

Syllabus For
2-Year PG Programme
in
Computer Applications

Master of Computer Applications
[MCA]

as per

NATIONAL EDUCATION POLICY-2020



**Aryavart School of
Computer Science and Engineering**
Aryavart International University

Tilthai, Dharmanagar, North Tripura-799260

Introduction

In the 21st century, information technology and computer applications have emerged as the foremost drivers of digital transformation, innovation, and economic growth across the globe. The rapid evolution of software systems, cloud computing, data science, and artificial intelligence has made advanced computing education an indispensable pillar of higher learning. The two-year Master of Computer Applications (MCA) Programme at Aryavart International University (AIU) is a postgraduate programme designed to develop the technical proficiency, problem-solving capability, and professional software-development skills essential for success in the information technology industry and beyond.

Aligned with the National Education Policy (NEP) 2020, the MCA Programme at AIU is a two-year (four-semester) postgraduate programme that offers students the flexibility to specialise in a domain of their choice — such as Software Engineering, Data Science, Artificial Intelligence & Machine Learning, or Cloud Computing & Cybersecurity. The Programme integrates theoretical learning with industry-oriented practical training, IBM-certified skill modules, live projects, and a major project in collaboration with leading industry partners.

Programme Objectives

The objectives of the two-year MCA Programme are:

- To provide students with a strong foundation in computing fundamentals, programming paradigms, and software development practices.
- To develop analytical, logical, and algorithmic problem-solving abilities required for effective software design and development.
- To foster an entrepreneurial mindset and innovation through project-based learning, incubation support, and industry interaction.
- To equip students with skills in data structures, database systems, cloud computing, and emerging technologies such as AI and machine learning.
- To inculcate ethical values, data privacy awareness, and professional integrity essential for responsible software practice.
- To prepare students for doctoral research (Ph.D.), competitive examinations, and senior technical leadership roles while facilitating direct industry placement.
- To develop effective communication, teamwork, and project-management skills through seminars, workshops, and group activities.
- To promote research aptitude and knowledge generation through project reports, case studies, and software development assignments.

Programme Learning Outcomes

On the completion of the MCA Programme, students will be able to:

- ✓ Demonstrate comprehensive understanding of core computing functions including programming, databases, networks, and software engineering.
- ✓ Apply computing theories and design frameworks to real-world software development challenges.
- ✓ Analyse large datasets and generate strategic insights using modern tools and data science techniques.
- ✓ Design and implement software products and start-up ventures, supported by AIU's incubation centre and industry network.
- ✓ Exhibit professional communication, presentation, and leadership skills essential for technical and managerial roles.
- ✓ Demonstrate awareness of emerging technology trends, cloud infrastructure, and cybersecurity practices.
- ✓ Uphold ethical standards and data-protection responsibility in professional conduct.
- ✓ Qualify for national competitive examinations and pursue doctoral research and advanced technical roles.

Need for Curriculum Development

The MCA curriculum at Aryavart International University is framed in alignment with the UGC guidelines and NEP 2020 to produce industry-ready software professionals and future technology leaders. The following factors necessitate a contemporary and flexible MCA curriculum:

Rapidly Evolving Technology Landscape

Rapid changes in programming frameworks, cloud platforms, and user expectations demand that students be trained in emerging technology domains such as artificial intelligence, cybersecurity, DevOps, and full-stack development. The curriculum integrates these evolving domains to bridge the gap between academia and industry.

Credit Transfer and Academic Mobility

In line with the UGC's credit transfer provisions under NEP 2020, the MCA Programme enables students to transfer credits between programmes and institutions, pursue interdisciplinary electives, and take vocational/skill enhancement courses from approved platforms such as IBM SkillsBuild.

Skill Enhancement and Employability

The curriculum places strong emphasis on industry-oriented skill development, IBM-certified modules, hackathons, and live software projects. Aryavart School of Computer Applications at AIU maintains active collaborations with IBM to ensure that students are trained on tools and competencies demanded by recruiters.

Career Options

The MCA Programme opens diverse and rewarding career pathways. Graduates of AIU's MCA Programme are equipped to pursue technical and leadership careers including:

- Software Developer / Full-Stack Engineer
- Data Scientist / Data Analyst
- Cloud Engineer / DevOps Engineer
- Artificial Intelligence & Machine Learning Engineer
- Database Administrator / Data Engineer
- Cybersecurity Analyst / Ethical Hacker
- Systems Analyst / IT Consultant
- Mobile Application Developer
- Entrepreneur / Start-up Founder (supported by AIU Incubation Centre)
- Project Manager / Scrum Master

Pedagogy

The MCA pedagogy at Aryavart International University is designed to create practice-oriented, industry-aligned software professionals and future technology leaders. The curriculum employs a blend of traditional and experiential learning methodologies.

Integration of Theory and Practice

Each course balances conceptual foundations with hands-on application. Students engage in coding labs, software project work, hackathons, and industry case studies to translate theoretical knowledge into professional development competencies.

ICT-Enabled Teaching

Smart classrooms, computing labs, cloud-based development environments, and IBM-powered training platforms are integrated into the instructional design to promote digital fluency and higher-order thinking.

Research-Based Learning

Students undertake semester-wise project reports and a final-year research/development project under faculty supervision, fostering analytical rigour and a research temperament consistent with NEP 2020's emphasis on inquiry-based education.

Methods of Instruction

- Lectures by subject experts and industry practitioners
- Seminars, workshops, and guest lectures
- Hands-on programming labs and coding sessions
- Case study analysis and software design exercises
- Hackathons and live industry projects
- IBM SkillsBuild and online certification modules

Methods of Evaluation

- Internal Sessional Examinations (Internal Assessment Exams)
- End Semester Examinations (Theory / Practical)
- Assignments, presentations, and project reports
- Lab-based and software project submissions
- Viva-voce and practical evaluations
- Attendance and class participation

Scheme of Evaluation

A. MCA — Scheme of Evaluation (Theory Courses)

Component	Sub-component	Marks
Internal Assessment (30 Marks)	Continuous Mode	15
	Internal Assessment Examination (2 Hours)	15
End Semester Examination (Theory)	3 Hours Written Examination	70
TOTAL		100

B. Scheme of Evaluation (Practical / Project Courses)

Applicable to all practical, lab-based, and project courses of the MCA Programme:

Component	Sub-component	Lab Exam / Viva	Marks
Internal Assessment (30 Marks)	Assignment / Presentation	Lab-Based Exam	30
End Semester Examination	Project / Report	Viva-Voce	70
TOTAL			100

C. Seminar — Scheme of Evaluation (100 Marks)

The seminar component is evaluated on a 70:30 basis, comprising 30 marks of internal (continuous) assessment and 70 marks of end-semester evaluation, with the viva-voce forming an integral part of the final seminar presentation.

Component	Sub-component	Mode of Evaluation	Marks
Internal Assessment (30 Marks)	Continuous Mode / Review Presentations	Continuous Evaluation	30
End-Semester Evaluation (70 Marks)	Final Seminar Presentation including Viva-Voce	Presentation & Viva-Voce	70
TOTAL			100

D. Major Project — Scheme of Evaluation (200 Marks, External Assessment)

Applicable to students in the 4th Semester (Year II) of the MCA Programme. The major project is undertaken under faculty guidance and evaluated entirely through external assessment for a total of 200 marks.

Component	Sub-component	Mode of Evaluation	Marks
External Assessment (200 Marks)	Project Report Evaluation	External Examiner	140
	Presentation & Viva-Voce	External Viva	60
TOTAL			200

Exit Options and Credit Requirements

As per NEP 2020 provisions, students may exit the programme at the end of each year and are awarded the corresponding certificate/degree:

Exit Option / Completion	Certificate / Award
Successful completion of Year I (2 Semesters)	Postgraduate Diploma in Computer Applications (PGDCA)
Successful completion of Year II (4 Semesters)	Master of Computer Applications (MCA)

Internal Assessment: Continuous Mode

The Internal Assessment for the MCA includes two components: (i) Internal Assessment Examinations and (ii) Continuous Mode Evaluation. The Continuous Mode constitutes 15 marks per theory/practical course and is assessed on an ongoing basis throughout the semester as detailed below.

MCA — Continuous Mode Assessment (15 Marks)

The Continuous Mode Assessment for the MCA is conducted on an ongoing basis throughout the semester and comprises the following components, emphasising broad-based computing knowledge, regular engagement, and all-round participation across core technical disciplines:

Sl. No.	Component	Criteria / Basis	Marks
1	Attendance	≥95%: 5 94-90%: 4 85-89%: 3 80-84%: 2 75-79%: 1 <75%: 0	5
2	Assignment / Lab Task / Quiz	Written assignments, coding tasks, or class quiz/test assigned by the course faculty	5
3	Teacher–Student Interaction / Presentation / Viva	Oral interaction, seminar/topic presentation, group discussion, or class participation	5
TOTAL			15

Note: Internal Assessment Examinations are conducted twice per semester. The average of the two sessional exams (each of 50 marks, averaged to 15) is taken for the purpose of Internal Assessment. The course faculty shall announce the schedule and modalities of Continuous Mode Assessment at the commencement of each semester.

Course Structure for the Master of Computer Application Program as Per NEP 2020								
Semester	Major Subject (Core Subject) (4 + 2 Credits)	Additional/ Interdisciplinary subject/ Multidisciplinary (4 Credits)	Skill Course/ Vocational Course (2 Credits)	Ability Enhancement Courses (2 Credits)	Discipline specific/ Open Elective/Project (4 + 2 Credits)	Extracurricular Courses (2 Credits)	Value Added Courses/ VAC (2 Credits)	Total Credits
I	Object-Oriented Programming with C++	Software Engineering/ Management Information System	Not Applicable	Business Communication	DSE-I (Choose any one paper)	Not Applicable	Not Applicable	24
	Computer Organization & Architecture							
II	Database Management System	Theory of Computation/ Introduction to Artificial Intelligence	Not Applicable	Environmental Studies	DSE-II (Choose any one paper)	Not Applicable	Health & Wellness	24
	Data Structure & Algorithm using C							
Exit option with Post Graduate Diploma in Computer Application (48 Credits)								
III	Cryptography	Introduction to Machine Learning/ Data Warehousing & Mining	Not Applicable	Not Applicable	DSE-III (Choose any one paper)	Not Applicable	Human Values & Ethics	24
	Design Analysis & Algorithms							
IV	Cloud Computing	Not Applicable	Not Applicable	Seminar	Major Project	Not Applicable	Drug Abuse, Road Safety & Traffic Rules	24
	Image Processing							
Exit with a Masters in Computer Application (96 Credits)								

*Major Project carries 8 credits.

#One Research/Review Paper to be published in peer-reviewed journal in both 3rd and 4th Sem.

ARYAVART INTERNATIONAL UNIVERSITY
Tilthai, Dharmanagar, North Tripura

Discipline Specific Elective (DSE- I): Choose any one paper with its Practical

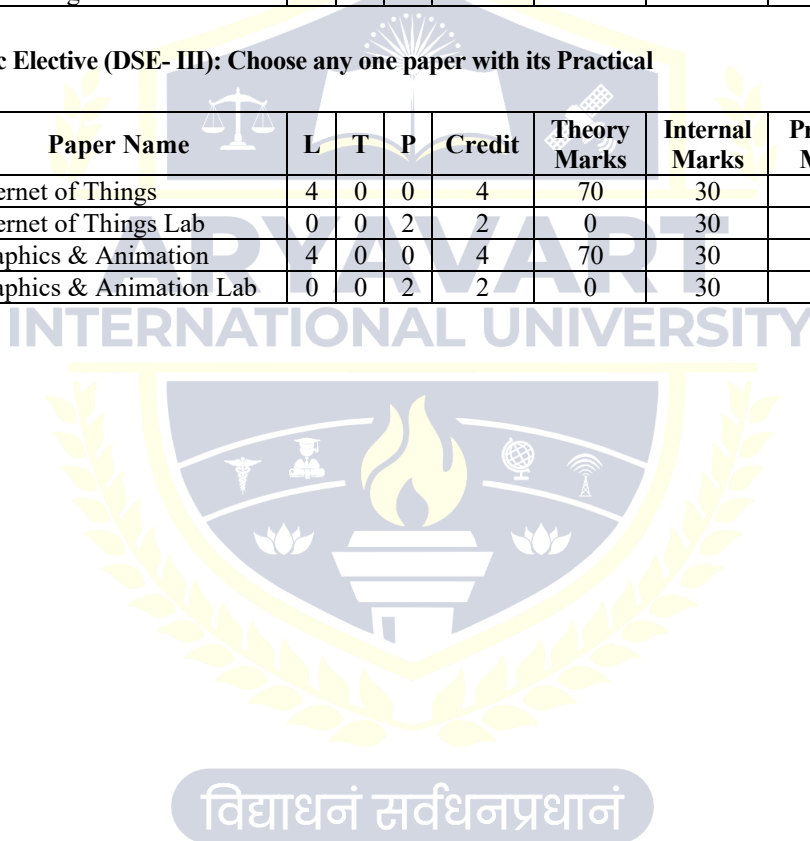
Course Code	Paper Name	L	T	P	Credit	Theory Marks	Internal Marks	Practical Marks	Total Marks
26CS132	Linux-OS	4	0	0	4	70	30	0	100
26CS293	Linux-OS Lab	0	0	2	2	0	30	70	100
26CS401	Computer Networks	4	0	0	4	70	30	0	100
26CS491	Computer Networks Lab	0	0	2	2	0	30	70	100

Discipline Specific Elective (DSE- II): Choose any one paper with its Practical

Course Code	Paper Name	L	T	P	Credit	Theory Marks	Internal Marks	Practical Marks	Total Marks
26CS231	Android App Development	4	0	0	4	70	30	0	100
26CS294	Android App Development Lab	0	0	2	2	0	30	70	100
26CS533	Natural Language Processing	4	0	0	4	70	30	0	100
26CS595	Natural Language Processing Lab	0	0	2	2	0	30	70	100

Discipline Specific Elective (DSE- III): Choose any one paper with its Practical

Course Code	Paper Name	L	T	P	Credit	Theory Marks	Internal Marks	Practical Marks	Total Marks
26CS731	Internet of Things	4	0	0	4	70	30	0	100
26CS793	Internet of Things Lab	0	0	2	2	0	30	70	100
26CS331	Graphics & Animation	4	0	0	4	70	30	0	100
26CS393	Graphics & Animation Lab	0	0	2	2	0	30	70	100



MCA

1st Semester

DETAILED SYLLABUS

OBJECT-ORIENTED PROGRAMMING WITH C++

Code: 26CS201

Max Marks: 70

Course Objectives: The objective of the course is to understand OOP principles like classes, inheritance, and polymorphism, develop modular and reusable code using C++ and implement real-world problem solutions using OOP techniques

UNIT I (10 Hrs)

Object-Oriented Paradigm: Procedural vs. object-oriented development, Basic concepts of object-oriented programming, Applications and benefits of OOP, Comparison between C and C++.

Beginning with C++: Stream-based I/O, Literals- constant qualifiers, Operators in C++, Reference variable, Functions, Default arguments, Parameter passing by value, Reference and pointer, Inline functions, Type conversion, Basic C++ programs, New, Delete operators- basic use and dynamic memory allocation for arrays.

UNIT II (10 Hrs)

Classes and Objects: C++ class declaration, Access specifiers, Member functions, Arrays within a class, Array of objects, Memory allocation of objects, passing objects as arguments, Returning objects from functions, Function overloading, Static data and member functions, Friend function and friend class, This pointer.

Constructors & Destructors: Introduction to constructors and destructors, Parameterised constructor, Constructor with default arguments, Multiple constructors in a class, Copy constructor.

UNIT III (10 Hrs)

Inheritance: Types of inheritance, Derivation – public, private & protected, Ambiguity resolution (function overriding), Aggregation, Composition v/s Classification, Virtual base class, Constructor and destructor in derived classes.

Polymorphism: Types of polymorphism, early v/s late binding, Virtual Functions: Need for virtual functions, Pointer to derived class objects, Pure virtual functions, Abstract classes.

Operator Overloading: Overloading unary operators, Nameless objects, Overloading binary operators, Overloading with friend functions, Conversion between basic types and user-defined types.

UNIT IV (10 Hrs)

Parametric polymorphism: Generic Programming with Templates, Introduction, Function templates/generic functions, Characteristics, Overloading of template functions, Class templates, Template arguments.

Exception Handling: Exception-handling model, Types of exception, Catching and Handling exceptions, Generic catch, Rethrowing an exception, Specifying exceptions for a function.

Streams & Files: C++ Streams, Basic stream classes, C++ predefined streams, I/O operations, Unformatted console I/O operations, Manipulators, Opening and closing a file- different modes and methods, Error handling during file operations, File pointers and their manipulations, Sequential access to file, Random input and output operations, Persistent objects, Command line arguments.

Text Books:

1. K.R. Venugopal, Rajkumar, T. Ravishanker, "Mastering C++", TMH
2. E. Balagurusamy, "Object Oriented Programming with C++", McGraw-Hill Education

Suggested Readings:

1. Ashok N. Kamthane, "Object-Oriented Programming with ANSI And Turbo C++", Pearson Education.
2. Schildt Herbert, "C++: The Complete Reference", Tata McGraw Hill.
3. R. Lafore, "Object Oriented Programming using C++", Galgotia Publications.

COMPUTER ORGANIZATION AND ARCHITECTURE

Code: 26CS202

Max Marks: 70

Course Objectives: The objective of the course is to learn the structure and functioning of computer hardware, understand instruction sets, memory hierarchy, and CPU organisation, and analyse performance and system-level operations.

UNIT I

(10 Hrs)

Boolean Algebra and Logic: Basics, Laws of Boolean Algebra, Logic Gates, Simplifications of Boolean equations using K-maps SOP and POS, Don't Care condition.

Arithmetic Circuits: Adder, Subtractor, Parallel binary adder/Subtractor.

UNIT II

(10 Hrs)

Combinational Circuits: Multiplexers, De-Multiplexers, Decoders, Encoders.

Flip-flops: S-R, D, J-K, T, Clocked Flip-flop, Race around condition, Master slave Flip-Flop, Realisation of one flip-flop using other flip-flop, Applications of flip-flop: Latch, Registers, Counters (elementary treatment to be given).

UNIT III

(10 Hrs)

Data Transfer Operations: Register Transfer, Bus and Memory Transfer, Registers and micro-operations.

Basic Computer Organisations and Design: Instruction Codes, Computer Registers, Instruction Cycle, General Register Organisation, Stack Organisation, Instruction Formats, Addressing Modes.

UNIT IV

(10 Hrs)

Input-Output Organisation: Peripheral Devices, Input-Output Interfaces, Asynchronous Data Transfer, Modes of Transfer, Priority Interrupt, Direct Memory Access (DMA).

Memory Organisation: Main Memory, Auxiliary Memory, Associative Memory, Cache Memory, Virtual Memory.

Text Books:

1. Morris Mano, Computer System Architecture, 3rd Edition, Prentice-Hall of India Private Limited, 1999.
2. Morris Mano, "Digital Logic and Computer Design", PHI Publications, 2002

Suggested Readings:

1. R. P. Jain, "Modern Digital Electronics", TMH, 3rd Edition, 2003.
2. William Stallings, Computer Organisation and Architecture, 4th Edition, Prentice Hall of India Private Limited, 2001
3. Subrata Ghosal, "Computer Architecture and Organisation", Pearson, 2011
4. Malvino, "Digital Computer Electronics: An Introduction to Microcomputers", McGraw-Hill.

विद्याधनं सर्वधनप्रधानं

SOFTWARE ENGINEERING

Code: 26CS112

Max Marks: 70

Course Objectives: The objective of the course is to enable students to learn the software development life cycle and methodologies, apply principles of requirement analysis, design, testing, and maintenance, and understand project management and quality assurance.

UNIT I

(10 Hrs)

Introduction of software engineering: Software Crisis, Software life cycle models, Waterfall, Prototype, Spiral Models, Agile model.

Software Requirements analysis & specifications: Requirement engineering, requirement elicitation techniques like FAST, QFD, Requirement analysis using (DFD use-case, sequence and class diagram (with case studies), ER Diagrams, Requirements documentation: SRS, Characteristics & organization of SRS.

UNIT II

(10 Hrs)

Software Project Planning: Software Metrics-Definition and Need, Types of Metrics-Product, Process and Project Metrics, Size Estimation like lines of Code & Function Count, Halstead Software Science measure, Cost Estimation: Need, Models COCOMO: Basic model, Intermediate model.

Risk Management: Software Risks, Types of risk, risk management activities: risk assessment, risk control.

UNIT III

(10 Hrs)

Software Design: Cohesion & Coupling, Classification of Cohesiveness & Coupling.

Quality management: Quality concept, software quality assurance, Total Quality Management (TQM), software review, software inspection.

Software Implementation: Structured coding techniques, coding style, Standards and guidelines, documentation guidelines. Reverse Engineering, Software Re-engineering, Configuration Management.

UNIT IV

(10 Hrs)

Software Testing: Testing Process, Levels of Testing: Unit testing, Integration testing and system testing. Types of Testing: Manual testing, Automation Testing. Methods of Testing- , Black box, White box and Grey Box Testing. Validation, Verification, Alpha-Beta testing, Acceptance testing, Functional Testing and its types, Structural Testing Difference between: Testing and Debugging.

Software Maintenance: Management of Maintenance, The Maintenance Process and Types of maintenance: Preventive, Perceptive, Adaptive and Corrective Maintenance. Maintenance tools and techniques.

Text Books:

1. K. K. Aggarwal & Yogesh Singh, "Software Engineering", 2nd Ed., New Age International, 2005.
2. I. Sommerville, "Software Engineering", 9th Edition, Pearson Edu

Suggested Readings:

1. Jibitesh Mishra and Ashok Mohanty, "Software Engineering", Pearson
2. R. S. Pressman, "Software Engineering – A practitioner's approach", 5th Ed., McGraw Hill Int. Ed., 2001.
3. James Peter, W. Pedrycz, "Software Engineering: An Engineering Approach", John Wiley & Sons.

विद्याधनं सर्वधनप्रधानं

ARYAVART INTERNATIONAL UNIVERSITY
Tilthai, Dharmanagar, North Tripura

MANAGEMENT INFORMATION SYSTEM

Code: 26MG103

Max Marks: 70

Course Objectives: To equip Management Information Systems (MIS), focusing on their role in decision-making, system design, IT integration, security, ethical issues, and strategic use to improve organizational efficiency and competitive advantage.

UNIT I (06 Hrs)

Basic Concepts of Information System Role of data and information, Organization structures, Business Process, Systems Approach and introduction to Information Systems.

UNIT II (08 Hrs)

Types of IS Resources and components of Information System, integration and automation of business functions and developing business models. Role and advantages of Transaction Processing System, Management Information System, Expert Systems and Artificial Intelligence, Executive Support Systems and Strategic Information Systems.

UNIT III (08 Hrs)

Architecture & Design of IS Architecture, development and maintenance of Information Systems, Centralised and Decentralised Information Systems, Factors of success and failure, value and risk of IS.

UNIT IV (06 Hrs)

Decision Making Process: Programmed and Non- Programmed decisions, Decision Support Systems, Models and approaches to DSS

UNIT V (08 Hrs)

Introduction to Enterprise Management technologies, Business Process Reengineering, Total Quality Management and Enterprise Management System viz. ERP, SCM, CRM and E-commerce.

UNIT VI (04 Hrs)

Introduction to SAD System Analysis and Design. Models and Approaches of Systems Development.

Text Books:

1. Management Information Systems, Effy OZ, Thomson Learning/Vikas Publications.
2. Management Information Systems, James A. O'Brein, Tata McGraw-Hill.

Suggested Readings:

1. Management Information System, W.S Jawadekar, Tata McGraw-Hill Publication.
2. Management Information System, David Kroenke, Tata McGraw-Hill Publication.
3. MIS: Management Perspective, D.P. Goyal, Macmillan Business Books.
4. MIS and Corporate Communications, Raj K. Wadhwa, Jimmy Dawar, P. Bhaskara Rao, Kanishka Publishers.
5. MIS: Managing the digital firm, Kenneth C. Landon, Jane P. Landon, Pearson Education.

ARYAVART INTERNATIONAL UNIVERSITY
Tilthai, Dharmanagar, North Tripura

BUSINESS COMMUNICATION

Code: 26GN102

Max Marks: 70

Course Objective: The objective of Business Communication is to develop effective communication skills in a business environment, focusing on clarity, conciseness, and professionalism. It aims to enhance students' ability to write and present business documents, engage in meaningful dialogues, and navigate cross-cultural communication challenges.

UNIT I: Concepts and Fundamentals

(04 Hrs)

Introduction to Technical Communication, Need, and importance of communication, Channel, Distinction between general and technical communication, Nature and features of technical communication, Seven Cs of communication, Types of Technical communication, Style in technical communication, technical communication skills, Language as a tool of Communication, History of the development of Technical Communication, Computer Aided Technical Communication

UNIT II: Oral Communication

(06 Hrs)

Principles of effective oral communication, Introduction of Self and others, Greetings, Handling Telephone Calls, Interviews: Meaning & Purpose, Art of interviewing, Types of Interviews, Interview styles, Essential, Techniques of interviewing, Guidelines for Interviewer, Guidelines for interviewee. Meetings: Definition, kinds of meetings, agenda, meeting minutes, advantages and disadvantages of meetings/ committees, and planning and organisation of meetings. Project Presentations: Advantages & Disadvantages, Executive Summary, Charts, Distribution of time (presentation, questions & answers, summing up), Visual presentation, Guidelines for using visual aids, electronic media (PowerPoint presentation). The technique of conducting a group discussion and a JAM session.

UNIT III: Written Communication

(06 Hrs)

Overview of Technical Writing: Definition and Nature of Technical Writing, Basic Principles of Technical Writing, Styles in Technical Writing.

Note – Making, Notice, E-mail Writing.

Writing letters: Business letters, persuasive letters, sales letters and complaint letters, office memorandum, and good news and bad news letters.

Report Writing: Definition & importance; categories of reports, Elements of a formal report, style, and formatting in the report.

Special Technical Documents Writing: Project synopsis and report writing, Scientific Article and Research Paper writing, Dissertation writing: Features, Preparation and Elements.

Proposal Writing: Purpose, Types, characteristics, and structure.

Job Application: Types of application, Form & Content of an application, Drafting the application, Preparation of a resume.

UNIT IV: Soft Skills and Language Skills

(04 Hrs)

Business Etiquettes – Professional Personality, Workplace Protocols, Cubicle. Non-Verbal Communication: Kinesics and Proxemics, Paralanguage.

Interpersonal Skills.

Improving command in English, improving vocabulary, Choice of words, Common problems with verbs, Adjectives, Adverbs, Pronouns, Tenses, Conjunctions, punctuation, prefixes, Suffixes, and idiomatic use of prepositions. Sentences and paragraph construction, Improved spellings, Common errors, and misappropriation, building advanced Vocabulary (Synonyms, Antonyms), and introduction to Business English.

Text Books:

1. Kavita Tyagi and Padma Misra, “Advanced Technical Communication”, PHI, 2011
2. P. D. Chaturvedi and Mukesh Chaturvedi, “Business Communication – Concepts, Cases and Applications”, Pearson, second edition.

ARYAVART INTERNATIONAL UNIVERSITY
Tilthai, Dharmanagar, North Tripura

3. Rayudu, "C. S- Communication", Himalaya Publishing House, 1994.
4. Asha Kaul, "Business Communication", PHI, second edition.

Suggested Readings:

1. Raymond Murphy, "Essential English Grammar- A self-study reference and practice book for elementary students of English", Cambridge University Press, second edition.
2. Manalo, E. & Fermin, V. (2007). "Technical and Report Writing". ECC Graphics. Quezon City.
3. Kavita Tyagi and Padma Misra, "Basic Technical Communication", PHI, 2011.
4. Herta A Murphy, Herbert W Hildebrandt, and Jane P Thomas, "Effective Business Communication", McGraw-Hill, seventh edition.

LINUX-OS

Code: 26CS132

Max. Marks:70

Course Objectives: The operating systems course is an essential part of any Computer-Science education. The purpose of this course is to understand the mechanisms of the Operating Systems like Process Management, Process Synchronization, Memory Management, File System Implementation, Storage Structures used in OS and Protection Principles. How effectively the OS is utilizing the CPU resources with the help of these mechanisms.

UNIT I

(8 Hrs)

Introduction to OS: Functionality of OS - OS design issues - Structuring methods (monolithic, layered, modular, micro-kernel models) - Abstractions, processes.

OS Principles: System calls, System/Application Call Interface – Protection: User/Kernel modes – Interrupts.

UNIT II

(10 Hrs)

Scheduling: Processes Scheduling - CPU Scheduling: Pre-emptive, non-pre-emptive – Multiprocessor scheduling – Deadlocks - Resource allocation and management - Deadlock handling mechanisms: prevention, avoidance, detection, recovery.

Concurrency: Inter-process communication, Synchronization - Implementing synchronisation primitives (Peterson's solution, Bakery algorithm, synchronisation hardware) - Semaphores – Classical synchronisation problems, Monitors: Solution to Dining Philosophers problem.

UNIT III

(8 Hrs)

Memory Management: Main memory management, Memory allocation strategies, Virtual memory: Hardware support for virtual memory (caching, TLB) – Paging - Segmentation - Demand Paging - Page Faults – Page Replacement –Thrashing.

UNIT IV

(8 Hrs)

Storage Management, Protection and Security: Disk structure and attachment – Disk scheduling algorithms (seek time, rotational latency based)-System threats and security – Policy vs mechanism - Access vs authentication.

UNIT V

(6 Hrs)

System protection: Access matrix – Capability-based systems - OS: performance, scaling, future directions in mobile OS, Recent Trends.

Text Books:

1. Abraham Silberschatz, Peter B. Galvin, Greg Gagne. "Operating System Concepts", Wiley (2018).
2. Stallings (2006), "Operating Systems, Internals and Design Principles", 5th edition, Pearson Education, India.

Suggested Readings:

1. Ramez Elmasri, A. Gil Carrick, David Levine. "Operating Systems: A Spiral Approach". McGraw Hill Higher Education (2010).
2. Remzi H. Arpaci-Dusseau, Andrea C. Arpaci-Dusseau. "Operating Systems: Three Easy Pieces". Arpaci-Dusseau Books, Inc (2015).
3. Andrew S. Tanenbaum, "Modern Operating Systems". Pearson, 4th Edition (2016).
4. William Stallings. "Operating Systems: Internals and Design Principles". Pearson, 9th Edition (2018).

LINUX-PROGRAMMING LAB

Code: 26CS293

Max Marks: 70

Core Practical:

1. Connect to the Linux Server and understand the basic Directory Structure of Linux.
2. To understand help commands like: -man, info, help, what is, apropos
3. To understand basic directory navigation commands like cat, cd, mv, cp, rm, mkdir, rmdir, file, pwd command.
4. To understand basic commands like: - date, cal, echo, bc, ls, who, whoami, hostname, uname, tty, alias.
5. To understand vi basics, three modes of vi Editor, how to write, save, execute a shell script in vi editor.
6. To understand process related commands like: -ps, top, pstree, nice, renice in Linux.
7. To understand how to examine and change File permissions.
8. Set a file to be read-only with the chmod command. Interpret the file permissions displayed by the ls -l command.
9. Delete one or more directories with the rmdir command. See what happens if the directory is not empty. Experiment (carefully!) with the rm -r command to delete a directory and its content
10. Change your directory to the directory exercises. Create a file in that directory, named the file as example1 using the cat command containing the following text: water, water everywhere and all the boards did shrink; water, water everywhere, no drop to drink.
11. Write basic shell script to display the table of a number.
12. Write basic shell script to input a character from user and then check whether it is uppercase, lowercase or digit.
13. Write basic shell script to calculate factorial of a number.
14. Write basic shell script to input the month number and generate corresponding calendar.
15. Write basic shell script to list all directories.
16. Write basic shell script to display greatest of three numbers.
17. Write basic shell script to check whether the number entered by user is prime or not.

Note:

1. In total 15 practical to be implemented. 2 additional practicals may be given by the course instructor.
2. This is a suggestive list of programs. However, the instructor may add programs as per the requirement of the course.

विद्याधनं सर्वधनप्रधानं

ARYAVART INTERNATIONAL UNIVERSITY
Tilthai, Dharmanagar, North Tripura

COMPUTER NETWORKS

Code: 26CS401

Max Marks: 70

Course Objectives: The objective of the course is to understand network models, protocols, and architecture, learn data transmission, routing, and switching concepts and study network security and communication standards.

UNIT I (10 Hrs)

Basic Concepts: Components of data communication, Distributed processing, Line configuration, Topology, Transmission mode and Categories of networks.

OSI and TCP/IP Models: Layers and their functions, Comparison of models.

Transmission Media: Guided and unguided, Attenuation, Distortion, Noise, Throughput, Propagation speed and time, Wavelength, Shannon Capacity.

UNIT II (10 Hrs)

Telephony: Multiplexing, WDM, TDM, FDM, Circuit switching, Packet switching and Message switching.

Data Link Layer: Types of errors, Framing (character and bit stuffing), Error detection & Correction methods; Flow control; Protocols: Stop & wait ARQ, Go-Back- NARQ, Selective repeat ARQ.

UNIT III (10 Hrs)

Network Layer: Internet working & Devices: Repeaters, Hubs, Bridges, Switches, Router, Gateway, Modems;

Addressing: IPv4 and IPv6 addressing, IPv4 subnetting; Routing: Unicast Routing Protocols: RIP, OSPF, BGP;

Routing: Routing Methods- Static and Dynamic Routing, Routing basic commands, Distance vector protocol, Link state protocol.

UNIT IV (10 Hrs)

Transport and upper layers in OSI Model: Transport layer functions and Protocols, Connection management, Functions of session layers, Presentation layer and Application layer.

Text Books:

1. A. S. Tanenbaum, "Computer Networks"; Pearson Education Asia, 4th Ed., 2003.
2. Behrouz A. Forouzan, "Data Communication and Networking", 2nd edition, Tata Mc Graw Hill.

Suggested Readings:

1. D. E. Comer, "Internetworking with TCP/IP", Pearson Education Asia, 2001.
2. William Stallings, "Data and computer communications", Pearson education Asia, 7th Ed., 2002.
3. Leinwand, A., Pinsky, B. (2001). Cisco router configuration. United Kingdom: Cisco Press.

COMPUTER NETWORKS LAB

Code: 26CS491

Max Marks: 70

List of Experiments:

1. Implement the data link layer framing methods such as character, character stuffing and bit stuffing.
2. Implement on a data set of characters the three CRC polynomials – CRC 12, CRC 16 and CRC CCIP.
3. Implement Dijkstra's algorithm to compute the Shortest path thru a graph.
4. Take an example subnet graph with weights indicating delay between nodes. Now obtain Routing table art each node using distance vector routing algorithm.
5. Take an example subnet of hosts. Obtain broadcast tree for it.
6. Take a 64 bit playing text and encrypt the same using DES algorithm.
7. Write a program to break the above DES coding.
8. Using RSA algorithm encrypt a text data and Decrypt the same.

C++ LAB

Code: 26CS291

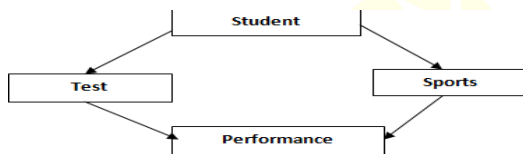
Max Marks: 70

Core Practical (Implement minimum 8 out of 10 practical)

1. WAP to implement 'Inline function'.
2. WAP to implement call by reference and return by reference using class. [Hint. Assume necessary functions].
3. WAP to implement friend function by taking some real-life examples.
4. WAP to implement 'Function Overloading'.
5. WAP to implement Parameterized Constructor, Copy Constructor and Destructor.
6. WAP to show the usage of constructor in base and derived classes, in multiple inheritance.
7. WAP to show the implementation of 'containership'.
8. WAP to show swapping using template function (Generic).
9. WAP to implement 'Exception Handling'.
10. WAP to read and write values through object using file handling.

Application Based Practical (Implement minimum 5 out of 10 practical)

11. Create a class employee which have name, age and address of employee, include functions getdata() and showdata(), getdata() takes the input from the user, showdata() display the data in following format:
Name:
Age:
Address:
12. Write a class called **C Account** which contains two private data elements, an integer **Account Number** and a floating-point **account Balance** and three-member functions:
 - A constructor that allows the user to set initial values for **Account Number** and **Account Balance** and a default constructor that prompts for the input of the values for the above data numbers.
 - A function called **Input Transaction**, which reads a character value for **Transaction Type** ('D' for deposit and 'W' for withdrawal) and a floating-point value for **Transaction Amount**, which updates **Account Balance**.
 - A function called **Print Balance**, which prints on the screen the **Account Number** and **Account Balance**.
13. Define a class *Counter* which contains an int variable *count* defined as static and a static function Display () to display the value of *count*. Whenever an object of this class is created *count* is incremented by 1. Use this class in main to create multiple objects of this class and display value of count each time.
14. WAP to add and subtract two complex numbers using classes.
15. Write program to overload Binary + to add two similar types of objects. (Both with and without using friend functions)
16. WAP to implement += and = operator
17. Implement the following class hierarchy considering appropriate data members and member functions:



18. Implement the following hierarchy considering appropriate data members and member functions (use Virtual functions).
-
- ```
graph TD; Shape --> Triangle; Shape --> Rectangle; Shape --> Circle;
```
- The diagram shows a class hierarchy. 'Shape' is the base class at the top. It has three subclasses: 'Triangle', 'Rectangle', and 'Circle'. Arrows point from 'Shape' to each of the three subclasses.
19. WAP to convert meter to centimetre and vice versa, using data conversions and operator overloading
  20. WAP to count digits, alphabets and spaces, stored in a text file, using streams.

**Note:**

1. In total 10 practicals to be implemented. 2 additional practicals may be given by the course

**ARYAVART INTERNATIONAL UNIVERSITY**  
**Tilthai, Dharmanagar, North Tripura**

instructor.

2. This is a suggestive list of programs. However, the instructor may add programs as per the requirement of the course.

**COMPUTER ORGANISATION & ARCHITECTURE LAB**

**Code: 26CS292**

Max Marks: 70

**Exercises in Digital Logic Design:**

1. Implement Logic gates using NAND and NOR gates
2. Design a Full adder using gates
3. Design and implement the 4:1 MUX, 8:1 MUX using gates /ICs.
4. Design and Implement a 3 to 8 decoder using gates
5. Design a 4 bit comparator using gates/IC
6. Design and Implement a 4 bit shift register using Flip flops
7. Design and Implement a Decade counter

**Exercises in Computer Organization:**

8. Implement a C program to convert a Hexadecimal, octal, and binary number to decimal number vice versa.
9. Implement a C program to perform Binary Addition & Subtraction.
10. Implement a C program to perform Multiplication of two binary numbers
11. Implement a C program to perform Multiplication of two binary numbers (signed) using Booth's Algorithms.
12. Implement a C program to perform division of two binary numbers (Unsigned) using restoring division algorithm.
13. Implement a C program to perform division of two binary numbers (Unsigned) using non-restoring division algorithm.

**Note:**

1. In total 15 practicals to be implemented. 2 additional practicals may be given by the course instructor.
2. This is a suggestive list of programs. However, the instructor may add programs as per the requirement of the course.

विद्याधनं सर्वधनप्रधानं

# **MCA**

# **2nd Semester**

## DETAILED SYLLABUS

### DATABASE MANAGEMENT SYSTEM

Code: 26CS302

Max Marks: 70

**Course Objectives:** The objective of the course is to understand database concepts, models, and architectures, design and implement relational databases using SQL and learn normalization, indexing, and transaction management.

#### UNIT I

(10 Hrs)

**Introduction:** An overview of database management system, Characteristics of database approach, DBMS architecture, client/server, data Models, Introduction to Distributed Data processing, schema and instances, data independence.

**Data Modelling using Entity Relationship Model:** Basic introduction about the terminologies like Entity, Entity types, Entity set, Notation for ER diagram, Attributes and keys, Types of attributes (composite, derived and multivalued attributes) and keys (Super Key, candidate key, primary key), Relationships, Relation types, Weak entities, Enhanced E-R, Specialization and Generalization.

#### UNIT II

(13 Hrs)

**Introduction to SQL:** Overview, Characteristics of SQL. Advantage of SQL, SQL data types and literals.

**Types of SQL commands:** DDL, DML, DCL. Basic SQL Queries.

**Logical operators:** BETWEEN, IN, AND, OR and NOT.

**Null Values:** Disallowing Null Values, Comparisons Using Null Values.

**Integrity constraints:** Primary Key, Not NULL, Unique, Check, Referential key.

Introduction to Nested Queries, Correlated Nested Queries, Set-Comparison Operators, Aggregate Operators: The GROUP BY and HAVING Clauses.

**Joins:** Inner joins, Outer Joins, Left outer, Right outer, full outer joins.

**Overview of other SQL Objects:** Views, Sequences, Indexes, Triggers and stored procedure.

#### UNIT III

(10 Hrs)

**Relational Data Models:** Relational model terminology domains, Attributes, Tuples, Relations, Characteristics of relations, Relational constraints Domain constraints, Key constraints and Constraints on null, Relational DB schema. Codd's Rules.

**Relational Algebra:** Basic operations selection and projection.

**Set Theoretic Operations:** Union, Intersection, Set difference and division (Order, Relational calculus: Domain, Tuple, Well Formed Formula, Specification, Quantifiers).

**Join operations:** Inner, Outer, Left outer, Right outer, and Full outer join.

**ER to relational mapping:** Steps to map ER diagram to relational schema.

**Data Normalization:** Functional dependencies, Armstrong's inference rule, & Normalization (Up to BCNF)

#### UNIT IV

(07 Hrs)

**Transaction Processing:** Definition of Transaction, Desirable ACID properties.

**Database recovery and Database Security:** System failure, Backup & recovery Techniques, Authentication, Authorization.

**Overview of Query by Language, No SQL databases.**

#### Text Books:

1. R. Elmars and SB Navathe, "Fundamentals of Database Systems", Pearson, 5th Ed.
2. Singh S.K., "Database System Concepts, design and application", Pearson Education [TB3] TB3.
3. Ramakrishnan and Gherke, "Database Management Systems", TMH.
4. Bipin Desai, "An Introduction to Database Systems", Galgotia Publications, 1991.

#### Suggested Readings:

1. Abraham Silberschatz, Henry Korth, S. Sudarshan, "Database Systems Concepts", 6th Edition, McGraw Hill, 2010.
2. Jim Melton, Alan Simon, "Understanding the new SQL: A complete Guide", Morgan Kaufmann Publishers, 1993.
3. A. K. Majumdar, P. Bhattacharya, "Database Management Systems", TMH, 2017.

**DATA STRUCTURE AND ALGORITHM USING C**

**Code: 26CS301**

Max Marks: 70

**Course Objectives:** The objective of the course is to learn fundamental data structures and their applications, implement algorithms for searching, sorting, and recursion in C and analyze algorithm efficiency and performance.

**UNIT I (14 Hrs)**

- **Linear Data Structures- Static:** Introduction to Algorithms- Attributes, Design Techniques, Time Space Trade Off, Data Structures, Classification and Operations of Data Structures.
- **Arrays:** Single Dimension, Two-Dimension and Introduction to Multi-Dimensional, Memory Representation, Address Calculation, Sparse Matrices- Types, Representation.
- **Searching and Sorting:** Linear and Binary Search, Selection Sort, Bubble Sort, Insertion Sort, Merge Sort, Elementary Comparison of Searching and Sorting Algorithms.
- **Hashing:** Hash Table, Hash Functions, and Collision Resolution.

**UNIT II (10 Hrs)**

- **Linear Data Structures- Dynamic**
- **Introduction:** Dynamic Memory Allocation, Dynamic Memory versus Static Memory Allocation.
- **Linked List Types:** Singly Linked List, Doubly Linked List, Header Linked List, Circular Linked List.
- **Operations:** Creation, Insertion, Deletion, Modification, Searching, Sorting, Reversing, and Merging.

**UNIT III (06 Hrs)**

- **Abstract Data Types:**
- **Stacks:** Introduction, Static and Dynamic Implementation, Operations, Applications- Evaluation and Conversion between Polish and Reverse Polish Notations.
- **Queues:** Introduction, Static and Dynamic Implementation, Operations, Types- Linear Queue, Circular Queue, Doubly Ended Queue, Priority Queue.

**UNIT IV (10 Hrs)**

- **Non-Linear Data Structures:**
- **Introduction to Graphs:** Notations & Terminologies, Representation of Graphs- Adjacency Matrix, Incidence Matrix and Linked Representation.
- **Trees:** Notations & Terminologies, Memory Representation, Binary Trees Types- Complete, Full, Strict, Expression Binary Tree, Tree Traversals (Recursive), Binary Search Tree and Basic Operations.
- **Introduction and Creation (Excluding Implementation):** AVL Tree, Heap Tree, M- Way Tree, and B Tree.

**Text Books:**

1. Schaum's Outline Series, "Data Structures", TMH, Special Indian Ed., Seventeenth Reprint, 2014.
2. Y. Langsam, M. J. Augenstein and A.M. Tanenebaum, "Data Structures using C and C++", Pearson Education India, Second Edition, 2015.
3. D. Samanta, "Classic Data Structures", PHI, Second Edition, 2009.

**Suggested Readings:**

1. Ashok N Kamthane "Introduction to Data Structures in C", Pearson, Third Edition, 2009.
2. E. Horowitz and S. Sahni, "Fundamentals of Data Structures in C". Universities Press, Second edition, 2008.
3. D. Malhotra and N. Malhotra, "Data Structures and Program Design using C", Laxmi Publications, Indian adapted edition from Mercury Learning and Information-USA, First edition, 2018.
4. Y. Kanetkar "Data Structures through C", BPB Publication, Third Edition, 2019.
5. R.F Gilberg, and B A Frouzan- "Data Structures: A Pseudocode Approach with C", Thomson Learning, Second Edition, 2004.
6. A. K. Rath, and A.K. Jagadev, "Data Structures and Program Design Using C", Scitech Publications, Second Edition, 2011.

**THEORY OF COMPUTATION**

**Code: 26CS213**

Max Marks: 70

**Course Objective:** This course introduces formal models of computation, namely, finite automata, pushdown automata, and Turing machines, and their relationships with formal languages. Students will also learn about the limitations of computing machines.

**UNIT I (04 Hrs)**

Languages: Alphabets, strings, languages, basic operations on language, concatenation, union, Kleene star.

**UNIT II (04 Hrs)**

Regular Expressions and Finite Automata: Regular expressions, Deterministic finite automata (DFA).

**UNIT III (10 Hrs)**

Regular Languages: Non-deterministic Finite Automata (NFA), relationship between NFA and DFA, Transition Graphs (TG), properties of regular languages, the relationship between regular languages and finite automata, Kleene's Theorem.

**UNIT IV (06 Hrs)**

Non-Regular Languages and Context Free Grammars: Pumping lemma for regular grammars, Context-Free Grammars (CFG),

**UNIT V (08 Hrs)**

Context-Free Languages (CFL) and PDA: Deterministic and non-deterministic Pushdown Automata (PDA), parse trees, leftmost derivation, pumping lemma for CFL, properties of CFL.

**UNIT VI (08 Hrs)**

Turing Machines and Models of Computations: Turing machine as a model of computation, configuration of a simple Turing machine, Church-Turing Thesis, Universal Turing Machine, decidability, halting problem.

**Suggested Readings:**

1. Cohen, D. I. A. (2011). Introduction to Computer Theory. 2nd edition. Wiley India.
2. Lewis, H.R. & Papadimitriou, H. R. (2002). Elements of the Theory of Computation. 6th edition. Prentice Hall of India (PHI)
3. Goodrich, M., Tamassia, R., & Mount, D.M. (2011). Data Structures and Algorithms Analysis in C++. 2nd edition. Wiley.
4. Gopalkrishnan, G.L. (2019). Automata and Computability: A programmer's perspective. CRC Press.
5. Linz, P. (2016). An Introduction to Formal Languages and Automata. 6th edition. Jones and Bartlett Learning.

विद्याधनं सर्वधनप्रधानं

## INTRODUCTION TO ARTIFICIAL INTELLIGENCE

**Code: 26CS601**

Max Marks: 70

**Course Objective:** Introduce core AI concepts, including problem-solving, knowledge representation, machine learning, and intelligent agent design.

### UNIT I

(10 Hrs)

**Overview of AI:** Introduction to AI, Importance of AI, AI and its related field, AI techniques, and Criteria for success. Problems, problem space and search: Defining the problem as a state space search, Production Systems and its characteristics, Issues in the design of the search programs.

**Heuristic search techniques:** Generate and test, hill climbing, best first search technique, problem reduction, constraint satisfaction.

### UNIT II

(10 Hrs)

**Knowledge Representation:** Definition and importance of knowledge, Knowledge representation, various approaches used in knowledge representation, and issues in knowledge representation.

**Logical Reasoning:** Logical agents, propositional logic, inferences, Syntax and semantics of First Order Logic, Inference in First Order Logic Knowledge Base, forward chaining, backward chaining, unification, resolution, Expert system: Case study of Expert system in PROLOG.

### UNIT III

(10 Hrs)

**Handling Uncertainty:** Non-Monotonic Reasoning, Probabilistic reasoning, Bayes 'Theorem, Certainty factors and Rule-based Systems, Bayesian Networks, Dempster-Shafer Theory, Introduction to Fuzzy logic. Fuzzy set definition & types. Membership functions. Designing a fuzzy set for a given application.

**Natural Language Processing:** Introduction, Syntactic Processing, Semantic Processing, Pragmatic Processing.

### UNIT IV

(10 Hrs)

**Learning:** Introduction to Learning, Rote Learning, learning by taking advice, learning in problem solving, learning from examples: Induction, Explanation-based Learning, Discovery, Analogy, Neural Networks, and Genetic Learning.

### Text Books:

1. Rich and Knight, "Artificial Intelligence", Tata McGraw-Hill, 1992.
2. Stuart Russell and Peter Norvig, "Artificial Intelligence: A Modern Approach", Prentice Hall, Second Edition (Indian Reprint: Pearson Education).

### Suggested Readings:

1. Ivan Bratko, "Prologue Programming for AI", Addison-Wesley.
2. George F. Luger, "Artificial Intelligence" Pearson Education.
3. Ben Coppin, "Artificial Intelligence Illuminated" Jones and Bartlett Publishers.

विद्याधनं सर्वधनप्रधानं

**ARYAVART INTERNATIONAL UNIVERSITY**  
**Tilthai, Dharmanagar, North Tripura**

**ENVIRONMENTAL STUDIES**

**Code: 26GN201**

Max Marks: 70

**Course Objectives:** The course will empower the students by gaining in-depth knowledge on natural processes that sustain life and govern economy, predicting the consequences of human actions on the web of life, global economy and quality of human life, developing critical thinking for shaping strategies (scientific, social, economic and legal) for environmental protection and conservation of biodiversity, social equity and sustainable development, acquiring values and attitudes towards understanding complex environmental economic-social challenges, and participating actively in solving current environmental problems and preventing the future ones and adopting sustainability as a practice in life, society and industry.

**UNIT I: Introduction to Environmental Studies (5 Hrs)**

- Environmental studies: Nature, Scope and Importance; Components of environment: atmosphere, hydrosphere, lithosphere, and biosphere; Concept of sustainability and sustainable development.
- Emergence of environmental issues: Climate change, Global warming, Ozone layer depletion, Acid rain, etc.; International agreements and programmes: Earth Summit, UNFCCC, Montreal and Kyoto protocols, Convention on Biological Diversity (CBD), Ramsar Convention, UNEP, CITES, etc.

**UNIT II: Ecosystems and Natural Resources: (5 Hrs)**

- Definition and concept of Ecosystem; Structure of ecosystem (biotic and abiotic components); Functions of Ecosystem: Physical (energy flow), Biological (food chains, food web, ecological succession), ecological pyramids and homeostasis; Types of Ecosystems: Tundra, Forest, Grassland, Desert, Aquatic (ponds, streams, lakes, rivers, oceans, estuaries); importance and threats with relevant examples from India.
- Ecosystem services (Provisioning, Regulating, Cultural, and Supporting); Ecosystem preservation and conservation strategies; Basics of Ecosystem restoration.
- Energy resources: Renewable and non-renewable energy sources; Use of alternate energy sources; Growing energy needs; Energy contents of coal, petroleum, natural gas and bio gas; Agro-residues as a biomass energy source.

**UNIT III: Biodiversity and Conservation (5 Hrs)**

- Definition of Biodiversity; Levels of biological diversity: genetic, species and ecosystem diversity.
- India as a mega-biodiversity nation; Biogeographic zones of India; Biodiversity hotspots; Endemic and endangered species of India; IUCN Red list criteria and categories.
- Value of biodiversity: Ecological, economic, social, ethical, aesthetic, and informational values of biodiversity with examples.
- Threats to biodiversity: Habitat loss, degradation, and fragmentation; Poaching of wildlife; Man-wildlife conflicts; Biological invasion with emphasis on Indian biodiversity; Current mass extinction crisis.
- Biodiversity conservation strategies: in-situ and ex-situ methods of conservation (National Parks, Wildlife Sanctuaries, and Biosphere reserves).

**UNIT IV: Environmental Pollution and Control Measures: (5 Hrs)**

- Environmental pollution (Air, water, soil, thermal, and noise): causes, effects, and controls; Primary and secondary air pollutants; Air and water quality standards.
- Nuclear hazards and human health risks.
- Solid waste management: Control measures for various types of urban, industrial waste, Hazardous waste, E-waste, etc.; Waste segregation and disposal.

**Text Books:**

1. Sanjay Kumar Batra, Kanchan Batra, Harpreet Kaur; "Environmental Studies"; Taxmann's, Fifth Edition.
2. M. M. Sulphery; "Introduction to Environment Management"; PHI Learning, 2019.
3. S. P. Mishra, S. N. Pandey; "Essential Environmental Studies"; Ane Books Pvt. Ltd.; Sixth Edition.

**ARYAVART INTERNATIONAL UNIVERSITY**  
**Tilthai, Dharmanagar, North Tripura**

**Suggested Readings:**

1. Asthana, D. K. (2006). "Text Book of Environmental Studies". S. Chand Publishing.
2. Basu, M., Xavier, S. (2016). "Fundamentals of Environmental Studies", Cambridge University Press, India.
3. Bharucha, E. (2013). "Textbook of Environmental Studies for Undergraduate Courses". Universities Press.
4. Mahapatra, R., Jeevan, S. S., Das, S. (Eds) (2017). "Environment Reader for Universities", Centre for Science and Environment, New Delhi.
5. Masters, G. M. & Ela, W. P. (1991). "Introduction to environmental engineering and science". Englewood Cliffs, NJ: Prentice Hall.
6. Odum, E. P., Odum, H. T. & Andrews, J. (1971). "Fundamentals of Ecology". Philadelphia: Saunders.
7. Sharma, P. D. & Sharma, P. D. (2005). "Ecology and Environment". Rastogi Publications.

**HEALTH & WELLNESS**

**Code: 26GN202**

Max Marks: 70

**Course Objectives:** The objective of the course is to introduce learners to the concept of health and wellness and its relevance in daily life; to introduce the relation between mind and body and its relevance; and to introduce learners to health behaviour and the promotion of human strengths for well-being.

**UNIT I: INTRODUCTION TO HEALTH & WELLNESS**

**(10 Hrs)**

- Definition of health- WHO definition
- Importance of health in everyday life
- Components of health- physical, social, mental, spiritual and its relevance
- Concept of wellness
- Mental Health & wellness
- Determinants of health behaviours
- Using the mass media for health promotion

**UNIT II: MIND – BODY AND WELL-BEING**

**(10 Hrs)**

- Mind- Body connection in health- concept and relation
- Implications of mind-body connections.
- Wellbeing- why it matters?
- Digital wellbeing
- Understanding health beliefs, and perspectives of indigenous people about Assam and North East India
- Promoting Human strengths and life enhancement: Classification of human strengths and virtues; cultivating inner strengths: Hope and optimism

विद्याधनं सर्वधनप्रधानं

**Suggested Readings:**

1. Carr, A. (2004). Positive Psychology: The science of happiness and human strength. UK: Routledge.
2. Forshaw, M. (2003). Advanced psychology: Health psychology. London: Hodder and Stoughton. Hick,
3. J.W. (2005). Fifty signs of Mental Health. A Guide to understanding mental health. Yale University Press.
4. Snyder, C.R., & Lopez, S.J. (2007). Positive psychology: The scientific and practical explorations of human strengths. Thousand Oaks, CA: Sage.

**ANDROID APP DEVELOPMENT**

**Code: 26CS231**

Max Marks: 70

**Course Objectives:** This course is concerned with the development of applications on mobile and wireless computing platforms. Android will be used as a basis for teaching programming techniques and design patterns related to the development of standalone applications and mobile portals to enterprise and commerce systems. Emphasis is placed on the processes, tools and frameworks required to develop applications for current and emerging mobile computing devices. Students will work at all stages of the software development life-cycle from inception through to implementation and testing. In doing so, students will be required to consider the impact of user characteristics, device capabilities, networking infrastructure and deployment environment, in order to develop software capable of meeting the requirements of stakeholders.

**UNIT I**

**(12 Hrs)**

**Introduction:** Brief history of mobile applications, Different types of mobile applications, Brief history of Android, Introduction to Android Development Environment, Android Application

**Design Essentials:** Anatomy of an Android applications, Creating First Android Application, Creating Android project, Project organization, setting up real Android device, setting up Android emulator, developing simple user interface, Running your first application

Android terminologies, Application Context, Activities, Services, Intents, Receiving and Broadcasting Intents, Android Manifest File and its common settings, Using Intent Filter, Permissions.

**UNIT II**

**(10 Hrs)**

**User Interface in Android:** Adaptive and responsive user interfaces, User Input Controls, Menus, Screen Navigation, RecyclerView, Drawable, Themes and Styles, Fragments, Fragment Life Cycle, Introduction to Material Design.

**Android Application Components:** App Widgets, Processes and Threads, User Interface Components, Views and layouts, Input controls, Input Events, Settings, Dialogs, Menus, Notifications, Toasts, Testing the user interface

**UNIT III**

**(8 Hrs)**

**Background tasks:** AsyncTask, AsyncTaskLoader, Connecting App to Internet, Broadcast receivers, Services, Notifications, Alarm managers.

**Sensor, Location, and Maps:** Sensor Basic, Motion and Position Sensors, Location services, Google Maps API, Google Places API

**UNIT IV**

**(10 Hrs)**

**Working with data in Android:** Shared Preferences, App Settings, SQLite primer, Store data using SQLite database, Content Providers, Content Resolver, Loader

**Publishing Your App:** Preparing for publishing, signing, and preparing the graphics, and publishing to the Android Market

**Using Common Android APIs:** Using Android Data and Storage APIs, managing data using SQLite, Sharing Data between Applications with Content Providers, Using Android Networking APIs, using Android Web APIs, Using Android Telephony APIs, Deploying Android Application to the World.

**Text Books:**

1. Lauren Darcey and Shane Conder, "Android Wireless Application Development", Pearson Education, 2nd ed. (2011)
2. Wei-Meng Lee, "Beginning Android 4 Application Development", Wiley India Pvt. Ltd.
3. J. F. DiMarzio, "Android: A Programmers Guide", McGraw Hill Education (India) Private Limited.

**Suggested Readings:**

1. Paul Deitel "Android for Programmers: An App-Driven Approach" 1st Edition, Pearson India.
2. Wei-Meng Lee, "Beginning Android Application Development", Wiley Publishing

**ARYAVART INTERNATIONAL UNIVERSITY**  
**Tilthai, Dharmanagar, North Tripura**

**ANDROID APP DEVELOPMENT LAB**

**Code: 26CS294**

Max Marks: 70

**List of Practical:**

1. Set up ANDROID STUDIO using ECLIPSE IDE.
2. Build and Run first ANDROID program in Emulator.
3. Create an APP for currency converter.
4. Create an APP to registration form.
5. Create a calculator using ANDROID APP.
6. Create an APP to insert data from registration form into database.
7. Create an APP to conduct online quiz.
8. Create Slide show of multiple images using ANDROID APP.
9. Develop an APP which plays video files.
10. Create an App to identify the current location using GOOGLE MAP API.

**NATURAL LANGUAGE PROCESSING**

**Code: 26CS533**

Max Marks: 70

**Course Objectives:** This course will enable students to, learn the importance of natural language Modelling, understand the Applications of natural language processing, study spelling, error detection and correction methods and parsing techniques in NLP and illustrate the information retrieval models in natural language processing.

**UNIT I**

**(08 Hrs)**

**Introduction:** What is Natural Language Processing? Origins of NLP, Language and Knowledge, The Challenges of NLP, Language and Grammar, Processing Indian Languages, NLP Applications.  
**Language Modelling:** Statistical Language Model - N-gram model (unigram, bigram), Paninion Framework, Karaka theory.

**UNIT II**

**(08 Hrs)**

**Word Level Analysis:** Regular Expressions, Finite-State Automata, Morphological Parsing, Spelling Error Detection and Correction, Words and Word Classes, Part-of-Speech Tagging.  
**Syntactic Analysis:** Context-Free Grammar, Constituency, Top-down and Bottom-up Parsing, CYK Parsing.

**UNIT III**

**(08 Hrs)**

**Naive Bayes, Text Classification and Sentiment:** Naive Bayes Classifiers, Training the Naive Bayes Classifier, Worked Example, Optimising for Sentiment Analysis, Naive Bayes for Other Text Classification Tasks, Naive Bayes as a Language Model.

**UNIT IV**

**(08 Hrs)**

**Information Retrieval:** Design Features of Information Retrieval Systems, Information Retrieval Models - Classical, Non-classical, Alternative Models of Information Retrieval - Custer model, Fuzzy model, LSTM model, Major Issues in Information Retrieval. Lexical Resources: WordNet, FrameNet, Stemmers, Parts-of-Speech Tagger, Research Corpora.

**UNIT V**

**(08 Hrs)**

**Machine Translation:** Language Divergences and Typology, Machine Translation using Encoder-Decoder, Details of the Encoder-Decoder Model, Translating in Low-Resource Situations, MT Evaluation, Bias and Ethical Issues.

**ARYAVART INTERNATIONAL UNIVERSITY**  
**Tilthai, Dharmanagar, North Tripura**

**Suggested Readings:**

1. Tanveer Siddiqui, U.S. Tiwary, "Natural Language Processing and Information Retrieval", Oxford University Press.
2. Daniel Jurafsky, James H. Martin, "Speech and Language Processing, An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition", Pearson Education, 2023.
3. Akshay Kulkarni, Adarsha Shivananda, "Natural Language Processing Recipes – Unlocking Text Data with Machine Learning and Deep Learning using Python", Apress, 2019.
4. T V Geetha, "Understanding Natural Language Processing – Machine Learning and Deep Learning Perspectives", Pearson, 2024.
5. Gerald J. Kowalski and Mark.T. Maybury, "Information Storage and Retrieval systems", Kluwer Academic Publishers.

**NATURAL LANGUAGE PROCESSING LAB**

**Code: 26CS595**

**Max. Marks: 70**

**List of Experiments:**

1. Installation and exploring features of NLTK and spaCy tools. Download Word Cloud and few corpora.
2. (i) Write a program to implement word Tokenizer, Sentence and Paragraph Tokenizers.  
(ii) Check how many words are there in any corpus. Also check how many distinct words are there?
3. (i) Write a program to implement both user-defined and pre-defined functions to generate
  - (a) Uni-grams
  - (b) Bi-grams
  - (c) Tri-grams
  - (d) N-grams  
(ii) Write a program to calculate the highest probability of a word (w2) occurring after another word(w1).
4. (i) Write a program to identify the word collocations.  
(ii) Write a program to print all words beginning with a given sequence of letters.  
(iii) Write a program to print all words longer than four characters.
5. (i) Write a program to identify the mathematical expression in a given sentence.  
(ii) Write a program to identify different components of an email address.
6. (i) Write a program to identify all antonyms and synonyms of a word.  
(ii) Write a program to find hyponymy, homonymy, polysemy for a given word.
7. (i) Write a program to find all the stop words in any given text.  
(ii) Write a function that finds the 50 most frequently occurring words of a text that are not stopwords.
8. Write a program to implement various stemming techniques and prepare a chart with the performance of each method.
9. Write a program to implement various lemmatization techniques and prepare a chart with the performance of each method.
10. (i) Write a program to implement Conditional Frequency Distributions (CFD) for any corpus.  
(ii) Find all the four-letter words in any corpus. With the help of a frequency distribution (FreqDist), show these words in decreasing order of frequency.  
(iii) Define a conditional frequency distribution over the names corpus that allows you to see which initial letters are more frequent for males versus females.
11. (i) Write a program to implement Part-of-Speech (PoS) tagging for any corpus.  
(ii) Write a program to identify which word has the greatest number of distinct tags? What are they, and what do they represent?  
(iii) Write a program to list tags in order of decreasing frequency and what do the 20 most frequent tags represent?  
(iv) Write a program to identify which tags are nouns most commonly found after? What do these tags represent?

**ARYAVART INTERNATIONAL UNIVERSITY**  
**Tilthai, Dharmanagar, North Tripura**

12. Write a program to implement TF-IDF for any corpus.
13. Write a program to implement chunking and chunking for any corpus.
14. (i) Write a program to find all the mis-spelled words in a paragraph.  
(ii) Write a program to prepare a table with frequency of mis-spelled tags for any given text.
15. Write a program to implement all the NLP Pre-Processing Techniques required to perform further NLP tasks.

**Case Studies: (At Least any one Case Study has to be performed)**

- Case Study-1. Write a program to implement Named Entity Recognition (NER) for any corpus.
- Case Study-2. Write a program to perform Auto-Correction of spellings for any text.
- Case Study-3. Check for all positive words in a news article/ any text.

**DATABASE MANAGEMENT SYSTEM LAB**

**Code: 24CS392**

Max. Marks: 70

**Core Practicals (Implement All the mentioned practicals)**

**The following are two suggestive databases. The students may use either or both databases for their core practicals. However, the instructor may provide any other databases for executing these practices.**

**1. COLLEGE DATABASE:**

STUDENT (USN, SName, Address, Phone, Gender) SEMSEC (SSID, Sem, Sec)

CLASS (USN, SSID)

SUBJECT(Subcode, Title, Sem, Credits)

IA MARKS (USN, Subcode, SSID, Test1, Test2, Test3, Final IA)

**2. COMPANY DATABASE:**

EMPLOYEE (SSN, Name, Address, Sex, Salary, SuperSSN, DNo)

DEPARTMENT (DNo, DName, MgrSSN, MgrStartDate)

DLOCATION (DNo, DLoc)

PROJECT (PNo, PName, PLocation, DNo)

WORKS\_ON (SSN, PNo, Hours)

1. Draw an E-R diagram from given entities and their attributes.
2. Convert the E-R diagram in to a Relational model with proper constraints.
3. Write queries to execute following DDL commands:  
CREATE: Create the structure of a table with at least five columns  
ALTER: Change the size of a particular column.  
Add a new column to the existing table.  
Remove a column from the table.  
DROP: Destroy the table along with its data.
4. Write queries to execute following DML commands:  
INSERT: Insert five records in each table.  
UPDATE: Modify data in single and multiple columns in a table.  
DELETE: Delete selective and all records from a table
5. Write queries to execute following DML command:  
SELECT: Retrieve the entire contents of the table.  
Retrieve the selective contents (based on provided conditions) from a table.  
Retrieve contents from a table based on various operators i.e. string operators, logical operators, conditional operators and Boolean operators.  
Sort the data in ascending and descending order in a table on the basis of one column or more than one column.
6. Create table using following integrity constraints:
  - PrimaryKey
  - Unique Key
  - Not Null
  - Check Default

**ARYAVART INTERNATIONAL UNIVERSITY**  
**Tilthai, Dharmanagar, North Tripura**

- Foreign Key
- 7. Write queries to execute following Aggregate functions:  
Sum, Avg, Count, Minimum and Maximum value of a numeric column of a table using aggregate function.
- 8. Retrieve data from a table using alias names.
- 9. Retrieve data of a table using nested queries.
- 10. Retrieve data from more than one table using inner join, left outer, right outer and full outer Joins.
- 11. Create view from one table and more than one table.
- 12. Create index on a column of a table.

**Application Based Practicals**

13. Consider the Insurance company's Database given below. The primary keys are underlined and the data types are specified.  
PERSON (driver\_id#: string, name: string, address: string)  
CAR (reg no : string, model: string, year: int)  
ACCIDENT (report\_number:int, acc\_date: date, location: string)  
OWNS (driver\_id#: string, reg no: string)  
PARTICIPATED (driver\_id#: string, reg no: string, report\_number: int, damage\_amount: number (10, 2) )
  - (i) Create the above tables by properly specified the primary key and the foreign key
  - (ii) Enter at least five tuples for each relation
  - (iii) Demonstrate how you can
    - a) Update the damage amount for the car with a specific reg no, the accident with report number 12 to 25000.
    - b) Add a new accident to the database.
  - (iv) Find the total number of people who owned cars that were involved in accident in 2002.
  - (v) Find the number of accident in which cars belonging to a specific models were involved.
14. Consider the following schema of a library management system. Write the SQL queries for the questions given below:  
Student (Stud\_no: integer, Stud\_name: string)  
Membership (Mem\_no: integer, Stud\_no: integer)  
Book (book\_no: integer, book\_name: string, author: string)  
Iss\_rec (iss\_no: integer, iss\_date: date, Mem\_no: integer, book\_no: integer)
  - (i) Create the tables with the appropriate integrity constraints.
  - (ii) Insert around 10 records in each of the tables.
  - (iii) Display all records for all tables.
  - (iv) List all the student names with their membership numbers.
  - (v) List all the issues for the current date with student and Book names.
  - (vi) List the details of students who borrowed book whose author is Elmarsi & Navathe.
  - (vii) Give a count of how many books have been bought by each student.
  - (viii) Give a list of books taken by student with stud\_no as 1005.
  - (ix) Delete the List of books details which are issued as of today.
  - (x) Create a view which lists out the iss\_no, iss\_date, stud\_name, bookname.
15. Use the relations below to write SQL queries to solve the business problems specified.  
CLIENT (clientno#, name, client\_referred\_by#)  
ORDER (orderno#, clientno#, order\_date, empid#)  
ORDER\_LINE (orderno#, orderlinenumber#, item\_number#, no\_of\_items, item\_cost, shipping\_date)  
ITEM (item\_number#, item\_type, cost)  
EMPLOYEE (empid#, emp\_type#, deptno, salary, first name, last name)  
Notes:
  - a. Column followed by # is the primary key of the table.
  - b. Each client may be referred by another client. If so, the client number of the referring client is stored in referred\_by.
  - c. The total cost for a particular orderline=no\_of\_items\*item\_cost.c.
16. Write queries for the following:
  - (i) Create all the above tables.
  - (ii) Insert at least five records.

**ARYAVART INTERNATIONAL UNIVERSITY**  
**Tilthai, Dharmanagar, North Tripura**

- (iii) Display all the rows and columns in the CLIENT table. Sort by client name in reverse alphabetical order.
- (iv) Display the item number and total cost for each order line (total cost = no of items X item cost). Name the calculated column TOTAL COST.
- (v) Display all the client numbers in the ORDER table. Remove duplicates.
- (vi) Display the order number and client number from the ORDER table. Out put the result in the format. Client <clientno> ordered <orderno>
- (vii) Display full details from the ORDER\_LINE table where the item number is (first condition) between 1 and 200 (no > or < operators) OR the item number is greater than 1000 AND (second condition) the item cost is not in the list 1000, 2000, 3000 OR the order number is not equal to 1000.
- (viii) Display the client name and order date for all orders.
- (ix) Repeat query (6) but also display all clients who have never ordered anything.
- (x) Display the client name and order date for all orders using the join keywords.
- (xi) Display the client name and order date for all orders using the JOIN method.
- (xii) Display the client number, order date and shipping date for all orders where the shipping date is between three and six months after the order date.
- (xiii) Display the client number and name and the client number and name of the person who referred that client.
- (xiv) Display the client name in upper case only and in lower case only.
- (xv) Display the second to fifth characters in each client name.

**Note:**

1. In total 15 practicals to be implemented. 2 additional practicals may be given by the course instructor.
2. This is a suggestive list of programs. However, the instructor may add programs as per the requirements of the course.

**ARYAVART**  
**INTERNATIONAL UNIVERSITY**



विद्याधनं सर्वधनप्रधानं

**ARYAVART INTERNATIONAL UNIVERSITY**  
**Tilthai, Dharmanagar, North Tripura**

**DATA STRUCTURE USING C LAB**

**Code: 26CS391**

Max. Marks: 70

**Core Practicals (Implement minimum 8 out of 10 practicals)**

1. WAP to implement following operation on one dimensional array (i) Insertion (ii) Deletion (iii) Traversal (iv) Reverse (v) Merge.
2. WAP to Sort an array using menu driven:  
(i) BUBBLE SORT (ii) MERGE SORT (iii) INSERTION SORT (iv) SELECTION SORT.
3. WAP to implement a Singly Linked List.
4. WAP to implement a Circular Linked Lists.
5. WAP to implement Doubly Linked Lists.
6. Write a menu driven program to implement (i) Static Stack (ii) Dynamic Stack.
7. WAP to implement a: (i) Static (ii) Dynamic Circular Queue.
8. WAP to implement a: (i) Static (ii) Dynamic De-Queue..
9. Implement recursive algorithms for the following operations on Binary Search Tree.  
(i) Insertion  
(ii) Searching
10. Implement recursive algorithms for BST traversal- Inorder, Preorder, Postorder.

**Application Based Practicals (Implement minimum 5 out of 8 practicals)**

11. WAP to search & display the location of an element specified by the user, in an array using: (i) Linear Search (ii) Binary Search technique.
12. WAP to accept a matrix from user, find out matrix is sparse or not and convert into triplex matrix.
13. WAP to implement Polynomial addition operation using linked list.
14. Write a C program to create two linked lists from a given list in following way:  
INPUT List:- 1 2 3 4 5 6 7 8 9 10 OUTPUT:-  
First List:- 1 3 5 7 9  
Second List:- 2 4 6 8 10
15. WAP to implement Student Database using Linked List with the following structure:
  - Name
  - Rollno
  - Marks of 5 subjects
  - Average
  - Result, If the average < 50, then print 'Fail', otherwise 'Pass'
16. Write a program to convert Infix to equivalent (i) Prefix expression (ii) Postfix expression
17. Write a program to evaluate (i) Prefix Expression (ii) Postfix Expression using stack.
18. Let us assume a Patient's coupon generator for the Doctors' clinic. The patients are given the coupons on first-come-first-serve basis. After the visit of a patient, patient-ID is kept stack-wise. At the end of the day, the count is generated from the stack. Construct a menu-based program for patients' coupons generator using an appropriate data structure.
19. WAP to implement an expression tree. (For example:  $(a + b / (c * d) - e)$  ).
20. Sometimes a program requires two stacks containing the same type of items. Suppose two stacks are stored in separate arrays, then one stack might overflow while there is considerable unused space in the other. A neat way to avoid this problem is to put all spaces in one stack and let this stack grow from one end of the array, and the other stack starts from the other end and grows in the opposite direction, i.e., toward the first stack. In this way, if one stack turns out to be large and the other small, then they will still both fit, and there will be no overflow until all space is used. Declare a new structure that includes these two stacks and perform various stack operations.

**Note:**

1. In total 15 practicals to be implemented. 2 additional practicals may be given by the course instructor.
2. This is a suggestive list of programs. However, the instructor may add programs as per the requirement of the course.

# **MCA**

# **3rd Semester**

## Detailed Syllabus

### CRYPTOGRAPHY

**Code: 26CS701**

Max Marks: 70

**Course Objectives:** The objective of the course is to study the fundamentals of cryptographic techniques and protocols for ensuring confidentiality, integrity, and security in digital communication.

**UNIT I: Mathematical Foundations Number Theory (10 Hrs)**

Fermat's theorem - Cauchy's theorem - Chinese remainder theorem - Primality testing algorithm - Euclid's algorithm for integers - quadratic residues - Legendre symbol - Jacobi symbol.

**UNIT II: Classical Cryptosystems (10 Hrs)**

Cryptography and cryptanalysis - Classical Cryptography - different type of attack: CMA - CPA - CCA - Shannon perfect secrecy - OTP - Pseudo random bit generators - stream ciphers and RC4.

**UNIT III: Symmetric Key Ciphers Block ciphers (04 Hrs)**

Modes of operation - DES and its variants - finite fields ( $2^n$ ) - AES - linear and differential cryptanalysis.

**UNIT IV: Asymmetric Key Ciphers (10 Hrs)**

One-way function - trapdoor one-way function - Public key cryptography - RSA cryptosystem - Diffie-Hellman key exchange algorithm - ElGamal Cryptosystem

**UNIT V: Message Authentication (6 Hrs)**

Cryptographic hash functions - secure hash algorithm - Message authentication - digital signature - RSA digital signature.

**Text Book:**

1. Stinson. D., "Cryptography: Theory and Practice", Third Edition, Chapman & Hall/CRC, 2012.

**Suggested Readings:**

1. W. Mao, "Modern Cryptography: Theory & Practice", Pearson Education, 2010.
2. W. Stallings, "Cryptography and Network Security Principles and practice", Fifth Edition, Pearson Education Asia, 2013.
3. Behrouz A. Forouzan, Debdeep Mukhopadhyay, "Cryptography and Network Security", Second Edition, Tata McGraw Hill, 2013.
4. Thomas Koshy, "Elementary Number Theory with Applications", Elsevier India, 2005.

विद्याधनं सर्वधनप्रधानं

**DESIGN AND ANALYSIS OF ALGORITHMS**

**Code: 26CS602**

Max Marks: 70

**Course Objectives:** The objective of the course is to design efficient algorithms, analyze their time and space complexity, and understand algorithmic strategies such as divide and conquer, greedy, dynamic programming, and backtracking.

**UNIT I: Introduction**

**(08 Hrs)**

Algorithm - Pseudo Code for Expressing Algorithms - Performance Analysis- Space Complexity - Time Complexity- Asymptotic Notation - Big Oh Notation - Omega Notation - Theta Notation and Little Oh Notation. - Recurrences - Substitution method, Recursion-tree method, Master method.

**UNIT II: Disjoint Sets, Divide and Conquer**

**(08 Hrs)**

Disjoint Sets: Disjoint Set Operations - Union and Find Algorithms.  
Divide and Conquer: General Method - Applications-Binary Search – Quick Sort - Merge Sort-Strassen's Matrix Multiplication.

**UNIT III: Dynamic Programming**

**(08 Hrs)**

General Method –Applications-Matrix Chain Multiplication - Optimal Binary Search Trees - 0/1 Knapsack Problem - All Pairs Shortest Path Problem - Travelling Sales Person Problem – Reliability Design Problem.

**UNIT IV: Greedy Method and Backtracking**

**(08 Hrs)**

Greedy Method: General Method –Applications- Job Sequencing with Deadlines - Knapsack Problem - Minimum Cost Spanning Trees - Single Source Shortest Path Problem - Backtracking: General Method – Applications-N-Queens Problem - Sum of Subsets Problem - Graph Colouring - Hamiltonian Cycles.

**UNIT V: Branch and Bound, Np-Hard and Np-Complete Problems**

**(08 Hrs)**

Branch and Bound: General Method - Applications - Travelling Sales Person Problem - 0/1 Knapsack Problem- LC Branch and Bound Solution - FIFO Branch and Bound Solution.  
NP Hard and NP-Complete Problems: Basic Concepts - Non-deterministic algorithms - NP – Hard and NP Complete Classes - Cook's Theorem.

**Text Books:**

1. Ellis Horowitz, Satraj Sahni, Sanguthevar Rajasekharan, "Fundamentals of Computer Algorithms", Universities Press, 2nd Edition, 2015.
2. Alfred V. Aho, John E. Hopcroft, Jeffrey D, "The Design And Analysis Of Computer Algorithms", Pearson India, 1st Edition, 2013.

**Suggested Readings:**

1. Knuth Donald E, "Art of Computer Programming: Fundamental Algorithms Volume 1 - Fundamental Algorithms", Third Edition, Pearson Publishers, 2011.
2. Levitin A, "Introduction to the Design and Analysis of Algorithms", Pearson Education, 3rd Edition, 2012.
3. Thomas H. Cormen, C.E. Leiserson, R L.Rivest and C. Stein, Introduction to Algorithms , Third edition, MIT Press, 2009.
4. Jon Kleinberg, Éva Tardos , Algorithm Design, Pearson education, 2014.

## INTRODUCTION TO MACHINE LEARNING

**Code:** 26CS732

**Max Marks:** 70

**Course Objectives:** The course is designed to understand machine learning and its perspectives, to understand and apply the concepts of machine learning algorithms, design and implement supervised machine learning algorithms for real-world applications, appreciate the underlying mathematical relationships within and across the machine learning algorithms and develop ability to solve problems by implementing machine learning.

### UNIT I

(10 Hrs)

**Overview and Introduction to Machine Learning:** Data Science, AI & ML, Introduction of Machine intelligence and its applications, Machine learning concepts, Components of a learning problem, supervised, unsupervised, and reinforcement learning, inductive learning, and deductive learning.

### UNIT II

(10 Hrs)

**Foundations of Machine Learning:** Hypothesis Space and Inductive Bias, feature selection. Classification, regression linear and polynomial regression, logistic regression, decision tree, random forest, naïve bayes, SVM. **Clustering and dimensionality Reduction:** Adaptive hierarchical clustering, SVD PCA, K-means, association analysis, apriori, hidden Markov model.

### UNIT III

(10 Hrs)

**Reinforcement learning:** Elements of Reinforcement Learning, Characteristics of reinforcement learning, various techniques used in reinforcement, Model-Based Learning, Temporal Difference Learning, Markov decision process, Deep Learning.

### UNIT IV

(10 Hrs)

**Learning with Neural Networks:** Introduction to Artificial Neuron, Architectures, Learning Methods, Taxonomy of NN Systems, Single-Layer NN System, Applications. Back Propagation Network: Background, Back-Propagation Learning, Back-Propagation Algorithm.

### Text Books:

1. E. Alpaydin, "Introduction to Machine Learning", Prentice Hall of India.
2. Tom M. Mitchell, "Machine Learning", McGraw-Hill Education (India) Private Limited.
3. Stephen Marsland, "Machine Learning: An Algorithmic Perspective", CRC Press.
4. Bishop C., "Pattern Recognition and Machine Learning", Berlin: Springer-Verlag.

विद्याधनं सर्वधनप्रधानं

**DATA WAREHOUSING & MINING**

**Code: 26CS734**

Max Marks: 70

**Course Objective:** The objective of the course is to learn methods for storing, managing, and analyzing large volumes of data using data warehousing and mining techniques for informed decision-making.

**UNIT I: Introduction to Data Warehousing and ETL tools (10 Hrs)**

Overview, Difference between Database System and Data Warehouse, The Compelling Need for data warehousing, Data warehouse – The building Blocks: Defining Features, data warehouses and data marts, overview of the components, three-tier architecture, Metadata in the data warehouse.

Defining the business requirements: Dimensional analysis, information packages – a new concept, requirements gathering methods, requirements definition: scope and content.

**UNIT II: Dimensional Modelling and OLAP (10 Hrs)**

Principles of Dimensional Modelling: Objectives, From Requirements to data design, Multi-Dimensional Data Model, Schemas: the STAR schema, the Snowflake schema, the fact constellation schema.

OLAP in the Data Warehouse: Demand for Online Analytical Processing, limitations of other analysis methods, OLAP definitions and rules, OLAP characteristics, major features and functions, hyper cubes.

OLAP Operations: Drill-down and roll-up, slice-and-dice, pivot or rotation, OLAP models, overview of variations, the MOLAP model, the ROLAP model, the DOLAP model, ROLAP versus MOLAP, OLAP implementation considerations. Query and Reporting, Executive Information Systems (EIS), Data Warehouse, and Business Strategy.

**UNIT III: Data mining and data pre-processing (10 Hrs)**

Data mining: Introduction, What kind of data can be mined, What kind of patterns to be mined, Which technologies are used, What kinds of applications are targeted, Major issues in data mining.

Data pre-processing: Overview of Data pre-processing, data cleaning, data integration, data reduction, data transformation and data discretization, exploring data using the IRIS datasets. Introduction to Apriori algorithm for association mining rules.

**UNIT IV: Data mining applications, and Data mining Tools (10 Hrs)**

Applications of data mining: Data mining for retail and telecommunication industries, data mining and recommender systems.

Introduction to data mining tools (open source): Weka, Rapid Miner, IBM Watson for classification and clustering algorithms using IRIS Datasets.

**Text Books:**

1. Kamber and Han, "Data Mining Concepts and Techniques", Third edition, Hartcourt India P.Ltd.,2012.
2. Pang-Ning Tan, Michael Steinbach, Vipin Kumar, "Introduction to data mining", Pearson Education, 2006
3. Paul Raj Poonia, "Fundamentals of Data Warehousing", John Wiley & Sons, 2004

**Suggested Readings:**

1. Ashok N. Srivastava, Mehran Sahami, "Text Mining Classification, Clustering, and Applications", Published by Chapman and Hall/CRC1st Edition, June 23, 2009
2. Ian H., Eibe Frank, Mark A. Hall, Christopher Pal. "Data Mining: Practical Machine Learning Tools and Techniques." Published by Morgan Kaufmann; 4th edition, December 1, 2016
3. G. K. Gupta, "Introduction to Data Mining with Case Studies", PHI, 2006.
4. Alex Berson and Stephen J. Smith, "Data Warehousing, Data Mining & OLAP", Tata McGraw-Hill, 1 July 2017 RB5. Shmueli, "Data Mining for Business Intelligence: Concepts, Techniques and Applications in Microsoft Excel with XL Miner", Wiley Publications

**INTERNET OF THINGS**

**Code: 26CS731**

Max Marks: 70

**Course Objectives:** This course focuses on the latest microcontrollers with application development, product design and prototyping. The Internet of Things (IoT) is the next wave, world is going to witness. Internet of Things is a term given to the attempt of connecting objects to the internet and also to each other, allowing people and objects themselves to analyse data from various sources in real-time and intelligently take necessary actions.

**UNIT I (10 Hrs)**

**Introduction:** Definition, Functional requirements, Characteristics, Foundations, architectures, challenges and issues, Physical design of IoT, Logical design of IoT, Web 3.0 of IoT, IoT World Forum (IoTWF) and Alternative IoT models, IoT Communication Models, IoT in Global Context, Real world scenarios, Different Areas, Examples Trends in the Adaption of the IoT (Cloud Computing, Big Data Analytics, Concepts of Web of Things, Concept of Cloud of Things with emphasis on Mobile Cloud Computing, Smart Objects).

**UNIT II (10 Hrs)**

**Components in IoT:** Control Units, Sensors, Communication modules, Power Sources, Communication Technologies, RFID, Bluetooth, Zigbee, Wi-fi, RF links, Mobile Internet, Wired Communication;

**IoT Protocol and Technology:** RFID, NFC, Wireless Networks, WSN, RTLS, GPS, Agents, Multi-Agent Systems, IoT Protocols: M2M, BacNet, ModBus, Bluetooth, WiFi, ZigBee;

**Web of Things (WoT):** WoT vs. IoT, Architecture;

**Cloud of Things (CoT):** Grid/SOA and Cloud Computing, Standards, Cloud Providers and Systems, Architecture.

**UNIT III (10 Hrs)**

**Data Analytics for IoT:** Introduction, Machine Learning, Big Data Analytics Tools and Technology, Apache Hadoop, Using Hadoop MapReduce for Batch Data Analysis, Apache Oozie, Apache Spark, Apache Storm, Apache Kafka, Edge Streaming Analytics and Network Analytics, Xively Cloud for IoT, Using Apache Storm for Real-time Data Analysis, Structural Health Monitoring Case Study.

**Tools for IoT:** Chef, Chef Case Studies, Puppet, Puppet Case Study – Multi-tier Deployment, NETCONF-YANG Case Studies, IoT Code Generator.

**UNIT IV (10 Hrs)**

**Domain-specific applications of IoT:** Home automation, Industry applications, Surveillance applications, Smart Homes, Ambient Assisted Living, Intelligent Transport, Other IoT applications: Use-Case Examples.

**Developing IoT solutions:** Introduction to Python, Introduction to different IoT tools, Introduction to Arduino and Raspberry Pi, Implementation of IoT with Arduino and Raspberry Pi, Cloud Computing, Fog Computing, Connected Vehicles, Data Aggregation for the IoT in Smart Cities, Privacy and Security Issues in IoT.

**Text Books:**

1. A. Bahga, V. Madiseti, "Internet of Things: A Hands-on Approach", 1st Edition, Universities Press, 2015/ Latest Edition.
2. R. Kamal, "Internet of Things: Architecture and Design Principles", 1st Edition, McGraw-Hill Education Private Limited, 2017/ Latest Edition.

**Suggested Readings:**

1. D. Uckelmann, M. Harrison, "Architecting the Internet of Things", 1st Ed., Springer, 2011/ Latest Edition.
2. O. Hersent, D. Boswarthick, O. Elloumi, "The Internet of Things – Key applications and Protocols", 2nd Edition, Wiley, 2012/ Latest Edition

**ARYAVART INTERNATIONAL UNIVERSITY**  
**Tilthai, Dharmanagar, North Tripura**

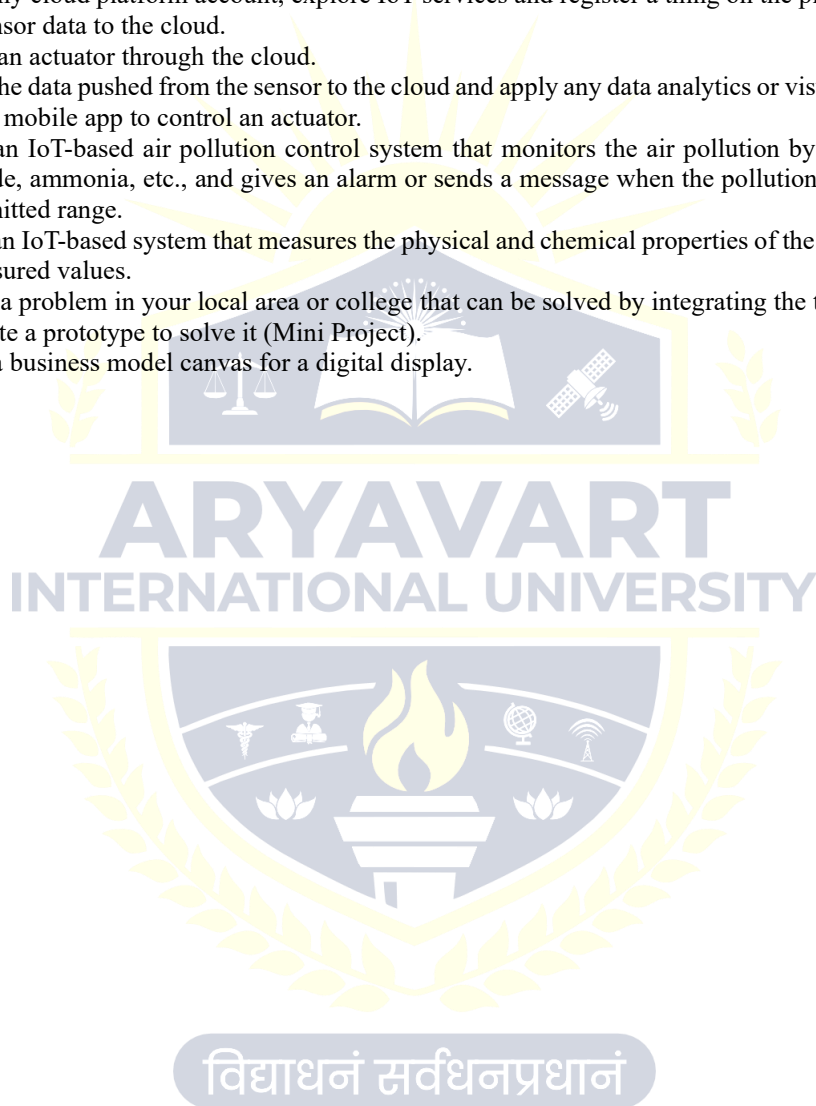
**INTERNET OF THINGS LAB**

**Code: 26CS793**

**Max Marks: 70**

**List of Practicals:**

1. Select any one development board (E.g., Arduino or Raspberry Pi) and control the LED using the board.
2. Using the same board as in (1), read data from a sensor. Experiment with both analogue and digital sensors.
3. Control any two actuators connected to the development board using Bluetooth.
4. Read data from the sensor and send it to a requesting client. (using socket communication) Note: The client and server should be connected to the same local area network.
5. Create any cloud platform account, explore IoT services and register a thing on the platform.
6. Push sensor data to the cloud.
7. Control an actuator through the cloud.
8. Access the data pushed from the sensor to the cloud and apply any data analytics or visualisation services.
9. Create a mobile app to control an actuator.
10. Design an IoT-based air pollution control system that monitors the air pollution by measuring carbon monoxide, ammonia, etc., and gives an alarm or sends a message when the pollution level is more than the permitted range.
11. Design an IoT-based system that measures the physical and chemical properties of the water and displays the measured values.
12. Identify a problem in your local area or college that can be solved by integrating the things you learned, and create a prototype to solve it (Mini Project).
13. Design a business model canvas for a digital display.



**GRAPHICS & ANIMATION**

**Code: 26CS331**

Max Marks: 70

**Course Objectives:** This course aims at teaching students about algorithms involved in 2D and 3D computer graphics and animation. It gives a clear foundation of the graphic operations performed on 2D objects. It enables the students to create 3D realistic objects and generate 2D, 3D animations.

**UNIT I**

**(14 Hrs)**

- Computer graphics fundamentals –overview of CG pipeline- 2D shapes, clipping, 2D Geometric transformations. Simple Animation with 2D transformations.
- Three-dimensional viewing: viewing pipeline, projections: parallel projections, perspective projections.
- 3D transformations

**UNIT II**

**(14 Hrs)**

- Illumination models and surface rendering – polygon rendering methods: constant intensity shading, Gouraud shading, Phong shading. – colour models, Visible surface detection.
- Creating 3D Graphics scene using open-source tools such as blender: Creating Three-dimensional (3D) objects, creating 3D graphics scene with objects, Textures, cameras and lighting, Rendering the scene created.

**UNIT III**

**(12 Hrs)**

- Creating 3D animation using open-source tools such as Blender: animation – basic keyframing, auto keyframing, creating bones and skeletons and animating their movement, inverse kinematics.
- Case study in video sequence editing.

**Suggested Readings:**

1. Hearn D and Baker P. “Computer Graphics Open GL Version”, Fourth Edition, Prentice Hall of India, 2013.
2. Lance Flavell, “Beginning Blender: Open-Source 3D Modelling, Animation, and Game Design”, Apress, 2010
3. James Chronister, “Blender basics classroom tutorial book”, Fourth edition, 2011
4. Plastock R A and Kalley G. “Theory and Problems of Computer Graphics”, Schaum’s Outline Series, TMH; 1985.
5. Foley J D, Eiener S K and Hughes J.F. “Computer Graphics Principles and Practice”, Second Edition, Pearson Education; 1996.

विद्याधनं सर्वधनप्रधानं

**ARYAVART INTERNATIONAL UNIVERSITY**  
**Tilthai, Dharmanagar, North Tripura**

**GRAPHICS & ANIMATION LAB**

**Code: 26CS393**

Max. Marks: 70

**List of Experiments:**

1. Write a program to draw a line using the DDA algorithm.
2. Write a program to draw a line using Bresenham's algorithm.
3. Write a program to draw a polygon (Triangle, square, pentagon, etc.).
4. Write a program to draw a circle/Ellipse using the Mid-Point Circle algorithm.
5. Write a program to implement Area filling using the scan line method.
6. Write a program to implement the Boundary fill 4-connected / 8-connected Algorithm using Recursion/Non-Recursion.
7. Write a program to implement the flood fill Algorithm using Recursion/ Non-Recursion. OR Write a program to fill a solid-coloured area. / Write a program to fill a multicoloured boundary area.
8. Write a program to translate a Line/Polygon.
9. Write a program to rotate and scale a Line/Polygon With respect to
  - i) Origin
  - ii) Pivot Rotation
10. Write a program to perform the shearing of a polygon with respect to
  - i) X-axes
  - ii) Y-axes
11. Write a program to perform a reflection of a polygon with respect to
  - i) X-axes
  - ii) Y-axes
  - iii) With respect to origin
  - iv) With respect to line  $Y=X$
  - v) With respect to line  $Y=mX+C$
12. Draw a Bezier curve using Local Control Point (60,20),(80,100),(150,90),(180,50).
13. Write a program to perform Clipping of Line using Cohen-Sutherland Algorithm.
14. Prepare a game using graphics, basic objects, and various transformations.  
OR  
Create a paintbrush-like application that includes the facility to draw all the basic objects.  
OR  
Develop any useful tool (like a watch) using basic graphics objects and various transformations.
15. Implementation of text compression using dynamic Huffman coding/ static Huffman coding

**HUMAN VALUES AND ETHICS**

**Code: 26GN302**

Max Marks: 70

**Course Objectives:** The course aims to develop ethical reasoning and awareness of human values, promoting responsible behaviour in personal, professional, and societal contexts.

**UNIT I: Introduction to human values**

**(05 Hrs)**

- Understanding the need, Basic guidelines, Process of Value Education.
- Understanding the thought-provoking issues- Continuous happiness and Prosperity.
- Right understanding- relationship and physical facilities, Choice making- choosing, Cherishing and Acting.
- Understanding values- Personal Values, Social values, Moral values and Spiritual values, Self-Exploration and Awareness leading to Self-Satisfaction; Tools for Self-Exploration.

**UNIT II: Harmony and role of values in family, society and human relations**

**(05 Hrs)**

**ARYAVART INTERNATIONAL UNIVERSITY**  
**Tilthai, Dharmanagar, North Tripura**

- Understanding harmony in the Family- the basic unit of human interaction; Understanding values in human-human relationships; Understanding harmony in society-human relations.
- Interconnectedness and mutual fulfilment; Coexistence in nature.
- Holistic perception of harmony at all levels of existence, universal harmonious order in society.
- Visualising a universal harmonium order in society- undivided society (Akhand Samaj), universal order (Sarvabhaum Vyawastha)- from family to world family.

**UNIT III: Coexistence and role of Indian Ethos**

**(05 Hrs)**

- Interconnectedness and mutual fulfilment among the four orders of nature: recyclability and self-regulation in nature.
- Ethos of Vedanta; Application of Indian Ethos in organisations in management; Relevance of Ethics and Values in organisations in current times.

**UNIT IV: Professional ethics**

**(05 Hrs)**

- Understanding about Professional Integrity, respect and equality, Privacy, Building Trusting relationships, Co-operation, and Respecting the competence of other professions.
- Understanding about taking initiative, Promoting the culture of openness, and Demonstrating loyalty towards goals and objectives.
- Ethics at the workplace: - cybercrime, plagiarism, sexual misconduct, fraudulent use of institutional resources, etc.
- Ability to utilise professional competence for augmenting the universal human order.

**Text Books:**

1. A Textbook on Professional Ethics and Human Values by R S Naagarazan.
2. A Foundation Course in Human Values and Professional Ethics by R.R. Gaur, R. Sangal, G.P. Bagaria.
3. Indian Ethos and Modern Management by B L Bajpai, New Royal Book Co., Lucknow, 2004, Reprinted 2008.

**Suggested Readings:**

1. A N Tripathy, 2003, Human Values, New Age International Publishers
2. Human Values and Professional Ethics by Vaishali R Khosla, Kavita Bhagat
3. I.C. Sharma. Ethical Philosophy of India, Nagin & Co., Julundhar

**Code: 26CS791**

Max Marks: 70

**List of Experiments:**

1. Working with Sniffers for monitoring network communication using
  - a) Ethereal
  - b) Wire shark
  - c) Snort
  - d) tcp dump.
2. Implementation and Performance evaluation of various cryptographic algorithms in C/C++
  - a) DES
  - b) RSA.
3. Using IP TABLES on Linux and setting the filtering rules.
4. Using open SSL for web server - browser communication.
5. Configuring S/MIME for e-mail communication.
6. Understanding the buffer overflow and format string attacks.
7. Using NMAP for ports monitoring.
8. Secure Socket programming.

**DESIGN AND ANALYSIS OF ALGORITHMS LAB**

**Code: 26CS692**

Max Marks: 70

**List of Practicals**

1. Write a program to sort the elements of an array using Insertion Sort (The program should report the number of comparisons).
2. Write a program to sort the elements of an array using Merge Sort (The program should report the number of comparisons).
3. Write a program to sort the elements of an array using Heap Sort (The program should report the number of comparisons).
4. Write a program to multiply two matrices using Strassen's algorithm for matrix multiplication
5. Write a program to sort the elements of an array using Radix Sort.
6. Write a program to sort the elements of an array using Bucket Sort.
7. Display the data stored in a given graph using the Breadth-First Search algorithm.
8. Display the data stored in a given graph using the Depth-First Search algorithm.
9. Write a program to determine a minimum spanning tree of a graph using Prim's algorithm.
10. Write a program to implement Dijkstra's algorithm to find the shortest paths from a given source node to all other nodes in a graph.
11. Write a program to solve the weighted interval scheduling problem.
12. Write a program to solve the 0-1 knapsack problem.

For the algorithms at S.No 1, 2 and 3, test run the algorithm on 100 different input sizes varying from 30 to 1000. For each size find the number of comparisons averaged on 10 different input instances; plot a graph for the average number of comparisons against each input size. Compare it with a graph of  $n \log n$ .



# **MCA**

# **4th Semester**

## Detailed Syllabus

### CLOUD COMPUTING

**Code: 26CS702**

Max Marks: 70

**Course Objectives:** The primary objectives of cloud computing include increasing efficiency, scalability, and flexibility of IT resources, while reducing costs and complexity. By leveraging cloud services, organizations can access computing resources on-demand, easily scale up or down based on their needs, and benefit from pay-as-you-go pricing models. This enables businesses to focus on their core competencies, innovate more rapidly, and respond quickly to changing market demands without significant upfront investments in hardware and infrastructure.

#### **UNIT I (10 Hrs)**

Cloud Computing Overview – Services of Internet, Origins of Cloud computing – Cloud components – Essential characteristics – On-demand self-service, The vision of cloud computing – Characteristics, benefits, and Challenges ahead

#### **UNIT II (10 Hrs)**

Cloud Computing Architecture-Introduction – Internet as a Platform, The cloud reference model - Types of clouds - Economics of the cloud, Computing platforms and technologies, Cloud computing economics, Cloud infrastructure - Economics of private clouds - Software productivity in the cloud - Economics of scale: public vs. private clouds.

#### **UNIT III (10 Hrs)**

Principles of Parallel and Distributed Computing: Parallel vs. distributed computing - Elements of parallel computing - Hardware architectures for parallel processing, Approaches to parallel programming - Laws of caution.

#### **UNIT IV (10 Hrs)**

Virtualization: Introduction - Characteristics of virtualized environments - Taxonomy of virtualization techniques - Virtualization and cloud computing - Pros and cons of virtualization - Technology example: VMware: full virtualization, Types of hardware virtualization: Full virtualization - partial virtualization - paravirtualization

#### **Text Books:**

1. Rajkumar Buyya, Christian Vecchiola and S. Thamarai Selvi, “Mastering Cloud Computing” - Foundations and Applications Programming, MK publications, 2013.
2. Gautam Shroff, “Enterprise Cloud Computing: Technology, Architecture, Applications” by Cambridge University Press, 2010.

#### **Suggested Readings:**

1. Michael J. Kavis, “Architecting the Cloud: Design Decisions for Cloud Computing Service Models (SaaS, PaaS, and IaaS)”, John Wiley & Sons Inc., Jan 2014.

## **IMAGE PROCESSING**

**Code: 26CS631**

Max Marks: 70

**Course Objective:** This course introduces students to the fundamentals of image processing, and various image transforms, image restoration techniques, image compression and segmentation used in image processing

### **UNIT I: Introduction to Digital Image Processing**

**(10 Hrs)**

The origins of Digital Image Processing, Examples of Digital Image Processing application, Fundamental steps in Digital Image processing, Components of Image Processing system Fundamentals: Elements of Visual Perception, Light and Electromagnetic Spectrum, Image Sensing and Acquisition, Image Sampling and Quantization, Some basic Relationships between Pixels, Linear and Nonlinear Operations

### **UNIT II: Image Enhancement in the spatial domain**

**(08 Hrs)**

Background, some basic gray level transformation, Introduction of Histogram processing, Enhancement using Arithmetic/Logic operations, Basics of spatial filtering, smoothing spatial filters, Sharpening spatial filters, Image Enhancement in the Frequency Domain: Introduction.

### **UNIT III: Image Restoration**

**(12 Hrs)**

Model of the Image Degradation/Restoration process, Noise Models, Restoration in the presence of noise only spatial filtering, Inverse filtering, Minimum Mean Square Error (Wiener) filtering, Geometric mean filter, Geometric Transformations, Image Compression: Fundamentals, Lossy Compression, Lossless Compression, Image Compression models, Error-free Compression: Variable length coding, LZW coding, Bit plane coding, Run length coding, Introduction to JPEG.

### **UNIT IV: Morphology**

**(04 Hrs)**

Dilation, Erosion, Opening and Closing, Hit-and Miss transforms, Morphological Algorithms: Boundary Extraction, Region filling, Extraction of connected components, Convex Hull

### **UNIT V: Image Segmentation**

**(06 Hrs)**

Definition, characteristics of segmentation Detection of Discontinuities, Edge Linking and Boundary Detection, Thresholding, Region based segmentation. Introduction to Representation and Description, Introduction to Object Recognition.

### **Suggested Readings:**

1. Digital Image Processing: Rafael C. Gonzalez and Richard E. Woods. Addison Wesley.
2. Fundamentals of Digital Image Processing. Anil K. Jain, PHI.

विद्याधनं सर्वधनप्रधानं

**DRUG ABUSE, ROAD SAFETY & TRAFFIC RULES**

**Code: 24GN402**

Max Marks: 70

**Course Objectives:** This course aims to create awareness about the harmful effects of drug abuse, the importance of road safety, and the need to follow traffic rules. It seeks to help learners understand the physical, mental, social, and legal consequences of substance abuse, while also developing responsible attitudes and decision-making skills to avoid peer pressure and risky behaviour. The course further intends to build knowledge of traffic signals, road signs, and safe road use practices, so that learners can behave responsibly as pedestrians, cyclists, passengers, and drivers.

**UNIT I:**

**(10 Hrs)**

Problem of Drug Abuse

- a) Concept and overview
- b) Types of drugs often abused and their effects (Stimulants, Depressants, Narcotics, Hallucinogens, and Miscellaneous).

Causes of consequences of drug abuse

- a) Theories of drug abuse: Physiological theory. Psychological theory. Sociological theory.
- b) Consequences of drug abuse: For individuals, families, society and the economy.

Extent and nature of the problem

- a) Magnitude of the menace of drug abuse.
- b) Vulnerable age groups.
- c) Characteristic and features of proneness.
- d) Signs and symptoms of drug abuse (Physical indicators, Academic indicators, Behavioural and psychological indicators).

Prevention and management of drug abuse

- a) Legislation, public policies and programs for the prevention and cure of drug abuse.
- b) Prevention and Management of drug abuse. Medical management.
- c) Working of drug De-addiction Centres. Role of Family, School and Media.

**UNIT II:**

**(10 Hrs)**

- a) Concept and Significance of Road Safety.
- b) Role of Traffic Police in Road Safety.
- c) Traffic Engineering – Concept & Significance.
- d) Traffic Rules & Traffic Signs.
- e) How to obtain a driving license.
- f) Traffic Offences, Penalties and Procedures.
- g) Common Driving mistakes.
- h) Significance of First-aid in Road Safety.
- i) Role of Civil Society in Road Safety.
- j) Traffic Police-Public Relationship.

**Suggested Readings:**

1. Clayton, J.M., and Scott, M.A (2014). Drugs and Drug Policy: the control of consciousness alteration. New Delhi: Sage Publications India Pvt. Ltd.
2. Kapoor, T. (1985). Drug epidemic among Indian Youth, New Delhi: Mittal Pub
3. Modi, I and Modi, S. (1997). Drugs: Addiction and prevention, Jaipur: Rawat Publication.
4. 2003 National Household Survey of Alcohol and Drug Abuse. New Delhi, Clinical Epidemiological Unit, AIIMS, 2004
5. World Drug Report (updated every year), United Nations office of Drug and Crime.
6. Extent, pattern and Trend of Drug use in India, Ministry of Social Justice and Empowerment, Government of India, 2004.
7. The Narcotic Drugs and Psychotropic Substances Act, 1985. (New Delhi: Universal, 2012).
8. Government of India (2015). Scheme of assistance for prevention and alcoholism and substance (Drugs) abuse and for social defence services-Guidelines. Ministry of social Justice and Empowerment. New Delhi.
9. NCERT (2010). Training Resource Materials (Adolescence Education Programme).

**ARYAVART INTERNATIONAL UNIVERSITY**  
**Tilthai, Dharmanagar, North Tripura**

10. The Motor Vehicle Act, 1988 (2010), Universal Law Publishing Co. Pvt. Ltd., New Delhi.
11. Road Safety Signage and Signs (2011), Ministry of Road Transport and Highways, Government of India.

**CLOUD COMPUTING LAB**

**Code: 26CS792**

Max Marks: 70

**List of Practicals**

1. Install Virtualbox/VMware Workstation with different flavours of Linux or windows OS on top of windows7 or 8.
2. Install a C compiler in the virtual machine created using virtual box and execute Simple Programs.
3. Install Google App Engine. Create hello world app and other simple web applications using python/java.
4. Use GAE launcher to launch the web applications.
5. Simulate a cloud scenario using CloudSim and run a scheduling algorithm that is not present in CloudSim.
6. Find a procedure to transfer the files from one virtual machine to another virtual machine.
7. Find a procedure to launch virtual machine using trystack (Online Openstack Demo Version).
8. Install Hadoop single node cluster and run simple applications like wordcount.

**DIGITAL IMAGE PROCESSING LAB**

**Code: 26CS693**

Max Marks: 70

**List of Practicals:**

1. Display of Grey-scale Images.
2. Histogram Equalisation.
3. Design of Non-linear Filtering.
4. Determination of Edge Detection Using Operators.
5. 2-D DFT and DCT.
6. Filtering in the frequency domain.
7. Display of colour images.
8. Conversion between colour spaces.
9. DWT of images.
10. Segmentation using watershed transform.

विद्याधनं सर्वधनप्रधानं

**ARYAVART INTERNATIONAL UNIVERSITY**  
**Tilthai, Dharmanagar, North Tripura**

**SEMINAR/CONFERENCE**

**Code: 26PR102**

Max Marks: 70

**OBJECTIVE:**

Seminars/Conferences and Presentations provide a platform to the students, where they can learn from what others are doing, learn about new things, ideas and important tips related to new technologies. To foster the Innovations happening in upcoming technologies and harnessing the entrepreneurial opportunities, Institutes must provide ample opportunities to the students to learn and yield the advantages of new advancements in the field of computer applications. It is expected from a student to learn latest in the industry and write an article related to it and present their findings in front of a panel.

The following points need to be considered while planning and evaluating the presentation:

- The seminars must be conducted after every 15 days/ or a month. A minimum of 3-4 seminar sessions can be organized during the semester.
- A minimum of 7-8 slides must be there which would include the title slide. The first slide should be the Introduction slide and the last one reference slide wherein all the links/books references/paper reference to paper must be quoted. The rest of the slides should focus on the technology, application areas etc.
- The title of the seminar must be related to the field of Computer Applications and must talk about the latest innovation/technology like IOT, Machine learning, Deep learning, AI Cloud computing, Mobility, Hand held devices, Social Computing, NOSQL Database, CRM, Social CRM, Open Source Application Development Frameworks, Zero Trust Security Framework/ Architecture, Big Data/ Data Lake, Emerging and Innovative Technologies, Conversational AI, Sentiments Analysis, DevOps, Real time Analytics, Fraud Detection. Proper approval must be taken before starting the work.
- Student's feedback must be taken after taking the seminar as to what learning they have gathered after studying the topics. For this, a feedback form may be designed using Google form utility.

**ARYAVART**  
**INTERNATIONAL UNIVERSITY**



विद्याधनं सर्वधनप्रधानं

## **MAJOR PROJECT**

**Code: 26PR104**

**Max Marks: 70**

### **PROJECT REPORT**

**All the students are required to submit a report based on the project work done by them during the fourth semester.**

#### **SYNOPSIS (SUMMARY/ABSTRACT):**

All students must submit a summary/abstract separately with the project report. Summary, preferably, should be about 3-4 pages. The content should be as brief as is sufficient to explain the objective and implementation of the project that the candidate is going to take up. The write-up must adhere to the guidelines and should include the following:

- Name/Title of the Project
- Statement about the Problem
- Why are the particular topic chosen?
- Objective and scope of the Project
- Methodology (including a summary of the project)
- Hardware & Software to be used
- Testing Technologies used
- What contribution would the project make?

**TOPIC OF THE PROJECT-** This should be explicitly mentioned at the beginning of the Synopsis. Since the topic itself gives a peek into the project to be taken up, the candidate is advised to be prudent in naming the project. This is the overall impression of the future work; the topic should corroborate the work.

**OBJECTIVE AND SCOPE:** This should give a clear picture of the project. The objective should be clearly specified. What the project ends up to and in what way this is going to help the end user has to be mentioned.

**PROCESS DESCRIPTION:** The process of the whole software system proposed to be developed should be mentioned in brief. This may be supported by DFDs / Flowcharts to explain the flow of the information.

**RESOURCES AND LIMITATIONS:** The requirements of resources for designing and developing the proposed system must be given. The resources might be in the form of hardware/software or data from the industry. The limitation of the proposed system with respect to a larger and comprehensive system must be given.

**CONCLUSION:** The write-up must end with the concluding remarks- briefly describing innovation in the approach for implementing the Project, main achievements and also any other important feature that makes the system stand out from the rest.

**The following suggested guidelines must be followed in preparing the Final Project Report:** Good quality white A4 size paper should be used for typing and duplication. Care should be taken to avoid smudging while duplicating the copies.

Page Specification: (Written paper and source code)

- Left margin- 3.0 cms
- Right margin- 2.0 cms
- Top margin- 2.54 cms
- Bottom margin- 2.54 cms
- Page numbers- All text pages as well as Program source code listing should be numbered at

**ARYAVART INTERNATIONAL UNIVERSITY**  
**Tilthai, Dharmanagar, North Tripura**

the bottom centre of the pages.

**Normal Body Text: Font Size:** 12, Times New Roman, Double Spacing, justified 6 points above and below para spacing.

**Paragraph Heading Font Size:** 14, Times New Roman, Underlined, Left Aligned. 12 points above & below spacing.

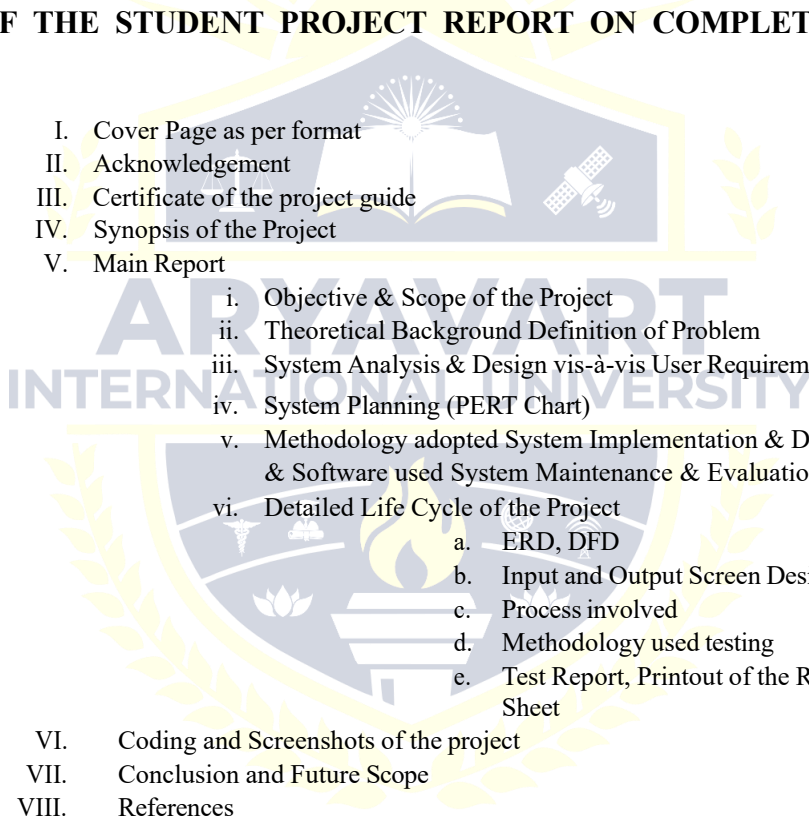
**Chapter Heading Font Size:** 20, Times New Roman, Centre Aligned, 30 points above and below spacing.

**Coding Font size :**10, Courier New, Normal

**Submission of Project Report to the University:** The student will submit his/her project report in the prescribed format. The Project Report should include:

1. One copy of the summary/abstract.
2. One hard Copy of the Project Report.
3. The Project Report may be about 75 pages (excluding coding).

**FORMAT OF THE STUDENT PROJECT REPORT ON COMPLETION OF THE PROJECT**

- 
- I. Cover Page as per format
  - II. Acknowledgement
  - III. Certificate of the project guide
  - IV. Synopsis of the Project
  - V. Main Report
    - i. Objective & Scope of the Project
    - ii. Theoretical Background Definition of Problem
    - iii. System Analysis & Design vis-à-vis User Requirements
    - iv. System Planning (PERT Chart)
    - v. Methodology adopted System Implementation & Details of Hardware & Software used System Maintenance & Evaluation
    - vi. Detailed Life Cycle of the Project
      - a. ERD, DFD
      - b. Input and Output Screen Design
      - c. Process involved
      - d. Methodology used testing
      - e. Test Report, Printout of the Report & Code Sheet
  - VI. Coding and Screenshots of the project
  - VII. Conclusion and Future Scope
  - VIII. References

**Formats of various certificates and formatting styles are as:**

**1. Certificate from the Guide**

**CERTIFICATE**

This is to certify that this project entitled “.....” submitted in partial fulfillment of the degree of Masters of Computer Applications to the “.....” through ..... done by Mr./Ms. .... Enrollment No. .... is an authentic work carried out by him/her at .....my guidance. The matter embodied in this project work has not been submitted earlier for award of any degree to the best of my knowledge and belief.

Signature of the student

Signature of the Guide

**ARYAVART INTERNATIONAL UNIVERSITY**  
**Tilthai, Dharmanagar, North Tripura**

**2. Project Report Cover Page Format:**

**Title of the Project/report**

(Times New Roman, Italic, Font size= 24)

**Submitted in partial fulfilment of the requirements for the  
award of the degree of Masters of Computer Applications -  
(Bookman Old Style, 16points, centre)**

**Submitted to:**  
**(Guide Name)**

**Submitted by:**  
**(Student's name)**  
**Enrolment No.:**

**3. Self-Certificate by the students**

**SELF CERTIFICATE**

This is to certify that the dissertation/project report entitled “\_\_\_\_\_” is done by me is an authentic work carried out for the partial fulfilment of the requirements for the award of the degree of Masters of Computer Applications under the guidance of \_\_\_\_\_. The matter embodied in this project work has not been submitted earlier for award of any degree or diploma to the best of my knowledge and belief.

Signature of the student

Name of the Student

Enrollment No.

**4. ACKNOWLEDGEMENTS**

In the “Acknowledgements” page, the writer recognizes his indebtedness for guidance and assistance of the thesis adviser and other members of the faculty. Courtesy demands that he also recognizes specific contributions by other persons or institutions such as libraries and research foundations. Acknowledgements should be expressed simply, tastefully, and tactfully.

विद्याधनं सर्वधनप्रधानं