT-I : Introduction to Internet of Things: 10 Hours

Internet of Things Concepts, Characteristics of IoT ,Physical and Logical design of IoT, Functional blocks of IoT, Communication models, Smart Object and Smart Environments

UNIT-II: Technologies and Application domains of IoT: 10 Hours

Internet of Things Framework, Communication technology infrastructure, Architecture and reference models, Application Domains: Energy, Agriculture, Health Care, Manufacturing, Smart cities.

UNIT-III: Overview of Computing Paradigm: 15 Hours

Grid Computing, Cluster Computing, Distributed Computing, Utility Computing and Cloud Computing, Evaluation to Cloud Computing. Introduction to Cloud Computing: Defining Cloud Computing. Cloud Types: The NIST Model, Cloud Cube Model, Deployment Models and Service Models. Characteristics of Cloud Computing: Paradigm Shift, Benefits of Cloud Computing, Disadvantages of Cloud Computing, Assessing the Role of Open Standards.

UNIT-IV: Cloud Computing Architecture: 10 Hours

Comparison with Traditional Computing Architecture (Client/Server), Cloud Computing Stack, Connecting to the Cloud. Introduction to Service Models: Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Software as a Service (SaaS), Defining Identity as a Service (IDaaS) and Defining Compliance as a Service (CaaS).

UNIT-V : Abstraction and Virtualization: 15 Hours

Virtualization and Cloud Computing, Types of Hardware Virtualization: Full, Partial and Para. Virtualization: Software Virtualization, Memory Virtualization, Storage Virtualization, Data Virtualization and Network Virtualization, Load Balancing, Abstraction Technique using Hypervisors, Machine Imaging. Capacity Planning: Defining Baseline and Metrics, Network Capacity, Scaling.

Text/Reference Books:

1. Cloud computing a practical approach - Anthony T.Velte , Toby J. Velte Robert Elsenpeter, TATA McGraw- Hill , New Delhi – 2010
2. Cloud Computing: Web-Based Applications That
3. Cathy O'Neil, Rachel Schutt, Doing Data Science, Straight Talk from The Frontline. O'Reilly, 2013.

4. Internet of Things (A Hands-on-Approach) , Vijay Madisetti , Arshdeep Bahga, University Press, First Edition, 2014.
5. Hassan, Qusay F., ed. Internet of things A to Z: technologies and applications. John Wiley & Sons, 2018.

*****

Name of the Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Minor 1 MCA-206	Advanced Java Programming Laboratory	00	00	02	00	00	50	50

List of Programs for Laboratory:

1. Write a Java Program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green. When a radio button is selected, the light is turned on, and only one light can be on at a time No light is on when the program starts.
2. Write a Java Program a simple user form which reads the name of a user and mail id in Text fields, select gender with radio buttons, and selects some Known languages using checkboxes, and also enters an address in a text area. After filling details whenever a user press the "submit" button, then displays all the information about the user input.
3. Write a Java Program to create multiple frames, which create a Frame2 with a 'back 'button, such that when a user click 'back' button, Frame 2 is closed and we see the Frame1 only?
4. Write a Java Program to create a frame using swing in which create a push button with a label and image. When the button is clicked an image is displayed in the Frame?
5. Write a Java Program to create a student table, which includes name, roll no, branch and age or DOB?
6. Write a Java Program to create a tabbed pane with two tabs. In the first tab sheet, display some push buttons with names of Branches. In second tab sheet, display checkboxes with names of subjects.
7. Write a java program to create a menu with several menu items by implementing JMenu.
8. Write a java program to create a combo box with some name of some places. The user can select any one name from the list and the selected country name is displayed in the frame? (Use JComboBox)
9. Write a java program to select multiple places and displayed in Frame using JList?
10. Write a java program to create a simple visual bean with a area filled with a color. The shape of the area depends on the property shape. If it is set to true then the shape of the area is Square and it is Circle, if it is false. The color of the area should be changed dynamically for every mouse click. The color should also be changed if we change the color in the "property window ".

11. Write a java program to create a bean that performs conversion of American dollar to Indian rupee.
12. Write a java program to create a bean that counts the number of buttons clicks?
13. Write a Java program that implements a simple client/server application. The client sends data to a server. The server receives the data, uses it to produce a result, and then sends the result back to the client. The client displays the result on the console. For ex: The data sent from the client is the radius of a circle, and the result produced by the server is the area of the circle. (Use java.net).
14. Installation of Apache Tomcat webserver.
15. Write a java Program to create a simple servlet and run it using tomcat server.
16. Write a java Program to create a servlet to read information from client Registration page?
17. Write a java Program to create a JSP page to display a simple message along with current Date?
18. Write a java Program to create a JSP page to display the random number?
19. Write a java Program to create a user request page in JSP?
20. Write a Hibernate program to create a database.

*****

Name of the Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
Minor 2: Vocational :MCA-206	Advanced RDBMS Laboratory	00	00	02	00	00	00	50

Course Description: This course provides a comprehensive understanding of database system design and implementation, covering key aspects such as physical and logical design, data modeling, relational models, and SQL for querying and managing data. Students will gain a deep understanding of essential DBMS concepts like transaction processing, data integrity, concurrency control, and recovery mechanisms. Through practical exercises, students will design and build a simple database system, demonstrating their ability to model, design, and implement a functional DBMS.

Course Learning Outcomes:

- On completion of this course, the student will be able to:
- Demonstrate an understanding of the relational data model.
- Performing PL/SQL programming using concept of Cursor Management, Error Handling, Triggers.
- Apply various normalization techniques.
- Formulate, using SQL, solutions to a broad range of query and data update problems.

- To develop appropriate Databases to a given problem that integrates ethical, social, legal, and economic concerns

List of experiments:

1. Implementation of DDL commands of SQL with suitable examples
    a. • Create table
    b. • Alter table
    c. • Drop Table
2. Implementation of DML commands of SQL with suitable examples
    a. • Insert
    b. • Update
    c. • Delete
3. Perform the following operation for demonstrating the insertion , updation and deletion using the referential integrity constraints
4. Implementation of different types of function with suitable examples:
5. Number function, Aggregate Function, Character Function, Conversion Function, Date Function
6. Implementation of different types of operators in SQL:
   Arithmetic Operators, Logical Operators, Comparison Operator, Special Operator, Set Operation
   Implementation of different types of Joins
    a. Inner Join
    b. Outer Join
    c. Natural Join etc..
   Implementation the following:
    d. Group By & having clause
    e. Order by clause
    f. Indexing
   Implementation the following:
    g. Sub queries
    h. Views
7. Study & Implementation of Database Backup & Recovery commands.
8. Study & Implementation of Rollback, Commit, Savepoint.
9. Implementation the following:
    a. Creating Database /Table Space
    b. Managing Users: Create User, Delete User
    c. Managing roles:-Grant, Revoke.
10. Perform the queries for triggers
11. Write the query for creating the users and their role.
12. Create table with attributes emp. No., emp. Name, Designation, Salary, and Department no. Construct for following queries:
    a. Display complete information of all the employees working as a manager.

b. Display name of all the employees working as a clerk.
c. Suppose DA for manager is 75% of salary then display name of all managers.
d. Select names and designation whose salary is greater then 15000.
e. Apply key constraints as Primary Key, Foreign Key etc as per requirement.
f. Between operation- list of all Employee Name & DOJ (date of joining) to join the Company in 2010

13. AND/OR operation- make a table that have an employee Perform AND/ORoperation.
14. Consider the following tables namely "DEPARTMENTS" and "EMPLOYEES" Their schemas are as follows, Departments (dept _no , dept_ name , dept_location); Employees (emp_id , emp_name , emp_salary,dept_no); and write SQL command to get the following:
    a. Develop a query to grant all privileges of employees table into departments table
    b. Develop a query to grant some privileges of employees table into departments table
    c. Develop a query to revoke all privileges of employees table from departments table
    d. Develop a query to revoke some privileges of employees table from departments table
    e. Write a query to implement the save point.
15. Group by function- Create the table for facilities having faculty-id, dept. no., designation name and group by similar dept.no. Facilities by using count function.
16. Create a table for emp. Using following data:- emp. name, emp age, emp salary, emp city & display the emp salary in ascending and descending order.
17. Max-Min function- create a table for student having similar attributes s_name, S_marks, s_id, s_sec&remark.
    a. Find the maximum marks obtained by student.
    b. Find the minimum marks obtained by student.
    c. Sum of all students marks using sum function.
    d. Find the average of marks using avg function.
18. Create forms and reports.
19. Practice on transaction processing.
20. Practice on functional dependencies.
21. Implement used defined procedures and functions using PL/SQL blocks.
22. Implement PL/SQL programmes using exception handling.
23. Implementation of different types of constraints.
24. Implement PL/SQL programmes using control structure.

Text/Reference Books:

1. Database system concepts, design and applications, S.K. Singh, Pearson Education, New Delhi.
2. Sql/PL/SQL, Batross, Ivan BPB.

****

MCA - First Year, III-Semester

MCA THIRD SEMESTER							
Name of Course	Name of course and code	Course Title	MM	L	T	P	C
Discipline Specific: Major 1	MCA-301	DSE-1(I/II/III/IV)	100	3	1	0	4
Discipline Specific: Major 2	MCA-302	DSE-2(I/II/III/IV)	100	3	1	0	4
Discipline Specific: Major 3	MCA-303	DSE-3(I/II/III/IV)	100	3	1	0	4
Discipline Specific: Major 4	MCA-304	Seminar/ Poster Presentation/Group Discussion	100	0	0	4	4
Discipline Specific: Major 5	MCA-305	Project Work-I	100	0	0	4	4
		Total Credit					20

DSE-1(I/II/III/IV), DSE-2 (I/II/III/IV) & DSE-3 (I/II/III/IV) details are given below

*****

MCA - First Year, IV-Semester

MCA FOURTH SEMESTER							
Name of Course	Name of course and code	Course Title	MM	L	T	P	C
Discipline	MCA-401	Project Work-II	500	0	0	20	20

Specific: Major 1		1. Project Synopsis	100	4	0	0	4
		2. Presentation of work	100	0	0	4	4
		3. Project Prototype	100	0	0	4	4
		4. Project Dissertation	100	4	0	0	4
		5. Viva-Voce	100	0	0	4	4
		Total Credit					20

*****

Master of Computer Applications (M.C.A.)-Semester Wise Curriculum Framework Under NEP 2020

One-Year MCA - First Year, I-Semester

MCA FIRST SEMESTER							
Name of Course	Name of course and code	Course Title	MM	L	T	P	C
Discipline Specific: Major 1*	MCA-101	DSE-1 (I/II/III/IV)	100	3	1	0	4
Discipline Specific: Major 2*	MCA-102	DSE-2 (I/II/III/IV)	100	3	1	0	4
Discipline Specific: Major 3*	MCA-103	DSE-3 (I/II/III/IV)	100	3	1	0	4
Discipline Specific: Major 4	MCA-104	Seminar/ Poster Presentation/Group Discussion	100	0	0	4	4
Discipline Specific: Major 5	MCA-105	Project Work-I	100	0	0	4	4
		Total Credits					20

* Pick subjects from Departmental Basket Pools

*****

MCA SECOND SEMESTER							
Name of Course	Name of course and code	Course Title	MM	L	T	P	C
Discipline	MCA-201	Project Work-II	500	0	0	20	20

Specific: Major 1		1. Project Synopsis	100	4	0	0	4
		2. Presentation of work	100	0	0	4	4
		3. Project Prototype	100	4	0	4	4
		4. Project Dissertation	100	4	0	0	4
		5. Viva-Voce	100	0	0	4	4
		Total Credits					20

*****

DSE Course Catalog for One- and Two-Year Postgraduate Programs

Discipline Specific Elective	Offered Subjects
DSE-I	1. Compiler Design 2. Computer Graphics and Multimedia Application 3. Soft Computing 4. Image Processing Techniques
DSE-II	1. Cyber Security and Cyber Forensics Management 2. Wireless Ad-hoc and Sensor Networks 3. Distributed and Parallel System 4. Simulation and Modeling
DSE-III	1. Software Security 2. Advanced Software Engineering and Project Management 3. Blockchain Computing 4. Intoduction to High Performance Computing

Discipline Specific Elective (DSE): details of Syllabus- offering- as below.

DSE-I

Name of the Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
MCA-101-DSE-1 -(I)	Compiler Design	03	01	00	20	20	60	100

Course Description: This course deals with the basic techniques of Compiler Construction and tools that can be used to perform Syntax-directed translation of a high-level programming language into an executable code. This will provide deeper insights into the more advanced semantics aspects of programming languages, code generation, machine independent optimizations, dynamic memory allocation, types and their inferences, object orientation.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Use compiler construction tools and describes the Functionality of each stage of compilation process
- Construct Grammars for Natural Languages and find the Syntactical Errors/Semantic errors during the compilations using parsing techniques
- Analyze different representations of intermediate code.
- Participate in GATE, UGC-NET and other competitive examinations.

UNIT-I: Compiler Design and Phases: **12 Hours**

Introduction to Compilers: Definition of compiler, interpreter and its differences, the phases of a compiler, phases of compilation process, Lexical analysis, Syntax analysis, Intermediate code generation, Code Optimization, Code Generation, Book keeping, Error handling.

UNIT-II: Finite Automata and lexical analysis: **12 Hours**

Role of the lexical analyzer, A simple approach to the design of lexical analyzers, Lexical errors, Input Buffering – Specification of Tokens, regular expressions, Finite Automata, Form regular expressions to finite automata, Minimizing the number of state of a DFA.

UNIT-III: Parsing Techniques and Algorithms: **12 Hours**

Parsing, Role of parser, Context-free grammars, Derivations and parse trees, Capabilities of context free grammars. Basic Parsing Techniques, Parsers, Shift-reduce parsing, Operator precedence parsing, Top-Down parsing, Predictive parsers, Automatic Construction of Efficient Parsers, LR parsers, canonical collection of LR(0) items, Constructing SLR parsing tables, Constructing canonical LR parsing tables, Constructing LALR parsing tables, Using Ambiguous grammars, An automatic parser generator, implementation of LR parsing tables, Constructing LALR sets of items.

UNIT-IV: Syntax-Directed Translation and Code: **12 Hours**

Syntax-directed Translation, Syntax-directed translation schemes implementation of syntax directed translators, Intermediate code, Postfix notation, Parse tree, Three address code: Addresses and Instructions, quadruples, and triples, Brief discussion on symbol tables and Contents of a Symbol Table, Run-time storage administration: Static Versus Dynamic Storage Allocation, Stack Allocation of Space.

UNIT-V: Code Optimization: **12 Hours**

Organization of code optimizer, basic blocks and flow graphs, optimization of basic blocks, the principal sources of optimization, the directed acyclic graph (DAG) representation of basic block, global data flow analysis. Peephole optimization.

Text/Reference Books:

1. Principals of Compiler Design By Alfred V. Aho, Jeffrey D. Ullman
2. Compilers-Principles, Techniques and Tools-A.V. Aho, R. Shethi and J.D.
3. Introduction to system software By D.M. Dhamdhere
4. Andrew W. Appel, Modern Compiler Implementation C, Cambridge University Press, UK, 2004.
5. Kenneth C. Louden, Compiler Construction– Principles and Practice, 1st edition, PWS Publishing, 1997.

*****

Name of the Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
MCA-DSE-1-(II)	Computer Graphics and Multimedia Applications	03	01	0	20	20	60	100

Course Description: Understand and apply advanced computer graphics concepts and techniques. Develop multimedia applications integrating graphics, audio, and video. Explore and implement modern tools and technologies in graphics and multimedia.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Understand Fundamental Concepts of computer graphics and multimedia.
- Apply 2D and 3D Graphics Techniques
- Develop Interactive Graphics Applications
- Implement Animation Techniques

UNIT-I: Introduction to Computer Graphics: 12 Hours

Overview of Computer Graphics: Historical context, applications, and trends. Graphics Systems : Video Display Devices, Raster Scan Systems, Random Scan Systems, Graphics Monitors and Work Stations, Input Devices, Hard Copy Devices, Graphics Software. Graphics Hardware and Software: Understanding GPUs, frame buffers, and graphics APIs.

UNIT-II: Fundamental Techniques in Graphics: 12 Hours

Line Drawing Algorithms: DDA Algorithm, Bresenham's Line algorithm, Circle Generating Algorithms: Midpoint Circle Algorithm. Filled-Area Primitives: Scan-line polygon fill algorithm, Inside-Outside Tests, boundary Fill Algorithm, Flood- Fill algorithm.

UNIT-III: 2D and 3D Transformations: 12 Hours

Basic Transformations- Translation, Rotation, Scaling. Matrix representations and Homogeneous Coordinates, Composite Transformations. Other Transformations: Reflection, Shearing. Two-Dimensional Viewing: The Viewing Pipeline, Clipping operations: Point clipping, Line Clipping: Cohen Sutherland line clipping, Polygon Clipping, Curve Clipping, Text Clipping, Exterior Clipping. Three-Dimensional Concepts**:** 3-D display methods: Parallel projection, Perspective projection, Depth cueing, Visible line and surface identification, Surface rendering.

UNIT-IV: Multimedia: Introduction to Multimedia: 12 Hours

Classification of Multimedia, Multimedia Software, Components of Multimedia – Audio: Analog to Digital conversion, sound card fundamentals, Audio play backing and recording Video, Text: Hypertext, Hyper media and Hyper Graphics, Graphics and Animation: Classification of Animation. Authoring Process and Tools.

UNIT-V: Multimedia and Internet: 12 Hours

Internet, HTML and web authoring, Multimedia considerations for Internet, Design consideration for web pages.

Text/Reference Books:

1. Computer Graphics C Version, Donald Hearn and M Pauline Baker, Pearson Education, 2nd edition, 2006
2. Introduction to Computer Graphics, J.D. Foley, A.V. Dam, Addison-Wesley Publishing Company, 2nd edition, 1994.
3. Ze-NianLi and Mark S. Drew, “Fundamentals of Multimedia”, First Edition, 2004, PHI Learning Pvt. Ltd., New Delhi.
4. Computer Graphics (Schaums Outline Series), R.A. Plastock et.al., TMH, 2nd edition, 2006
5. Computer Graphics, J.D. Foley, Pearson Education, 2nd edition, 2004

*****

Name of the Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
MCA-DSE-1-(III)	Soft Computing	03	01	00	20	20	60	100

Course Description: To comprehend input space partitioning and fuzzy modeling to investigate the ideas of neuro, fuzzy, and soft computing. To gain knowledge of Downhill Simplex Search, Random Search, Simulated Annealing, and Derivative-based Optimization, as well as their

application development. To get familiar about the use of supervised and unsupervised learning neural networks in computer science problem-solving. To become familiar with Neuro Fuzzy Spectrum, Framework Neuron Functions for Adaptive Networks, and Methods of ANFIS and RBFN. Acquire knowledge of the Genetic Algorithm and comprehend its Working Principle, Method, Flow Chart, Genetic Representations, Encoding, and Application.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Be aware of input space partitioning, fuzzy modeling, and the ideas of neuro, fuzzy, and soft computing.
- Recognize the notions of derivative-based optimization, downhill simplex search, random search, and simulated annealing, as well as how to construct applications using them.
- Recognize the principles of both supervised and unsupervised learning neural networks and how they are used to solve issues in computer science.
- Recognize the ideas behind Neuro Fuzzy Spectrum, Framework Neuron Functions for Adaptive Networks, Methods that Cross-Pollute ANFIS and RBFN, and Neuro Fuzzy Modeling.
- To comprehend the ideas behind genetic algorithms, including their working principle, method, flow chart, genetic representations, encoding, and application.

UNIT-I: Fuzzy Set Theory: **15 Hours**

Introduction to Neuro, Fuzzy and Soft Computing, Fuzzy Sets, Basic Definition and Terminology, Set-theoretic Operations, Member Function Formulation and Parameterization, Fuzzy Rules and Fuzzy Reasoning, Extension Principle, Fuzzy Relations, Fuzzy If-Then Rules, Fuzzy Inference Systems, Mamdani Fuzzy Models, Sugeno Fuzzy Models, Tsukamoto Fuzzy Models, Input Space Partitioning, Fuzzy Modeling.

UNIT-II: Optimization: **10 Hours**

Derivative-based Optimization, Descent Methods, Steepest Descent Method, Classical Newton's Method, Step Size Determination, Derivative-free Optimization, Simulated Annealing, Random Search, Downhill Simplex Search.

UNIT-III: Neural Networks: **10 Hours**

Supervised Learning Neural Networks, Perceptron's, Adaline, Back Propagation Multilayer Perceptron's, Radial Basis Function Networks, Unsupervised Learning Neural Networks, Competitive Learning Networks, Kohonen Self-Organizing Networks, Learning Vector Quantization, Hebbian Learning.

UNIT-IV: Neuro Fuzzy Modeling: **15 Hours**

Adaptive Neuro, Fuzzy Inference Systems, Architecture, Hybrid Neuro Fuzzy Modeling Learning Algorithm, Learning Methods that Cross-fertilize ANFIS and RBFN, Coactive Neuro Fuzzy Modeling, Framework Neuron Functions for Adaptive Networks, Neuro Fuzzy Spectrum.

UNIT-V: Genetic Algorithm: **10 Hours**

Fundamentals of Genetic Algorithms, Basic Concepts, Working Principle, Procedure of GA, Flow chart of GA, Genetic Representations, Encoding, Application of GA.

Text/Reference Books:

1. Davis E.Goldberg, "Genetic Algorithms: Search, Optimization and Machine Learning", Addison Wesley, N.Y., 1989.
2. S. Rajasekaran and G.A.V.Pai, "Neural Networks, Fuzzy Logic and Genetic Algorithms", PHI, 2003.
3. K. Mehrotra, Mohan, Ranka "Elements of Artificial Neural Networks", Penram International Publishing.
4. Timothy J.Ross, "Fuzzy Logic with Engineering Applications", McGraw-Hill, 1997
5. Fuzzy Logic: A Pratical approach, F. Martin, , Mc neill, and Ellen Thro, AP Professional, 2000.

*****

Name of Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
DSE-1-(IV)	Image Processing and Pattern Recognitions	03	01	00	20	20	60	100

Course Description: This course provides a comprehensive introduction to digital image processing, covering image acquisition, enhancement, restoration, compression, and analysis. Students will gain hands-on experience developing algorithms, preparing them for research and advanced applications.

Course Learning Outcomes:

- Master fundamental concepts of digital image representation, transformations, and operations.
- Apply spatial and frequency domain techniques for image enhancement and noise reduction.
- Implement and evaluate image compression algorithms, understanding trade-offs between size and quality.
- Utilize morphological operations and segmentation techniques for object detection and feature extraction.
- Gain familiarity with pattern recognition concepts and apply classification algorithms to image data.

UNIT-I: **Introduction to Digital Image and Image Restoration:** **12 Hours**

Introduction – Origin – Steps in Digital Image Processing – Components – Elements of Visual Perception – Image Sensing and Acquisition – Image Sampling and Quantization – Relationships Between Pixels, Elements of Digital Image Processing, Digital Image, File formats, Raster and Vector Form. Image Restoration, Importance, Basic Framework, Interactive Restoration, Image deformation and geometric transformations, image morphing, Restoration techniques, Noise models.

UNIT –II: **Fundamental of the Image Enhancement:** **12 Hours**

Introduction to Spatial Domain: Gray Level Transformations – Introduction to Histogram Processing, Smoothing and Sharpening Spatial Filtering. Fundamental of Frequency Domain: Introduction to Fourier Transform –with Basics Filters – Ideal, Butterworth And Gaussian Filters.

UNIT-III : **Image Compression and Processing:** **12 Hours**

Basics of the Image Compression, Redundancies, Image compression models, Error free Compression, lossy compression, Image compression standards. Morphological Image Processing and Image Segmentation: Basics, Structuring E, Erosion, Dilation, Opening, Closing, Hit-or-Miss Transform, Dimensionality Reduction: PCA, t-SNE, and LDA, Boundary Detection,thinning,Image Segmentation: Boundary detection based techniques. Trends in DIP: Introduction to Biometric System and Recognition, Biomedical and Geospatial image processing,Multi-scale Image Representation: Wavelets and Pyramid Methods.

UNIT- IV: **Essential of Pattern Recognitions:** **12 Hours**

Introduction to pattern recognition, features and feature vectors, concepts of learning: supervised, unsupervised and reinforced. Basic concepts of clustering and classification, classifiers: based on Bayesian decision theory, perceptron model, artificial neural networks, support vector machine, Object Detection and Segmentation: YOLO, SSD, and Mask R-CNN, nearest neighbours. Ensemble Methods: Bagging, Boosting, Random Fores, Principal component analysis and Linear Discriminant analysis, Expectation Maximization (EM), Hidden Markov Models.

UNIT-V: **Fundamentals of the Non-Parametric:** **12 Hours**

Density Estimation, Parzen window method, Probabilistic Neural Network (PNN), K- Nearest Neighbor, Estimation and rules, Nearest Neighbor and Fuzzy Classification, Linear Discriminant function-based Classifiers: Perception, Support Vector Machines (SVM).

Text/ Reference Books:

1. Rafael C. Gonzalez, Richard E. Woods, Steven L. Eddins, “Digital Image Processing Using MATLAB”, Third Edition Tata Mc Graw Hill Pvt. Ltd., 2011.

2. Anil Jain K. “Fundamentals Of Digital Image Processing”, PHI Learning Pvt. Ltd., 2011.
3. Digital Image Processing by Bhabatosh Chanda and Dwijesh Majumder, PHI
4. Digital Image Processing, Second Edition by Rafel C. Gonzalez and Richard E. Woods, Pearson Education
5. William K Pratt, "Digital Image Processing", John Willey 2001
6. Millman Sonka, Vaclav Hlavac, Roger Boyle, Broos/Colic, "Image Processing Analysis and Machine Vision" - Thompson Learning, 1999.
7. Chanda S., Dutta Majumdar - "Digital Image Processing and Applications", Prentice Hall of India, 2000.

*****

DSE-II

Course Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
MCA - - DSE-II-1	Cyber Security and Cyber Forensics Management	03	01	00	20	20	60	100

Course Description: This course will cover the essential aspects of cybersecurity, from basic terminology to the latest threats and defense strategies. Students will gain a comprehensive understanding of cyber warfare, cybercrime, and the legal frameworks in place to combat them. Additionally, the course will delve into data privacy and security, including the Data Protection Bill and international regulations. Finally, students will be introduced to the fundamentals of cyber forensics, equipping them with the knowledge to investigate and analyze cybercrime.

Course Learning Outcomes: On completion of this course, the student will be able to:

- They will also develop understanding about the Cyberwarfare and necessity to strengthen the cyber security of end user machine, critical IT and national critical infrastructure.
- They will also develop understanding about the type and nature of cyber crimes and as to how report these crimes through the prescribed legal and Government channels.
- Understand and utilize fundamental cybersecurity terminology.
- Analyze the current cybersecurity threat landscape.
- Explain the concepts of cyber warfare and its implications.
- Identify various types of cybercrimes and understand their impact.
- Comprehend the legal frameworks surrounding cybercrime, including the IT Act, 2000 and its limitations.
- Demonstrate knowledge of data privacy and security principles, including the Data Protection Bill and GDPR.
- Grasp the basics of cyber forensics, including incident response, evidence collection, and analysis techniques.

UNIT- I: Overview of Cyber security: **12 Hours**

Cyber security increasing threat landscape, Cyber security terminologies- Cyberspace, attack, attack vector, attack surface, threat, risk, vulnerability, exploit, exploitation, hacker, Non-state actors, Cyber terrorism, Protection of end user machine, Critical IT and National Critical Infrastructure, Cyberwarfare, Case Studies,Cryptographic Protocols: SSL/TLS, IPsec, and VPNs.

UNIT-II: Cyber Crimes and Scams: **12 Hours**

Cyber crimes : Cyber crimes targeting Computer systems and Mobiles- data diddling attacks, spyware, logic bombs, DoS, DDoS, APTs, virus, Trojans, ransomware, data breach. Online scams and frauds- email scams, Phishing, Vishing, Smishing, Online job fraud, Online sextortion, Debit/ credit card fraud, Online payment fraud, Cyberbullying, website defacement, Cyber- squatting, Pharming, Cyber espionage, Cryptojacking, Darknet- illegal trades, drug trafficking, human trafficking. Social Media Scams & Frauds- impersonation, identity theft, job scams, misinformation, fake newscyber crime against persons - cyber grooming, child pornography, cyber stalking. Social Engineering attacks, Cyber Police stations, Crime reporting procedure, Case studies. After completion of the module, students will have complete understanding of the cyber- attacks that target computers, mobiles and persons.

UNIT-III: Cyber Law and Data Security: **12 Hours**

Cyber Law Cyber crime and legal landscape around the world, IT Act,2000 and its amendments. Limitations of IT Act, 2000. Cyber crime and punishments, Cyber Laws and Legal and ethical aspects related to new technologies- AI/ML, IoT, Blockchain, Darknet and Social media, Cyber Laws of other countries, Case Studies. Students after completing this module will be able to understand the legal framework that exist in India for cyber crimes and penalties and punishments for such crimes, It will also expose students to limitations of existing IT Act,2000 legal framework that is followed in other countries and legal and ethical aspects related to new technologies.

Data Privacy and Data Security Defining data, meta-data, big data, non- personal data. Data protection, Data privacy and data security, Personal Data Protection Bill and its compliance, Data protection principles, Big data security issues and challenges, Data protection regulations of other countries- General Data Protection Regulations(GDPR),2016 Personal Information Protection and Electronic Documents Act (PIPEDA), Social media- data privacy and security issues.

UNIT-IV: **Fundamentals of Cyber Forensics:** **12 Hours**

What is Cyber? Concepts of Cyber Forensics, Need and Application, Role of Cyber Forensics Experts, Cyber Analysis, The Standard Method of true Cyber Forensics, Concept of Chain of custody, SoP of Cyber Forensics process, Incident Response and Handling, Where to look evidence, determining what data to collect and analyze.

Cyber Crime and Need of Investigation

Definition, Crimes on Internet, Hacking, Virus, Worms, Cookies, Obscenity and Pornography, Program Manipulation, Software Piracy and Intellectual Property, Concept of Network Security and Cyber-Crime Investigation.Digital Audits, Open Sauces tools and techniques for Cyber Forensic investigation, E-Mail header and Email Investigation by using online mode.

UNIT-V: Cyber security Management: 12 Hours

Compliance and Governance Cyber security Plan- cyber security policy, cyber crises management plan, Business continuity, Risk assessment, Types of security controls and their goals, Cyber security audit and compliance, National cyber security policy and strategy, Digital Evidence Collection: Forensic Imaging, Chain of Custody, and Evidence Preservation, Forensic Tools and Software: EnCase, FTK, and Volatility.

Text/Reference Books:

1. Cyber Security Understanding Cyber Crimes, Computer Forensics and Legal Perspectives by Sumit Belapure and Nina Godbole, Wiley India Pvt. Ltd.
2. Information Warfare and Security by Dorothy F. Denning, Addison Wesley.
3. Security in the Digital Age: Social Media Security Threats and Vulnerabilities by Henry A. Oliver,
4. Albert J. M. & Guillossou F. (2012), Cyber Forensics: From Data to Digital Evidence. New Jersey, Wiley Corporate F&A.
5. Casey E. (2009). Handbook of Digital Forensics and Investigation. USA, Academic Press.
6. Marcella A. & Menendez D. (2007). Cyber Forensics: A Field Manual for Collecting, Examining, and Preserving Evidence of Computer Crimes. NY, CRC Press.

*****

Name of the Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
MCA-DSE-II-2	Wireless Ad hoc and Sensor Networks	03	01	00	20	20	60	100

Course Description: To understand the concepts of sensor networks. To understand the MAC and transport protocols for ad hoc networks. Students can understand the applications of adhoc and sensor networks.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Ability to understand the state-of-the-art research in the emerging subject of Ad Hoc and Wireless Sensor Networks
- Ability to solve the issues in real-time application development based on ASN.

- Ability to conduct further research in the domain of ASN

UNIT-I: Introduction to Ad Hoc Networks: 12 Hours
Characteristics of MANETs, Applications of MANETs and Challenges of MANETs. Routing in MANETs - Criteria for classification, Taxonomy of MANET routing algorithms, Topologybased routing algorithms-Proactive: DSDV; Reactive: DSR, AODV; Hybrid: ZRP; Position-based routing algorithms-Location Services-DREAM, Quorum-based; Forwarding Strategies: Greedy Packet, Restricted Directional Flooding-DREAM, LAR.

UNIT-II: Data Transmission and Multicasting: 12 Hours
Data Transmission:Broadcast Storm Problem, Rebroadcasting Schemes-Simple-flooding, Probability-based Methods, Area-based Methods, Neighbor Knowledge-based: SBA, Multipoint Relaying, AHBP. Multicasting: Tree-based: AMRIS, MAODV; Mesh-based: ODMRP, CAMP; Hybrid: AMRoute, MCEDAR.

UNIT-III: Geocasting and TCP Protocols: 12 Hours
Geocasting: Data-transmission Oriented-LBM; Route Creation Oriented-GeoTORA, MGR. TCP over Ad Hoc TCP protocol overview, TCP and MANETs, Solutions for TCP over Ad hoc.

UNIT-IV: Wireless Sensor Networks Basics: 12 Hours
Basics of Wireless, Sensors and Lower Layer Issues: Applications, Classification of sensor networks, Architecture of sensor network, Physical layer, MAC layer, Link layer, Routing Layer.

UNIT-V: Upper Layer Issues of WSN: 12 Hours
Transport layer, High-level application layer support, Adapting to the inherent dynamic nature of WSNs, Sensor Networks and mobile robots.

Text/References Books:

1. Ad Hoc and Sensor Networks – Theory and Applications, Carlos Corderio Dharma P. Aggarwal, World Scientific Publications, March 2006, ISBN – 981–256–681–3.
2. Wireless Sensor Networks: An Information Processing Approach, Feng Zhao, Leonidas Guibas, Elsevier Science, ISBN – 978-1-55860-914-3 (Morgan Kauffman).
3. K. Akkaya and M. Younis, “A survey of routing protocols in wireless sensor networks”, Elsevier Ad Hoc Network Journal, Vol. 3, no. 3, pp. 325--349
4. Philip Levis, “ TinyOS Programming”
5. Anna Ha´c, “Wireless Sensor Network Designs”, John Wiley & Sons Ltd,

*****

Name of the Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		

MCA-DSE-II-3	Distributed and Parallel System	03	01	0	20	20	60	100

Course Description: Understand the principles and architecture of distributed and parallel systems. Learn techniques for designing and implementing scalable, efficient, and fault-tolerant systems. Explore various models and tools for parallel and distributed computing.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Understand Fundamental Concepts of Distributed and Parallel system
- Understand the design choices, challenges, and solutions.
- Design and Implementation of parallel and distributed system
- Identify and resolve issues in parallel and distributed systems

UNIT-I: Introduction to Parallel and Distributed Computing: 12 Hours

Definitions and Concepts: Understanding distributed systems and parallel systems., Historical, Context and Applications: Evolution of distributed and parallel computing., System Architectures: Comparison of distributed and parallel system architectures.

UNIT-II: Parallel Computing Fundamentals: 12 Hours

Principles of Parallel Algorithm Design, Parallel Computing Models: Shared memory, distributed memory, and hybrid models. Parallel Architectures: Multi-core processors, SIMD, MIMD, and clusters. Concurrency and Synchronization: Threads, process synchronization, and inter-process communication. Parallel Algorithms: Design and analysis of parallel algorithms, speedup, and scalability, Pipelining and Data Clustering.

UNIT-III: Parallel Programming Model: 12 Hours

Flynn's Taxonomy, PRAM, EREW, CREW, ERCW, CRCW, Simulating CRCW, CREW & EREW, PRAM algorithms, Concepts of Threads, Blocks, Grids, Developing a kernel function to be executed by individual threads.

UNIT-IV: Distributed Systems Fundamentals: 12 Hours

Distributed System Models: Client-server, peer-to-peer, and microservices. Communication Models: Sockets, RPC, RMI, and message passing. Coordination and Synchronization: Distributed clocks, consensus algorithms (e.g., Paxos, Raft), and mutual exclusion. Fault Tolerance and Reliability: Redundancy, replication, and recovery techniques.

UNIT-V: Distributed Shared Memory & File Systems: 12 Hours

Hardware DSM, Design issues in DSM Systems, Implementation issues, Heterogeneous and Other DSM Systems, Distributed File Systems, File Service Architecture.

Text/Reference Books:

1. A Grama, A Gupra, G Karypis, V Kumar. Introduction to Parallel Computing (2nd ed.). Addison Wesley, 2003.
2. C Lin, L Snyder. Principles of Parallel Programming. USA: Addison-Wesley Publishing Company, 2008.
3. J Jeffers, J Reinders. Intel Xeon Phi Coprocessor High-Performance Programming. Morgan Kaufmann Publishing and Elsevier, 2013.
4. T Mattson, B Sanders, B Massingill. Patterns for Parallel Programming. AddisonWesley Professional, 2004.
5. Distributed Systems, Principles and Paradigms, Andrew S. Tanenbaum, Maarten Van Steen, 2nd Edition, PHI.
6. Distributed Systems, An Algorithm Approach, Sukumar Ghosh, Chapman&Hall/CRC, Taylor & Fransis Group, 2007.

*****

Name of the Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
MCA-DSE-II-4	Simulation and Modeling	03	01	00	20	20	60	100

Course Description: Discuss system and simulation models. Analyzing different probability distribution functions. Compare simulation models and comprehend their differences. Analyze simulation models using input and output analyzers. Verify, validate, and analyze simulation model outputs.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Explain the key components of discrete event simulation and modeling paradigm.
- Conceptualize real-world conditions for systems development decisions based on requirements and goals.
- Develop simulation models using randomly generated and tested variables.
- Develop methods for simulating discrete systems with queuing systems.
- Learn how to use simulation tools to create and execute goal-driven system models.

UNIT-I: System Definition and Components: **12 Hours**

System Definition and Components, Stochastic Activities, Continuous and Discrete Systems, System Modeling, Types of Models, Static and Dynamic Physical Models, Static and Dynamic Mathematical Models, Full Corporate Model, Types of System Study.

UNIT-II: System Simulation: 12 Hours

System Simulation, Nature And Techniques of Simulation, Comparison of Simulation and Analytical Methods, Types of System Simulation, Real Time Simulation, Hybrid Simulation, Simulation of Pure-Pursuit Problem, Single-Server Queuing System and an Inventory Problem, Monte-Carlo Simulation, Distributed Lag Models, Cobweb Model, Simulation of Single Server Queuing system and Monte Carlo Simulation

UNIT-III: Simulation of Continuous Systems: 12 Hours

Simulation of Continuous Systems, Analog Vs. Digital Simulation, Simulation of Water Reservoir System, Simulation of a Servo System, Simulation of an Autopilot, Discrete System Simulation, Fixed Time-Step Vs. Even to Even Model, Generation of Random Numbers, Test for Randomness, Monte-Carlo Computation Vs. Stochastic Simulation. Demonstration of generation of Random Number through MATLAB

UNIT-IV: System Dynamics: 12 Hours

System Dynamics, Exponential Growth Models, Exponential Decay Models, Modified Exponential Growth Models, Logistic Curves, Generalization of Growth Models, System Dynamic Diagrams. Introduction to SIMSCRIPT: Program, System Concepts, Origination and Statements, Defining the Telephone System Model.

UNIT-V: Simulation of PERT Networks: 12 Hours

Simulation of PERT Networks, Critical Path Computation, Uncertainties in Activity Duration, Resource Allocation and Consideration, Simulation Languages and Software, Continuous and Discrete Simulation Languages, Expression Based Languages, Object Oriented Simulation, General Purpose Vs. Application Oriented Simulation Packages, CSMP-III, MODSIM-III.

Text/Reference Books:

1. Geoffrey Gordon, “System Simulation”, PHI.
2. Jerry Banks, John S. C Barry L. Nelson David M. Nicol, “Discrete Event System Simulation”, Pearson Education.
3. V P Singh, “System Modeling and simulation”, New Age International.
4. Averill M. Law, W. David Kelton, “System Modeling and Simulation and Analysis”, TMH.
5. Zeigler B.P. Praehofer. H. and Kim I.G. "Theory of modeling and simulation", 2 nd Edition. Academic press 2000

*****

DSE-III

Name of the Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
MCA-DSE-III-1	Software Security	03	01	00	20	20	60	100

Course Description: To presents the implementation of a unique socio-technical solution for real-time security awareness. To provides comprehensible knowledge about security, risk, protection, estimation, knowledge, and governance. To Various emerging standards, models, metrics, continuous updates, and tools are described to understand security principals and mitigation mechanisms for higher security. To explore common vulnerabilities plaguing today's web applications.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Capable of understanding the notion of software security and risk assessment
- Acquired knowledge of diverse security threats and metrics.
- Acquired knowledge of risk assessment and Secure Software Architecture.
- Capable of understanding security testing and how case studies aid in estimating security risk.

UNIT-I: Software and Security Concepts: 10 Hours

Software & Security, Security Overview, Security Components, Characteristics, Security Types, Security Myths, Software Security Assurance, Software Security Models and Metrics, Software Security Problems, Risk Management Framework

UNIT-II: Threat to Security: 10 Hours

Threats, Security Threats, Security Threats Classification, Threat Impact Analysis, Protection & Mitigation Strategies, Software Security Metrics, Security Metrics Development Framework.

UNIT-III: Software Security Estimation: 15 Hours

Software Security Estimation, Risk Estimation, Vulnerability Assessment, Vulnerability Assessment Framework, Security Profiling, Operation Ability

UNIT-IV: Secure Software Architecture & Security Assurance: 10 Hours

Software Architecture, Security Architecture and Models, Security Architecture Process, Software Security Best Practices, Software Security Assurance, Security Assurance Framework, Microsoft Secure Development Life Cycle, OWSAP Software Assurance Maturity Model

UNIT-V: Software Security Testing & Knowledge Management: 15 Hours

Software Testing, Security Testing, Security Testing Process, Implementing Security Testing: A Case Study, Implementing Security: A Case Study, Secure Knowledge Management, Security Governance, E-governance Framework in India, Digital India Initiatives

Text/Reference Books:

1. Suhel Ahmad Khan, Software Security: Concepts & Practices, CRC Press, 2023
2. Gary Mcgraw, Software Security: Building Security In, Addison Wesley Professional, 2006
3. Matt Bishop, Computer Security: Art & Science, Addison Wesley Professional, 2002
4. Julia Allen, Gary Macgraw, Software Security Engineering: A Guide for Project Managers, Addison Wesley Professional, 2008
5. Adam Shostack, Threat Modeling: Designing for Security, Wiley, 2014 (ISBN-13: 978-1118809990)
6. William Stallings, Lawrie Brown, Computer Security: Principles and Practice, 3rd Edition, Pearson, 2014 (ISBN-13: 978-0133773927)

*****

Name of the Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
MCA-DSE-III-2	Advanced Software Engineering and Project Management	03	01	00	20	20	60	100

Course Description: This course introduces the concepts and methods required for the construction of large software intensive systems. It aims to set these techniques in an appropriate engineering and management context. The course aims is to develop a broad understanding of the discipline of software engineering and management of software systems. This will provide he detailed knowledge regarding the concepts of OOPS, various testing techniques, cost analysis and software reliability and quality assurance.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Basic knowledge of various processes to be followed in the software development life-cycle models.
- Implement communication, modeling, construction and deployment practices in software development.
- Analyze & design the software models using the concepts of OOPs.
- Explain the concepts of various software testing methods & be able to apply appropriate testing approaches for development of software.

- Explain the project management & different types of metrics used in software development and risk and cost benefit analysis is implemented.
- Apply the concepts of project management & planning.
- Apply the concept of software reliability and quality assurance is implemented.

UNIT-I: Basics of Software Engineering: 15 Hours

Review of software engineering, Key concepts, Classical SDLC model of software engineering, COTS, RUP, Rapid Application Development, Agile methods, Extreme programming. **Advanced concepts of software engineering**: Cleanroom software engineering, Component based software engineering, Aspect oriented software engineering, Service oriented software engineering, Object oriented software engineering. **Software project management**: Introduction to software project management, Project managers roles and responsibilities, Project planning.

UNIT –II: Requirement Engineering: 15 Hours

Introduction to requirement engineering, Importance and challenges in requirement engineering, Success factors of requirement engineering, Requirement engineering framework, Characteristics of requirements, Issues and problems in requirement elicitation, Trends and challenges of requirement elicitation, Techniques and approaches of requirement elicitation, Role of soft skills in requirement analysis, Requirement reuse, Requirement verification & validation. **S**pecification of requirement models: Characteristics of SRS, Components of SRS, Specification language, Structure of requirement document, Guidelines for writing SRS. Requirement Prioritization: Aspects of prioritization, Prioritization techniques, Requirement negotiation, Ambiguity in requirement engineering.

UNIT-III: Software Design: 10 Hours

Design principles, Software design process, Design qualities, Design patterns. Design practices: Incremental design, Structured analysis and structured design, Jacksons structured programming, Introduction to object oriented design. Software visualization: Introduction and challenges, Static visualization, Dynamic program visualization, Evaluating visualizations, Data visualization. Coding guidelines and principles, Refactoring

UNIT –IV: Software Testing: 10 Hours

Terminology, Life cycle, Methodology, Types of testing, Test planning, Static testing, Black box testing, White box testing and their types, Model for software testing, Unit testing, Integration testing, System and Acceptance Testing. Object oriented testing: Object oriented testing model, Object oriented software test strategy. Software security and risk management: Fundamentals of software security and risk management, Reactive vs. proactive risk strategies, Software risks, Risk identification, Risk projection, Risk refinement, RMMM and RMMM Plan.

UNIT- V: Introduction to Software Quality: 10 Hours

Concepts of quality, perspectives and expectations, Quality Framework. Quality assurance: Classification, Quality assurance activities and techniques. Quality models: McCall's model,

Bohem's model, Dromey's model, FURPS Model, ISO-9126 model, Cost of quality, Software quality factors, Quality control, CMMI framework. Reengineering: Business process reengineering, Software reengineering, Reverse engineering, Restructuring, Forward reengineering.

Text/Reference Books:

1. Software Engineering - A practitioner's Approach', Roger S. Pressman, 7 editions, McGraw-Hill International Edition.
2. Software Security: Concepts & Practices, Suhel Ahmad Khan, Chapman & Hall, CRC Press, 9781032356310, 2023
3. Software Engineering,, Sommerville, 9th edition, Addison-Wesley Publication.
4. Software Engineering, K.K. Agarwal & Yogesh Singh, New Age International Publication.
5. Software Engineering: An Engineering approach, James F. Peters, Witold Pedrycz, Wiley Publication.

*****

Name of the Course and Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
MCA-DSE -III-3	Blockchain Computing	03	01	00	20	20	60	100

Course Description: To give knowledge and understanding of cryptography and blockchain. Learn about Bitcoin and cryptocurrencies, as well as their influence on blockchain. Learn about Ethereum and Hyperledger technology and its numerous components. Provide an overview of Solidity programming and Smart Contracts. To develop an understanding about the various applications of Blockchain.

Course Learning Outcomes: On completion of this course, the student will be able to:

- Improved ability to handle cryptographic challenges and implement them effectively in several fields.
- Understand the implementation of Ethereum Virtual Machine and Bitcoin Wallets, including transactional blocks.
- Understanding and implementing the Consensus Mechanism and Hyperledger Composer.
- Enables testing of Solidity Programming and Structure of Smart Contracts, as well as their performance.
- Understand Blockchain uses and future developments.

UNIT-I: Introduction to Cryptography and Blockchain: 12 Hours

Blockchain, Blockchain Technology, Mechanisms and Networks, Blockchain Origins, Objective of Blockchain, Blockchain Challenges, Transactions and Blocks, P2P Systems, Keys as Identity, Digital Signatures, Hashing, Public Key Cryptosystems, Private vs. Public Blockchain.

UNIT-II: Introduction to BitCoin and Cryptocurrency: 12 Hours

Bitcoin, Bitcoin Network, Bitcoin Mining Process, Mining Developments, Bitcoin Wallets, Decentralization and Hard Forks, Ethereum Virtual Machine (EVM), Merkle Tree, Double-Spend Problem, Blockchain and Digital Currency, Transactional Blocks, Impact of Blockchain Technology on Cryptocurrency

UNIT-III: Ethereum, Hyperledger: 12 Hours

Introduction, Consensus Mechanisms, Working of Smart Contracts, Metamask Setup, Ethereum Accounts, Receiving Ether's. Introduction, Distributed Ledger Technology & its Challenges, Hyperledger and Distributed Ledger Technology, Hyperledger Fabric, Hyperledger Composer.

UNIT-IV: Solidity Programming: 12 Hours

Solidity - Language of Smart Contracts, Installing Solidity and Ethereum Wallet, Basics of Solidity, Layout of a Solidity Source File, Structure of Smart Contracts, General Value Types (Int, Real, String, Bytes, Arrays, Mapping, Enum, Address.

UNIT-V: Blockchain Applications: 12 Hours

Internet of Things, Medical Record Management System, Domain Name Service and Future of Blockchain, Alt Coins.

Text/Reference Books:

1. Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller and Steven Goldfeder, Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction, Princeton University Press (July 19, 2016).
2. Merunas Grincalaitis, "Mastering Ethereum: Implement Advanced Blockchain Applications Using Ethereum-supported Tools, Services, and Protocols", Packt Publishing.
3. D. Drescher, Blockchain Basics. Apress, 2017.
4. Josh Thompson, 'Blockchain: The Blockchain for Beginnings, Guild to Blockchain Technology and Blockchain Programming', Create Space Independent Publishing Platform, 2017.
5. Imran Bashir, "Mastering Blockchain: Distributed ledger technology, decentralization, and smart contracts explained", Packt Publishing.

*****

Course Code	Course Title	L	T	P	Sessional		ESM	Total
					MSE	IA		
MCA-403 DSE-(4)	Introduction to High Performance Computing	03	01	00	20	20	60	100

Course Desriptions: This course provides a comprehensive introduction to the rapidly evolving fields of quantum computing and high-performance computing. Students will explore the fundamental principles of quantum mechanics that underpin quantum computation, learn to design and analyze quantum circuits, and delve into the architectures and programming paradigms of high-performance computing systems. The course will cover essential linear algebra concepts, quantum algorithms, classical optimization techniques, parallel computing models, and hybrid architectures, equipping students with the theoretical knowledge and practical skills to tackle computationally challenging problems in various domains.

Course Learning Outcomes: Upon successful completion of this course, students will be able to:

Quantum Computing:

- Understand the fundamental concepts of quantum mechanics relevant to quantum computing, including superposition, entanglement, and quantum measurement.
- Represent quantum states and operations using linear algebra and the Bloch sphere representation.
- Design and analyze basic quantum circuits using fundamental quantum gates.
- Explain the principles of quantum algorithms and their potential advantages over classical algorithms.

High-Performance Computing:

- Describe the evolution of computing architectures and the motivation behind high-performance computing.
- Analyze program performance using relevant metrics and identify performance bottlenecks.
- Apply optimization techniques for serial code, including loop optimization and memory access optimization.
- Differentiate between shared memory and distributed memory architectures and their respective programming models.
- Understand the principles of hybrid architectures and the use of accelerators like GPUs for high-performance computing.

UNIT- I: Foundations of High-Performance Computing: 12 Hours

Introduction: Motivation for high-performance computing, Evolution of computing architectures, Performance metrics and evaluation. Modern Processor Architectures: Instruction-level parallelism: Pipelining, superscalar execution, Memory hierarchy and cache optimization, Multicore processors and challenges of concurrency.

UNIT- II:Parallel Algorithm Design & Performance-1 12 Hours

One to all, All to All, All Reduce, Prefix Sum, Scatter Gather, All to All personalized communications, Analytical Modelling of Parallel Programs 6 Sources of overhead, performance matrix, Effect of granularity on performance

UNIT- III: Parallel Algorithm Design & Modelling 12 Hours

Analytical Modelling of Parallel Programs Scalability, Minimum execution time, cost optimal execution time, Dense matrix algorithms Matrix Vector, Matrix-Matrix multiplications, Solving linear equations, Sorting Issues, sorting network, Bubble sort, Graph Algorithms MST, SSSP, APSP.

UNIT- IV: Optimizing for Serial Performance: 12 Hours

Basic Optimization Techniques: Code profiling and bottleneck identification, Loop optimization: Unrolling, fusion, tiling, Data structures and memory access patterns. Compiler Optimizations: Understanding compiler optimization levels, Code restructuring for improved compiler analysis, Using compiler intrinsics for performance-critical operations.

UNIT- V: Parallel Computing Paradigms: 12 Hours

Taxonomy of Parallelism, Data parallelism vs. function parallelism, Shared memory vs. distributed memory models, Implicit vs. explicit parallelism. Shared Memory Architectures: Cache coherence and synchronization mechanisms, Programming models: Threads, OpenMP, Distributed Memory Architectures: Message passing and communication overheads, Programming models: MPI. Hybrid Architectures and Programming Model, Combining shared and distributed memory paradigms,Heterogeneous computing with accelerators (e.g., GPUs).

Text/Reference Books:

1. Quantum Computing Fundamentals, Chuck Easttom, 9356062595, Pearson Education, 2022
2. Michael A. Nielsen, Issac L. Chuang, “Quantum Computation and Quantum Information”, Tenth Edition, Cambridge University Press, 2010.
3. Scott Aaronson, “Quantum Computing Since Democritus”, Cambridge University Press, 2013.
4. N. David Mermin, “Quantum Computer Science: An Introduction”, Cambridge University Press, 2007.

5. High Performance Cluster Computing, Volume 1, Architecture and Systems, Rajkumar Buyya, Pearson Education.

Departmental Basket Pools:

DSM-1- MCA/BCA

1. Overview Programming Concepts
2. Full Stack Web Development -I

DSM- 2:MCA/BCA

1. Machine Learning
2. Artificial Intelligence and Automation

DSM- 3:MCA/BCA

1. Advanced-Data Structures and Algorithms
2. Knowledge Management in IT

DSM- 4:MCA/BCA

1. Digital Logic and Computer Design
2. Data Science and Big Data Analytics

—-------

- BCA 7th Semester extends to MCA 1st Semester of 2 years Programme
- BCA 8th Semester extends to MCA 2nd Semester of 2 years Programme

DSM-1: MCA/BCA

1. Advanced Java Programming
2. Full Stack Web Development -II

DSM- 2: MCA/BCA

1. Advanced Database Management System
2. Computer Forensics and Corporate Security

DSM- 3: MCA/BCA

1. Formal Languages and Automata Theory
2. Fundamental of UNIX and LINUX

DSM- 4: MCA/BCA

1. IoT and Cloud Computing
2. OOP using Python Programming

*****

Doctor of Philosophy

PhD Course Work

1. Introduction

The Ph.D programme in Computer Science imparts the knowledge, skills and attitude to do world class research in the area of computer science. Doctorate students are expected to publish their work in leading international journals and conferences pertaining to their area of research. Research programmes leading to Ph.D. degrees are the main thrust in the University. These programmes attract students from all corners of the country, in addition to some international students.

The research programmes have a limited amount of course work, essentially to prepare the student to carry out research, and to develop adequate breadth in the subject area. The main emphasis is on the research work leading to a thesis. This involves delving in depth into a research topic and includes grasping, evaluating and refining ideas, mathematical rigor, problem formulation, literature survey, implementation (programming) and performance evaluation. Typically students decide on their research area and faculty advisor within the first year while doing course work. Thereafter, the steps involved are finishing course work (to build research background and breadth), thesis proposal defense, and thesis defense. Upon clearing the requirements and the acceptance of their thesis proposal, the student works on the thesis, and its successful defense leads to the degree.

Course Duration

Flexible (6 semesters minimum; 14 semesters maximum). Typical duration varies between 4-6 years.

Eligibility

PG degree in Computer Science or equivalent as M.C.A./M.Sc./M,Tech,/M.E.

2. Structure of the Curriculum

2.1. Foundation Courses

1. Research Methodology
2. Research and Publication Ethics
3. Foundation of Computer Science

2.2. Elective Courses

Select any two Electives from the following.

1. Advance Software Engineering
2. Software Reliability
3. Software Testing and Quality Assurance
4. Wireless Sensor Network
5. Fuzzy Logic and Artificial Neural Network
6. Ad-Hoc and Wireless Network
7. Wireless Technologies for IoT
8. Software Architecture and Project Management
9. Information Security
10. Information Hiding
11. Image Processing Techniques
12. Machine Learning
13. Trends in Digital Image Processing and Pattern Learning
14. Advanced Research in Cyber Security
15. Research in Computer Forensics
16. Security in computing
17. Advance Topic in Artificial Intelligence
18. Advanced Theory of Comuteronics
19. Advanced Cloud Computing and IoT
20. Advanced Computer Networks

More Elective Papers may be added from time to time depending upon the availability of the Expertise in the Department and its suitability for the prospective researchers.

Structure of the Courses

	SEMESTER-I				
Subject Code	**Subject Name**	**Credit L+P+T**	**Internal Marks**	**External Marks**	**Total Marks**
RCS-101	Research Methodology	4	40	60	100
RCS-102	Research and Publication Ethics	2	40	60	100
RCS-103	Foundation of Computer Science	4	40	60	100
RCS-104	Discipline Specific Elective (DSE) Paper -I	3	40	60	100
RCS-105	Discipline Specific Elective (DSE) Paper -II	3	40	60	100

	Total	**16**	**160**	**240**	**400**

RCS- 101
RESEARCH METHODOLOGY

UNIT-I: The literature review: **10 Hours**

Importance of the literature review, Planning a literature search, Locating relevant literature, Reliability of a source, Note making, Writing a survey and identifying the problem. Planning and writing a research proposal: Research projects, Finding a research problem, and Analysis of research ideas.

UNIT-II: Planning a proposal: **10 Hours**

Some general considerations, Proposal outline, Presentation and evaluation of proposals, Students' outline of research work, Major funding agencies.

Presentations and lectures: Scientific meetings, General features of a Scientific conference, Planning and delivering a presentation, Preparing good presentation slides, Poster presentation in meeting, Students' seminar, Group and panel discussion, Academic lectures.

UNIT-III: Collection and analysis of data: **15 Hours**

Designing experiments for the purpose of testing research hypotheses, and evaluating the results of those experiments, Analysis of data, Descriptive statistics, Inferential Statistics, Some common statistical tests, Monitoring research, Records to be maintained by researchers, hypothesis testing and inferences. Tables and illustrations: Making of a table, Arrangement of data in tables, Illustrations, Graphs and visualization.

Information sources: Tools for identifying literature, Types of publications, Indexing and abstracting services, online library, Search engines, Citation indexes, Citations analysis, online searching methods, initiatives for knowledge management.

References: How to cite and list correctly, Why citation of source materials is important, Citation and listing system of documents, Common documentation styles, Citation of sources in the text, Examples of writing references, Electronic and online Sources, Some worked out examples for the prominent styles, Reference management softwares, Selecting a journal, conference, Impact Factor of journal as per Journal Citation Report, SNIP, SJR, IPP, Cite Score, Metrics: h-index, g index, i10 index, altmetrics

UNIT-IV: Scientific writing: **10 Hours**

Use of paragraphs and sentence checking. Abbreviations: Care with numerals, SI unit rules and style conventions. Titles, Verbs, Tenses, The active or passive voice,'articles', Prepositions, Relative pronouns, Transitional words and phrases, Economy in words, Singular and plural words, Commonly confused words, Choosing correct words and phrases, Punctuation. Preparation of thesis and research papers: Structure of a thesis/research article/ review article,

including title, introduction, literature review, methods and materials, referencing. Editing and proofreading, use of abbreviations.

TEXT/REFERENCE BOOKS:

1. C. R. Kothari, Research Methodology: Methods and Techniques, New Age International, 2004, ISBN 8122415229, 978812241522
2. Kumar R. Research Methodology: A Step by Step Guide for Beginners (2010) 3rd ed., Pearson Education. (ISBN-13: 978-1849203012)
3. Relevant study material from ACM, IEEE, Elsevier, Springer

RCS- 102
RESEARCH AND PUBLICATION ETHICS

- Syllabus will be as per the UGC Guideline.

RCS- 103
FOUNDATION OF COMPUTER SCIENCE

UNIT-I: Fundamental of Computer: **10 Hours**

Characteristics of Computers, Evolution of computers, computer memory, computer generations, Basic computer organization; System software, Application software, introduction to operating system and characteristics, Windows, Linux. Computer Network, Internet.

UNIT-II: Basics Software: **10 Hours**

Microsoft office, Word and Excel, Uses of Word in research, Editor and Researcher tools uses. Data Analysis using Ms-Excel. Research publishing tool- MS-Word, Adobe acrobat, Grammarly, Graphics tool- MSExcel, MS-Power Point: Creating presentations and adding effects.

UNIT-III: Uses of Internet in Research: **15 Hours**

Internet, Characteristics, internet services, electronic mail, www, downloading, basic security and safety measures. Article searching. Advanced Search in Google Scholar, Google scholar, Shodhganga, ShodhGangotri, ResearchGate, SWAYAM, NPTEL, UGC-CARE

Data Analysis: Data analysis software-SPSS, Definition, objectives and features, data analysis using SPSS: Data entry creating variables, switching to data labels, data analysis: Frequencies, recording into different variables, cross tabulations and layers. Core calculation software, developing utility programs for research.

UNIT-V: Understanding of Various Research Tools: **10 Hours**

Latex,- introduction and various features, Mendeley referencing manager characteristics and uses, What is IPR? Characteristics and applications, Patent and filling procedure.

Communications and collaboration: Basics of electronic mail; Getting an email account; Sending and receiving emails; Accessing sent emails; Using Emails; Document collaboration; Instant Messaging; Netiquettes.

TEXT/REFERENCE BOOKS:

1. Grotenhuis, Manfred te, and Matthijssen, Anneke. Basic SPSS Tutorial. United States, SAGE Publications, 2015.
2. Raubenheimer, Jacques. Mendeley: Crowd-sourced Reference and Citation Management in the Information Era. United Kingdom, True Insight Publishing, 2014.
3. Goel, Anita. Computer Fundamentals. India, Pearson Education, 2010.
4. Maidasani, Dinesh. Learning Computer Fundamentals, Ms Office and Internet & Web Tech.. India, Laxmi Publications, 2005.
1. Wishnietsky, Dan H.. Internet Basics: An Educator's Guide to Traveling the Information Highway. United States, Phi Delta Kappa Educational Foundation, 1997.
2. McCormick, Keith, and Salcedo, Jesus. SPSS Statistics for Data Analysis and Visualization. Germany, Wiley, 2017.

RCS- 104/105
ELECTIVE PAPER
ADVANCED SOFTWARE ENGINEERING

UNIT-I: Software Engineering: **10 Hours**

Review of software engineering, Key concepts, Classical SDLC model of software engineering, COTS, RUP, Rapid Application Development, Agile methods, Extreme programming. Advanced concepts of software engineering: Component based software engineering, Aspect oriented software engineering, Service oriented software engineering, Object oriented software engineering. Software project management: Introduction to software project management, Project managers roles and responsibilities, Project planning.

UNIT-II: Requirement Engineering: **10 Hours**

Introduction to requirement engineering, Importance and challenges in requirement engineering, Success factors of requirement engineering, Requirement engineering framework, Characteristics of requirements, Issues and problems in requirement elicitation, Trends and challenges of requirement elicitation, Techniques and approaches of requirement elicitation, Role of soft skills in requirement analysis, Requirement reuse, Requirement verification & validation. Specification of requirement models: Characteristics of SRS, Components of SRS, Specification language, Structure of requirement document, Guidelines for writing SRS. Requirement Prioritization:

Aspects of prioritization, Prioritization techniques, Requirement negotiation, Ambiguity in requirement engineering.

UNIT-III: Software Design: **15 Hours**

Design principles, Software design process, Design qualities, Design patterns. Design practices: Incremental design, Structured analysis and structured design, Jacksons structured programming, Introduction to object oriented design. Software visualization: Introduction and challenges, Static visualization, Dynamic program visualization, Evaluating visualizations, Data visualization.

Software Testing: Terminology, Life cycle, Methodology, Types of testing, Test planning, Static testing, Black box testing, White box testing and their types, Model for software testing, Unit testing, Integration testing, System and Acceptance Testing. Object oriented testing: Object oriented testing model, Object oriented software test strategy. Software security and risk management: Fundamentals of software security and risk management, Reactive vs. proactive risk strategies, Software risks, Risk identification, Risk projection, Risk refinement, RMMM and RMMM Plan.

UNIT-IV: Introduction to Software Quality: **10 Hours**

Concepts of quality, perspectives and expectations, Quality Framework. Quality assurance: Classification, Quality assurance activities and techniques. Quality models: McCall's model, Bohem's model, Dromey's model, FURPS Model, ISO9126 model, Cost of quality, Software quality factors, Quality control, CMMI framework. Reengineering: Business process reengineering, Software reengineering, Reverse engineering, Restructuring, Forward reengineering.

TEXT/REFERENCES BOOKS:

1. 'Software Engineering - A practitioner's Approach', Roger S. Pressman, 7 editions, McGraw-Hill International Edition.
2. 'Software Engineering', Sommerville, 9th edition, Addison-Wesley Publication.
3. 'Software Engineering' K.K. Agarwal & Yogesh Singh, New Age International Publication.
4. 'Software Engineering: An Engineering approach', James F. Peters, Witold Pedrycz, Wiley Publication.
5. 'Software Engineering Principles and Practice', Waman S Jawadekar, The McGraw -Hill Publishing Co. Ltd.

RCS- 104/105
ELECTIVE PAPER
SOFTWARE RELIABILITY

UNIT-I: Reliability Fundamentals: **10 Hours**

Need for Reliability Engineering, Definition, Causes of Failures, Catastrophic Failures and Degradation Failures, Characteristic Types of Failures, Useful Life of Components, Exponential Case of Chance Failures, Reliability Measures, Failure Data Analysis.
Reliability Mathematics: Probability Theory, Random Variables, Discrete Distributions, Continuous Distributions, Stochastic Processes, Markov Chains.

UNIT-II: Reliability Analysis of Series Parallel Systems: 10 Hours
Reliability Block Diagrams, Series Systems, Parallel Systems, Series Parallel Systems, Open and Short Circuit Failures, Standby Systems.
Reliability Analysis of Non-Series Parallel Systems: Path Determination, Boolean Algebra Methods, A Particular Method, Cut Set Approach, Delta Star Method, Logical Signal Relations Method, Bayes Theorem Method.

UNIT-III: Reliability Prediction: 15 Hours
Purpose, Classification, General Requirements, Prediction Methodologies, Software Prediction, Packages, Role and Limitation of Reliability Prediction.
Reliability Allocation: Subsystems Reliability Improvement, Apportionment for New Units, Criticality, Redundancy, Techniques for Reliability Optimization, Signal Redundancy, Time Redundancy, Hardware Redundancy.
Maintainability and Availability: Forms of Maintenance, Measures of Maintainability and Availability, Maintainability Function, Availability Function, Two Unit Parallel System with Repair, Preventive Maintenance, Provisioning of Spares.
Reliability Testing: Types of Testing, Component Reliability Measurements, Parametric Methods, Confidence Limits, Accelerated Testing, Equipment Acceptance Testing, Reliability Growth Testing.
Software Reliability: Software Reliability and Hardware Reliability, Failures and Faults, Software Reliability, Software Reliability Models, Execution Time Component, Calendar Time Component.
Reliability Analysis of Special Systems: Phased Mission Systems, Common Cause Failures, Reliability and Capacity Integration.

UNIT-IV: Economics of Reliability Engineering: 10 Hours
Effect of Reliability on Cost, Reliability Achievement Cost Models, Reliability Utility Cost Models, Availability Cost Models for Parallel Systems, Cost Effective Choice of Subsystems, Replacement Policies.
Reliability Management: Management Objectives, Top Managements Role in Reliability and Quality Control Programs, Cost Effectiveness Considerations, Management Matrix, Reliability and Quality Control Facilities and Equipment, Reliability Data.

TEXT/REFERENCES BOOKS:
1. K.K. Aggarwal, “Reliability Engineering”, Springer Science & Business Media.

2. E Balaguruswami, “Reliability Engineering”, TMH.

RCS- 104/105
ELECTIVE PAPER
SOFTWARE TESTING AND QUALITY ASSURANCE

UNIT-I: Software Testing Fundamentals: 10 Hours

Software Testing, Purpose, Goal and Objectives, Effective Software Testing, Types of Testing, Principles of Software Testing, Testing and Debugging, Psychology and Economics of Software Testing.

Software Testability: Testability Artifacts, Testability Facilitators, Estimation and Analysis. **Testing:** Program Inspection, Walkthrough and Reviews. **Principles of Static Testing:** General Methodology and Automated Techniques, Dynamic Testing

UNIT-II: Black Box Testing: 10 Hours

Equivalence Partitioning, Boundary Value Analysis, Robustness Testing, Syntax Testing, Finite State Testing. **White Box Testing:** White Box Technique, Modeling, Basic Path Testing, Control Structure Testing, Mutation Testing, Gray Box Testing.

Software Testing Strategies: Strategic Issues and Premises, Unit Testing, Integration Testing, Validation Testing, Syntax Testing, Regression Testing, Software Component Testing, Testing Real Time Systems, Models for Software Testing.

UNIT-III: System Testing: 15 Hours

Recovery Testing, Security Testing, Stress Testing, Performance Testing, **Planning for Software Testing:** Test Plan Specification, Leveled Test Plan, Development of Test Plan, Master Test Plan, Phase wise Test Plan. **Building Test Cases:** White Box Test Cases and Test Procedures, Test Data Selection and Outputs, Black-Box Test Cases and Test Procedures, Testing Specialized Systems, Object Oriented Testing: Pertinent Issues, Object Oriented Testing Model and Test Strategy.

Testing in Agile Environment: Features of Agile Development, Agile Testing, Extreme Programming and Testing. **Software Fault Tolerance:** Objective and Need, Software Failure, Principles and Techniques of Software Fault Tolerance, Fault based Testing Methods.

Testing Measurements and Tools: Test Standards, Product Defects, Process Measures. **Test Tools:** Test Planning and Management Tools.

UNIT-IV: Quality Assurance and Standards: **10 Hours**

Quality Concepts, Quality Control, Quality Assurance, Cost of Quality. **Software Quality Assurance:** Background Issues and SQA Activities, Software Reviews, Formal Technical Reviews, Formal Approaches to SQA.

Software Reliability: Measures of Software Reliability and Availability, Software Safety, ISO 9000 Quality Standards: ISO Approach to Quality Assurance System, ISO 9001 Standards, SQA Planning Assurance, SQA Plan, Software Testing and QA, Configuration Management.

TEXT/REFERENCES BOOKS:

1. Glenford J. Myers, Tom Badgett, Corey Sandler, "The Art of Software Testing", John Wiley & Sons, Inc.
2. K. Mustafa, R. A. Khan, "Software Testing: Concepts and Practices", Alpha Science, Oxford, UK.
3. R. A. Khan, "Software Quality: Concepts and Practice", Alpha Science, Oxford, UK.
4. W.M Perry, "Effective Methods for Software Testing", Wiley Publication.

RCS- 104/105
ELECTIVE PAPER
AD-HOC AND WIRELESS NETWORKS

UNIT-I: Introduction: **10 Hours**

Fundamentals of Wireless Communication Technology -The Electromagnetic Spectrum - Radio propagation Mechanisms - Characteristics of the Wireless channel mobile ad hoc networks (MANETs), concepts and architecture of MANETs - Applications of Ad Hoc network - Design Challenges in Ad hoc Networks.

UNIT-II: Mac Protocols for Ad hoc Wireless Networks: **10 Hours**

Issues in designing a MAC Protocol - Issues in Designing a MAC Protocol for Ad Hoc Wireless Networks - Design Goals of a MAC Protocol for Ad Hoc Wireless Networks - Classification of MAC Protocols -Contention based protocols - Contention based protocols with Reservation Mechanisms - Contention based protocols with Scheduling Mechanisms - Multi channel MAC - IEEE 802.11.

UNIT-III: Routing Protocols for Ad hoc Wireless Networks: **15 Hours**

Introduction, Issues in Designing a Routing Protocol for Ad hoc Wireless Networks; Classification of Routing Protocols; Table Driven Routing Protocols; On-Demand Routing Protocols, Hybrid Routing Protocols, Hierarchical Routing Protocols and Power-Aware Routing Protocols. Transport Layer protocol for Ad hoc networks, Network Security - Security in Ad Hoc Wireless Networks

Quality of Services and Energy Management in Ad hoc Networks: Introduction, Issues and Challenges in Providing QoS in Ad-hoc Wireless Networks, Classification of QoS Solutions,

MAC Layer Solutions, Network Layer Solutions; Energy Management in Ad-hoc Wireless Networks.

Introduction to Vehicular Ad hoc Networks (VANETs): Characteristics, architecture of VANETs, Applications, Challenges, Applicability of MANET Routing to Vehicular Environment, Routing protocols for VANET. Use of Infrastructure in VANETs, Vehicular Network Simulators, Vehicular Mobility Models.

UNIT-IV: Introduction to Flying Ad hoc Networks (FANETs): 10 Hours

Characteristics, architecture of FANETs, Applications, Challenges, Applicability of MANETs as well as VANETs routing in FANETs. Comparison FANETs with existing ad-hoc networks such as MANETs, and VANETs. Recent advancements in FANETs with simulation tools.

TEXT/REFERENCES BOOKS:

1. C.Siva ram murthy,B.S. Manoj, "Ad hoc wireless networks-Architectures and protocols" Pearson Education, 2005
2. Stefano Basagni, Marco Conti, "Mobile ad hoc networking", Wiely interscience 2004
3. Charles E.Perkins ,"Ad hoc networking", Addison Wesley, 2001
4. Xiuzhen Cheng, Xiao Huang ,Ding Zhu DU ,"Ad hoc wireless networking", Kluwer Academic Publishers, 2004
5. George Aggelou,"Mobile ad hoc networks-From wireless LANs to 4G networks, McGraw Hill publishers, 2005

RCS- 104/105
ELECTIVE PAPER
FUZZY LOGIC AND ARTIFICIAL NERUAL NETWORK

UNIT-I: Introduction: 10 Hours

crisp sets an overview, the notion of fuzzy sets, Basic concepts of fuzzy sets, classical logic an overview, Fuzzy logic. Applications of fuzzy logic: Washing Machine, Train braking system, Sugar mill control, Industrial applications

UNIT-II: Operations: 10 Hours

Operations on fuzzy sets, fuzzy complement, fuzzy union, fuzzy intersection, combinations of operations, general aggregation operations Crisp and fuzzy relations, binary relations, binary relations on a single set, equivalence and similarity relations.

UNIT III: Compatibility: 15 Hours

Compatibility or tolerance relations, orderings, Membership functions, methods of generation, defuzzification methods. General discussion, belief and plausibility measures, probability measures, possibility and necessity measures, relationship among classes of fuzzy measures.
Classical Logic: An overview, fuzzy logic, fuzzy rule-based systems, fuzzy decision making, fuzzy logic in database and information systems, fuzzy pattern recognition, fuzzy control systems.

UNIT-IV: Artificial Neural Networks: 10 Hours
Fundamentals of neural networks, model of an artificial neuron, neural network architectures, Learning methods, Taxonomy of Neural network architectures, Standard back propagation algorithms, selection of various parameters, variations Applications of back propagation algorithms.

TEXT/REFERENCES BOOKS:

1. George J Klir and Tina A Folger, "Fuzzy Sets, Uncertainty and Information", Prentice Hall of India, 1998.
2. H.J. Zimmerman, "Fuzzy Set Theory and its Applications", 4/e, Kluwer Academic Publishers, 2001.
3. George Klir and Bo Yuan, "Fuzzy Sets and Fuzzy Logic: Theory and Applications", Prentice Hall of India, 1997.
5. Timothy J Ross, "Fuzzy Logic with Engineering Applications", McGraw Hill International Editions, 1997.
6. Hung Nguyen and Elbert Walker, "A First Course in Fuzzy Logic, 2/e,, Chapman and Hall/CRC,1999.
7. Jerry M Mendel, "Uncertain Rule-based Fuzzy Logic Systems: Introduction and New Directions, PH PTR, 2000.

RCS-104/105
ELECTIVE PAPER
WIRELESS SENSOR NETWORKS

UNIT – I: Overview of Wireless Sensor Networks: 10 Hours
SingleNode Architecture Hardware Components Network Characteristic s unique constraints and challenges, Enabling Technologies for Wireless Sensor Networks Types of wireless sensor networks.

UNIT – II: Architecture: 10 Hours
Network Architecture S ensor NetworksScenario s Design Principle, Phys ical Layer and Transceiver Design Considerations, Optimization Goals and Figures of Merit, Gateway Concepts, Operating Systems and Execution Environments introdu ction to Tiny OS and nesC Internet to WSN Communication.

UNIT – III: Networking Sensors: **15 Hours**

MAC Protocols for Wireless Sensor Networks, Low Duty Cycle Protocols And Wakeup Concepts – SMAC, BMAC Protoco l, IEEE 802.15.4 standard and ZigBee, the Mediation Device Protocol, Wakeup Radio Concepts, Address and Name Management, Assignment of MAC Addresses, Routing Protocols EnergyEfficien t Routing, Geographic Routing. **Infrastructure Establishment:** Topology Control, Clustering, Time Synchronization, Localization and Positioning, Sensor Tasking and Control.

UNIT – IV: **Sensor Network Platforms and Tools:** **10 Hours**

Sensor Node Hardware – Berkeley Motes, Programming Challenges, Nodele vel software platforms, Node level Simulators, S tatecentric programmin g.

TEXT/REFERENCES BOOKS:

1. Holger Karl & Andreas Willig, "Protocols And Architectures for Wireless Sensor Networks", John Wiley, 2005.
2. Feng Zhao & Leonidas J.Guibas, “Wireless Sensor Networks An Information Processing Approach", Elsevier, 2007.
3. Waltenegus Dargie , Christian Poellabauer, “Fundamentals of Wireless Sensor Networks Theory and Practice”, John Wiley & Sons Publications, 2011
5. Kazem Sohraby, Daniel Minoli, & Taieb Znati, “Wireless Sensor NetworksT echnology, Protocols, and Applications”, John Wiley, 2007.
6. Anna Hac, “Wireless Sensor Network Designs”, John Wiley, 2003

RCS- 104/105
ELECTIVE PAPER
WIRELESS TECHNOLOGIES FOR IoT

UNIT-I: RF Basics: **10 Hours**

Radio Frequency (RF) Fundamentals: Introduction to RF & Wireless Communications Systems, RF and Microwave Spectral Analysis, Communication Standards, Understanding RF & Microwave Specifications. Spectrum Analysis of RF Environment, Protocol Analysis of RF Environment, Units of RF measurements, Factors affecting network range and speed, Environment, Line-of-sight, Interference, Defining differences between physical layers- OFDM.

UNIT-II: Cellular Standards: **10 Hours**

Cellular carriers and Frequencies, Channel allocation, Cell coverage, Cell Splitting, Microcells, Picocells, Handoff, 1st, 2nd, 3rd and 4 th Generation Cellular Systems (GSM, CDMA, GPRS, EDGE, UMTS), Mobile IP, WCDMA

UNIT-III: WLAN: **15 Hours**

Wi-Fi Organizations and Standards: IEEE, Wi-Fi Alliance, WLAN Connectivity, WLAN QoS & Power-Save, IEEE 802.11 Standards,802.11- 2007,802.11a/b/g, 802.11e/h/I,802.11n

Wi-Fi: Hardware & Software: Access Points, WLAN Routers, WLAN Bridges, WLAN Repeaters, Direct-connect Aps, distributed connect Aps, PoE Infrastructure, Endpoint, Client hardware and software, Wi-Fi Applications

UNIT-IV: WSN & WPN: **10 Hours**

Wireless Personal Area Networks, Bluetooth, Bluetooth Standards, Bluetooth Protocol Architecture, UWB, IEEE 802.15 standards, ZigBee, Sub1GHz, Sensor Networks, coexistence strategies in Sensor Networks, Routing protocols in Wireless Sensor Networks.

TEXT/REFERENCES BOOKS:

1. Wireless Communications – Principles and Practice; by Theodore S Rappaport, Pearson Education Pte. Ltd., Delhi
2. Wireless Communications and Networking; By: Stallings, William; Pearson Education Pte. Ltd., Delhi
3. Bluetooth Revealed; By: Miller, Brent A, Bisdikian, Chatschik; Addison Wesley Longman Pte Ltd., Delhi
4. Wilson, “Sensor Technology hand book,” Elsevier publications.
5. Andrea Goldsmith, “Wireless Communications,” Cambridge University Press
6. Mobile and Personal Communications Services and Systems; 1st Edition; By: Raj Pandya; PHI, New Delhi

RCS-104/105
ELECTIVE PAPER
SOFTWARE ARCHITECTURE AND PROJECT MANAGEMENT

UNIT-I: Software Architecture: **10 Hours**

Introduction, Evolution of Software Development, Fundamentals of Software Engineering, Elements of Software Architecture, Architectural Description, Types of Architecture, The Software Product Life Cycle, Management View, Software Engineering View, Engineering Design View, Architectural View, Synthesize the View: CASE study of Architectures.

UNIT-II:The Architecture Design Process: **10 Hours**

Identifying Design Elements and their Relationships, Evaluating the Architecture, Transforming the Architecture, Software Design, Problems in Software Architectural Design, The Scope of Design, The Psychology and Philosophy of Design, General Methodology of Design

UNIT-III:The Architecture Representation and Quality: 15 Hours

Goals of Architecture Representation, Foundations of Software Architecture Representation, Architecture Description Languages, Architectural Level of Design, Architecting with Design Operators, Functional Design Strategies, Importance of Assessing Software Architecture, How to Improve Quality, Architecture Evaluation, Assessing Performance

Project Concepts and its Management: Project Life Cycle Models, ISO 9001 Model, Capability Maturity Model Project, Software Management Process Framework: Phases, Artifacts, Workflows, Checkpoints, Software Management Disciplines: Planning / Project Organization and Responsibilities / Automation / Project Control, Modern Project Profiles, Classification of Levels, Work Break Down Structure, Assessing Duration, Line of Balance Technique.

UNIT-V: Project Evaluation and Emerging Trends: 10 Hours

Strategic Assessment, Technical Assessment, Cost Benefit Analysis, Cash Flow Forecasting, Cost Benefit Evaluation Technique, Risk Evaluation, Software Effort Estimation, Emerging Trends: Import of the Internet on Project Management, People Focused Process Models.

TEXT/REFERENCES BOOKS:

1. Stephen T. Albin, "The Art of Software Architecture: Design Methods and Techniques", John Wiley & Sons, 2003
2. Len Bass, Paul Clements, Rick Kazman, "Software Architecture in Practice", Third Edition, Addison-Wesley, 2003.
3. Richard N.Taylor, Nenad Medvidovic and Eric M.Dashofy, "Software Architecture, Foundations, Theory and Practice", Wiley 2010.
4. Bob Hughes and Mike Cotterell, "Software Project Management", Second Edition, 1999.
5. Royce, W., "Software Project Management: A Unified Framework", Addison-Wesley, 1998.

RCS- 104/105
ELECTIVE PAPER
INFORMATION SECURITY

UNIT-I: Information Security: 10 Hours

Understand the meaning of information security, Security models, security attacks, security services and mechanisms, Understand security threats: Social Engineering (Phishing), firewalls, buffer overflow, spoofing, Network Sniffing, Denial of services, Botnet, Phishing in Online Social Media(OSM) & Identifying fraudulent entities in OSM,Principles of Cryptography, Steganography and Watermarking

UNIT–II: Symmetric Key Cryptography: 10 Hours

Classical Encryption and Decryption Techniques, Stream Cipher, RC4, Block Ciphers, Data Encryption Standard (DES), Triple DES, Modes of DES (ECB, CBC, CFB, OFB), Advanced Encryption Standard, Confidentiality using symmetric encryption

UNIT-III:Public key cryptography: 15 Hours

Symmetric and asymmetric techniques (Diffie-Hellman, Needham-Schroeder), Key distribution, RSA, Elliptic Curve Cryptography, Message Authentication and Hash Function, Cryptographic hash function, Non Cryptographic Hash function, Birthday problem, birthday attack, HMAC, CMAC.

Digital Signature: Digital signature schemes: RSA and DSS (Digital Signature Standard),Message integrity and authentication protocols: definition and applications, Collision resistant hashing, authenticated encryption: security against active attacks.

UNIT-IV: Cryptanalysis & Security Protocol: 10 Hours

Cryptanalysis-Enigma, RC4 as used in WEP, Linear and Differential Cryptanalysis attacks on RSA, Security protocol: Security Protocols-SSH, SSL, IPSec, Kerberos, WEP, GSM

TEXT/REFERENCES BOOKS:

1. William Stallings, Cryptography and network security, Pearson Education.
2. Alfred J. Menezes, Paul C. van Oorschot and Scott A. Vanstone , Handbook of Applied Cryptography, CRC Press.
3. Stefan Katzenbeisser and Fabien Petitcolas, "Information Hiding, Techniques for Steganography and Digital Watermarking," Artech House, 2000.

RCS- 104/105
ELECTIVE PAPER
INFORMATION HIDING

UNIT-I: Information Hiding: 10 Hours

Introduction, Background, and Applications of Information hiding: Data hiding, applications of data hiding.Difference between Cryptography, Steganography and Watermarking

UNIT-II: Steganography: 10 Hours

Frameworks of secret communication, Security of steganography systems, Information hiding in noisy data, Adaptive & non-adaptive algorithms, Active and malicious attackers, Information hiding in written text, Invisible communication.

UNIT-III: Data Hiding in Still Images: 15 Hours

LSB encoding, BPCS steganography, Lossless data hiding, Data hiding by quantification, Patchwork, Transform domain methods, Robust data hiding in JPEG images, frequency domain

watermarking Detecting malicious tampering, Robust wavelet based watermarking, Kundur-Hatzinakos watermarking.

Data Hiding in Binary Images & Steganalysis:

Different Method for hiding in binary images: Zhao-kochmethod, Wu-Lee method, CPT method, TP method, Data hiding in fax images. Steganalysis - Steganalysis introduction and terminology, Detecting hiding information, Extracting hiding information, Disabling hidden information.

UNIT-V: Watermarking: 10 Hours

Introduction, Watermarking principals, Applications, Requirements and algorithmic design issues, Evaluation and standards of watermarking. Fingerprinting: Introduction, Terminology and requirements, Classifications, Research history, fingerprinting schemes, Statistical fingerprinting, and Collusion-secure fingerprinting.

TEXT/REFERENCES BOOKS:

1. I.J.Cox, M.L.Miller, J.A.Bloom, J.Fridrich, T.Kalker, Digital Watermarking and Stegonagraphy, Morgan Kaufman 2008.
2. F.Y.Shih, Digital Watermarking and Stegonagraphy Fundamentals and Techniques, CRC press 2008.
3. Stefon Katzeubeisser, F.A.Petitolos, Information Hiding Techniques for Stegonagraphy and Digital Watermarking, Aatech House London 2008.

RCS- 104/105
ELECTIVE PAPER
IMAGE PROCESSING TECHNIQUES

UNIT-I: Digital Image Fundamentals: 10 Hours

Introduction – Origin – Steps in Digital Image Processing – Components – Elements of Visual Perception – Image Sensing and Acquisition – Image Sampling and Quantization – Relationships Between Pixels, Elements of Digital Image Processing, Digital Image, File formats, Raster and Vector Form.

Introduction to Image Restoration, Importance, Basic Framework, Interactive Restoration, Image deformation and geometric transformations, image morphing, Restoration techniques. Noise models.

UNIT –II: Fundamental of the Image Enhancement: 10 Hours

Introduction to Spatial Domain: Gray Level Transformations – Introduction to Histogram Processing, Smoothing and Sharpening Spatial Filtering. Frequency Domain: Introduction to Fourier Transform –with Basics Filters – Ideal, Butterworth And Gaussian Filters.

UNIT- III: Image Compression: **15 Hours**

Image Compression Fundamentals, Redundancies, Image compression models, Error free Compression, lossy compression, Image compression standards.

Morphological Image Processing and Image Segmentation:

Basics, SE, Erosion, Dilation, Opening, Closing, Hit-or-Miss Transform, Boundary Detection, Hole filling, Connected components, convex hull, thinning, Image Segmentation: Boundary detection based techniques, Point, line detection, Edge detection, Edge linking, local processing, regional processing, Introduction to wavelets. Applications of Wavelet Transform.

UNIT-V: Fundamentals of Pattern Recognition: **10 Hours**

Introduction to pattern recognition, features and feature vectors, concepts of learning: supervised, unsupervised and reinforced. Basic concepts of clustering and classification, classifiers: based on Bayesian decision theory, perceptron model, artificial neural networks, support vector machine, nearest neighbours.

TEXT/ REFERENCES BOOKS:

1. Rafael C. Gonzales, Richard E. Woods, "Digital Image Processing", Third Edition, Pearson Education, 2010.
2. R.O.Duda, P.E.Hart and D.G.Stork, "Pattern Classification 2nd Edition", John Wiley, 2007
3. Christopher M. Bishop, "Neural Network for Pattern Recognition", Oxford Ohio Press.
4. Rafael C. Gonzalez, Richard E. Woods, Steven L. Eddins, "Digital Image Processing Using MATLAB", Third Edition Tata Mc Graw Hill Pvt. Ltd., 2011.
5. Anil Jain K. "Fundamentals Of Digital Image Processing", PHI Learning Pvt. Ltd., 2011.

RCS- 104/105
ELECTIVE PAPER
MACHINE LEARNING

UNIT- I: Introduction To Machine Learning: **10 Hours**

Concept of Machine Learning, Applications of Machine Learning, Key elements of Machine Learning, Supervised vs. Unsupervised Learning, Statistical Learning: Bayesian Method, The Naive Bayes Classifier, Decision Trees.

UNIT-II: Regression: **10 Hours**

Linear Regression: Prediction using Linear Regression, Gradient Descent, Linear Regression with one variable, Linear Regression with multiple variables, Polynomial Regression, Feature Scaling/Selection. Logistic Regression: Classification using Logistic Regression, Logistic

Regression vs. Linear Regression, Logistic Regression with one variable and with multiple variables, Optimization, Decision Trees vs Logistic Regression

UNIT-III: Neural Networks: **15 Hours**

Introduction, Model Representation, Gradient Descent vs. Perceptron Training, Stochastic Gradient Descent, Multilayer Perceptrons, Multiclass Representation, Back-propagation Algorithm and related Issues.

Support Vector Machines: Knowing SVMs, SVM as Large Margin Classifier, Kernels, Using SVMs in Learning complex nonlinear functions, Constrained Optimization, SVM utilization.

UNIT-IV: Introduction to Clustering: **10 Hours**

K-Means Algorithm, Random Initialization, Choosing the number of Clusters, Mixture Models, Dimensionality Reduction and its application, Principal Component Analysis (PCA). Error Analysis and Anomaly Detection: Utility of Error Analysis, Precision/Recall, Error Metrics for Skewed Classes, Introduction of Anomaly Detection, Gaussian Distribution, Anomaly Detection vs. Supervised Learning.

TEXT/REFERENCES BOOKS:

1. Ethem Alpaydin, "Introduction to Machine Learning", 3rd Edition, The MIT Press.
2. Simon O. Haykin, "Neural Networks and Learning Machines", Pearson Education, 2016.
3. C. M. Bishop, "Pattern Recognition and Machine Learning", Springer, 2010.
4. Smola and Vishwanathan, "Introduction to Machine Learning", Cambridge University Press, 2010.
5. T.M. Mitchell, "Machine Learning", McGraw Hill Education, 2017.

RCS- 104/105
ELECTIVE PAPER
TRENDS IN DIGITAL IMAGE PROCESSING AND PATTERN LEARNING

UNIT-I: Essential of Digital Image Processing: **10 Hours**

Introduction to Image processing, Basics, Steps of Fundamentals of Imaging, Imaging systems, radiometric models, geometric models, sampling and quantization, image models; Image File Formats, Colour Maps and Tables. Image Transforms: Fourier, Walsh, Haar.

UNIT-II: Image Enhancements Approaches: **10 Hours**

Spatial Operations including point Processes (histogram equalization, specification), windowing.Frequency Domain Operations for Noise removal, contrast enhancements etc, Image Restorations: model estimation,Noise Modeling, Motion blur, iener, inverse filter.Image compression: Lossless, lossy etc. Mathematical Morphology, Basic Algorithms, Advanced Topics and Applications.

UNIT-III: Geo-Special Image Processing: **15 Hours**

Introduction to Remote Sensing, Fundamentals of aerial photos and satellite image interpretation; Types of imaging, elements of interpretation; Techniques of Visual interpretation; Generations of Thematic maps. Noises, Study of satellite image annotation, Demarcation of contours & watershed using toposheets, Remote sensing applications: features extractions from remote sensing data, Understanding of spectral response pattern of different landforms. Image Interpretation and Analysis. Semantic Feature Extraction: An Intro.

Medical Image Processing: Introduction Medical imaging, Overview on Research Domains: Bio-Medical Image processing tools, methodology, Data sets. Overview of Research working methodology. Thresholding and Segmentation: Detection methods, optimal thresholding, multi-spectral thresholding. Edge based segmentation, Region based segmentation, Matching, Advanced optimal border and surface detection approaches.

UNIT-V: Essential of Pattern Learning Recognitions: 10 Hours

Define pattern recognition, Introduction, Gaussian model, discriminant functions, Introduction to Parametric and Nonparametric techniques. Classifiers, classifier performance, risk and errors; Supervised learning using parametric and nonparametric approaches: ML estimation, clustering: the clustering concept, c-means.

TEXT/REFERENCE BOOKS:

1. Digital Image Processing by Gonzalez and Woods, 3rd Edition, Addison-Wesley, 2008
2. Duda R O and P E Hart, Pattern classification and scene analysis, John Wiley & Sons, NY 1973
3. Andrew R. Webb, Statistical Pattern Recognition, Second Edition, 2002, John Wiley and Sons Ltd.
4. K.S.Fu, Syntactic pattern recognition and applications, Prentice Hall, NJ, 1982
5. T.Pavlidis, Structural pattern recognition, Springer-Verlag, NY, 1977

RCS- 104/105
ELECTIVE PAPER
ADVANCED RESEARCH IN CYBER SECURITY

UNIT-I: Systems Vulnerability Scanning: 10 Hours

Overview of vulnerability scanning, Open Port / Service Identification, Banner / Version Check, Traffic Probe, Vulnerability Probe, Vulnerability Examples, OpenVAS, Metasploit. Networks Vulnerability Scanning - Netcat, Socat, understanding Port and Services tools - Datapipe, Fpipe, WinRelay, Network Reconnaissance – Nmap, THC-Amap and System tools. Network Sniffers and Injection tools – Tcpdump and Windump, Wireshark, Ettercap, Hping Kismet.

UNIT-II: Network Defense tools: 10 Hours

Firewalls and Packet Filters: Firewall Basics, Packet Filter Vs Firewall, How a Firewall Protects a Network, Packet Characteristic to Filter, Stateless Vs Stateful Firewalls, Network Address Translation (NAT) and Port Forwarding, the basic of Virtual Private Networks, Linux Firewall, Windows Firewall, Snort: Introduction Detection System.

UNIT-III: Web Application Tools: 15 Hours

Scanning for web vulnerabilities tools: Nikto, W3af, HTTP utilities - Curl, OpenSSL and Stunnel, Application Inspection tools – Zed Attack Proxy, Sqlmap. DVWA, Webgoat, Password Cracking and Brute-Force Tools – John the Ripper, L0htcrack, Pwdump, HTC-Hydra.

Advanced Research Orientation: Foundations : Security Models, Formal Methods, Cryptography, Application Centric: Secure information Sharing, Social Computing, Health Care, Data Provenance,Technology Centric: Cloud Computing, Smart Grid, Trusted Computing, Attack Centric Related works.

UNIT-IV: Introduction to Cyber Crime Investigation: 10 Hours

Firewalls and Packet Filters,Password Cracking, Keyloggers and Spyware, Virus and Warms, Trojan and backdoors,Steganography, DOS and DDOS attack, SQL injection, Buffer Overflow, Attack on wireless Networks.

TEXT/REFERENCES BOOKS:

1. Anti-Hacker Tool Kit (Indian Edition) by Mike Shema, Publication Mc Graw Hill.
2. Cyber Security Understanding Cyber Crimes, Computer Forensics and Legal Perspectives by Nina Godbole and Sunit Belpure, Publication Wiley.

RCS- 104/105
ELECTIVE PAPER
SECURITY IN COMPUTING

UNIT-I: Security Models and Assessment: 10 Hours

Data Versus Information,Identification and Authentication Essentials, Access Control and Access Control Structures, Security Policies Security Models and Confidentiality, Organization Security Architecture, Risk Analysis, Vulnerability Analysis, Security Audits and Risk Management, Security Assurance and Evaluation Criteria.

UNIT-II: Network Security TCP/IP Security: 10 Hours

PPP, ECP. TLS EAP, DESE-bis, Firewall, IP Sec Architecture and Protocols, Dial in Operations, RAS PAP, CHAP, RADIUS, DIAMETER, Key distribution, IKE, Certification and Management, Intrusion Detection Systems, VLANs and VPNs, Email security, Network Attacks and DNS protection, DMZ , Proxy services.

UNIT-III: Operating System and Application Security: 15 Hours

PGP, Authentication, Processes, Files, Users, Buffer Overflow Attacks, Kernel Flaws, Logging, Backups, Access Control Systems using Swipe Cards, RFID, Biometrics.

Techniques Encryption Cryptography Techniques: Block ciphers & Data encryption standards, Public key encryption and hash functions, Authentication applications.

UNIT-IV: Security Evaluation: 10 Hours

Practices and Incident Response Security Audit, Network Audit, Security Policy, Risk Mitigation, Computer Forensics

TEXT/REFERENCES BOOKS:

1. William Stallings, Cryptography and network security principles and practices, Pearson Education, Fourth Edition , 2007
2. Douglas R.Stinson, Cryptography Theory and Practice, CRC Press, Second Edition, 2005.
3. Matt Bishop, Computer Security Art & Science, Pearson Education, second Indian reprint 2005.
4. Charlie Kauffman, Radia Perlman, Mike Speciner Network Security Priavte Communication in Public World, Pearson Education, 2005
5. IEEE Transactions on Information Forensics & Security (IEEE Journal ACM Transactions on Information and System Security (TISSEC)), International Journal of Information Security: By Springer.

RCS-104/105
ELECTIVE PAPER
RESEARCH IN COMPUTER FORENSICS

UNIT-I: Computer forensics analysis: 10 Hours

Determining what data to collect and analyze. Addressing data hiding techniques, Hiding partitions, Marking bad clusters, Bit –shifting, using steganography to hide data, Examining Encrypted files, Recovering Passwords, Performing Remote Acquisitions, RemoteAcquisitions with Runtime Software.

UNIT-II: Recovering graphics files: 10 Hours

Understanding vector Graphics, Understanding graphics file formats .Lossless and lossy compression. Identifying graphics file fragments, Repairing Damaged Headers, Searching For and carving data from unallocated space. Understanding steganography in graphics files. Using steganalysis tools. Understanding copyright issues with graphics.

UNIT-III: Virtual Machines, Network forensics, and Live Acquit ions: 15 Hours

Performing live acquisitions, Performing a live acquisition in windows, Developing standard procedures for network forensics, Reviewing network logs. Using network tools, usingUnix/Linux tools. Using packet sniffers, examining the honey net projects.

Mail Investigation: Exploring the role of email investigation, Exploring the role of client and server in email, Investigating E-mail crimes and violations, Examining E-mail Messages, Viewing E-mail headers, Examing E-mail headers, Examining additional E-mail files. Tracing an e-mail message.

UNIT-IV: Cell phone and mobile device forensics: 10 Hours

Understanding mobile device forensics, Mobile phone basics, inside mobile devices, insidePDAs, Understanding acquisition procedures for cell phones and mobile devices, Mobileforensics equipment.

TEXT/REFERENCES BOOKS:

1. Bill Nelson,Amelia Phillips,Christopher Steuart,"Guide to Computer Forensics and Investigations", Fourth Edition,Course Technology.
2. Angus M.Marshall, "Digital forensics: Digital evidence in criminal investigation", John – Wiley and Sons, 2008.

******End*****