

MADANAPALLE INSTITUTE OF TECHNOLOGY & SCIENCE

Madanapalle

(UGC-AUTONOMOUS)

www.mits.ac.in



DEPARTMENT OF COMPUTER APPLICATIONS

CURRICULUM STRUCTURE

&

DETAILED SYLLABUS

For the students admitted to

Master of Computer Applications from the academic year 2018-19

AND

Master of Computer Applications Lateral Entry Scheme

from the Academic Year 2019-20



Curriculum – MCA

Breakup of Courses

Sl.No	Category	No. of Theory Courses	No. of Practical Courses	Project Works	Seminars	Curriculum Credits	Weightage (%)
1	Programme Core Courses	18	13	2	1	108.5	77.5
2	Discipline Electives	4	5	-	-	21.5	15.4
3	Open Electives	3	-	-	-	10	7.1
4	Mandatory Courses	2	-	-	-	-	-
	Total	27	18	2	1	140	100



CURRICULUM STRUCTURE

Master of Computer Applications

MCA I Year I Semester

S.No	Course Code	Course Title	L	T	P	C
1	18ENGP104	English for Communication	4	0	0	4
2	18MATP101	Discrete Mathematics	4	0	0	4
3	18HUMP101	Accounting and Financial Management	4	0	0	4
4	18MCAP101	Computer Organization and Architecture	4	0	0	4
5	18MCAP102	Problem Solving with Python	4	0	0	4
6	18ENGP201	English for Communication Laboratory	0	0	3	1.5
7	18MCAP201	PC Hardware & Networking Laboratory	0	0	3	1.5
8	18MCAP202	Problem Solving with Python Laboratory	0	0	3	1.5
9	18MCAP203	Office Automation Laboratory	0	0	3	1.5
		Total	20	0	12	26

MCA I Year II Semester

S.No	Course Code	Course Title	L	T	P	C
1	18MATP114	Probability and Statistics	4	0	0	4
2	18MCAP103	Programming in C	4	0	0	4
3	18MCAP104	Database Management Systems	4	0	0	4
4	18MCAP105	Operating Systems	4	0	0	4
5	18MCAP106	Software Engineering	4	0	0	4
6	18MCAP204	Programming in C Laboratory	0	0	3	1.5
7	18MCAP205	Database Management Systems Laboratory	0	0	3	1.5
8	18MCAP206	Unix & Shell Programming Laboratory	0	0	3	1.5
9		Software Lab – I	0	0	3	1.5
		Total	20	0	12	26



CURRICULUM STRUCTURE

Master of Computer Applications

MCA II Year I Semester

S.No	Course Code	Course Title	L	T	P	C
1	18MCAP107	Object Oriented Programming	4	0	0	4
2	18MCAP108	Full Stack Web Development	4	0	0	4
3	18MCAP109	Introduction to Machine Learning	4	0	0	4
4	18MCAP110	Computer Networks	4	0	0	4
5		Open Elective – I	3	0	0	3
6	18MCAP207	Object Oriented Programming Laboratory	0	0	3	1.5
7	18MCAP208	Full Stack Web Development Laboratory	0	0	3	1.5
8	18MCAP209	Machine Learning Laboratory	0	0	3	1.5
9		Software Lab – II	0	0	3	1.5
10	18ENG905	Corporate Communication Skills (Mandatory Course)	2	0	0	0
		Total	21	0	12	25

MCA II Year II Semester

S.No	Course Code	Course Title	L	T	P	C
1	18MCAP111	Data Warehousing and Data Mining	4	0	0	4
2	18MCAP112	Data Structures and Algorithms	4	0	0	4
3	18MCAP113	Cloud Computing	4	0	0	4
4		Discipline Elective – I (18MCAP401-18MCAP405)	3	0	0	3
5		Discipline Elective – II (18MCAP406-18MCAP408)	4	0	0	4
6	18MCAP210	Data Warehousing and Data Mining Laboratory	0	0	3	1.5
7	18MCAP211	Data Structures and Algorithms Laboratory	0	0	3	1.5
8		Discipline Elective – I Laboratory (18MCAP501-18MCAP505)	0	0	3	1.5
9		Discipline Elective – II Laboratory (18MCAP506-18MCAP508)	0	0	3	1.5
10	18ENG906	Personality Development and Soft Skills (Mandatory Course)	2	0	0	0
11	18MCAP601	Technical Seminar	0	2	0	1
		Total	21	2	12	26

MADANAPALLE INSTITUTE OF TECHNOLOGY & SCIENCE: MADANAPALLE

(UGC – Autonomous)

Department of Computer Applications


CURRICULUM STRUCTURE
Master of Computer Applications
MCA III Year I Semester

S.No	Course Code	Course Title	L	T	P	C
1	18MCAP114	Agile Software Development Process and DevOps	4	0	0	4
2		Discipline Elective-III (18MCAP409-18MCAP411)	4	0	0	4
3		Discipline Elective-IV (18MCAP412-18MCAP415)	3	0	0	3
4		Open Elective-II	4	0	0	4
5		Open Elective –III	3	0	0	3
6	18MCAP212	Agile Software Development Process and DevOps Laboratory	0	0	3	1.5
7		Discipline Elective – IV Laboratory (18MCAP509 - 18MCAP512)	0	0	3	1.5
8	18MCAP701	Internship / Mini Project	0	0	4	2
		Total	18	0	10	23

MCA III Year II Semester

S.No	Course Code	Course Title	L	T	P	C
1	18MCAP702	Major Project	0	0	28	14
		Total				14



MADANAPALLE INSTITUTE OF TECHNOLOGY & SCIENCE: MADANAPALLE

(UGC – Autonomous)

Department of Computer Applications

CURRICULUM STRUCTURE

Master of Computer Applications

List of Discipline Core Courses (All Courses Carry Equal Marks (100))

S.No.	Course Code	Course Name	Credits
Theory Course			
1.	18ENGP104	English for Communication	4
2.	18MATP101	Discrete Mathematics	4
3.	18HUMP101	Accounting and Financial Management	4
4.	18MCAP101	Computer Organization & Architecture	4
5.	18MCAP102	Problem Solving with Python	4
6.	18MATP114	Probability & Statistics	4
7.	18MCAP103	Programming in C	4
8.	18MCAP104	Database Management Systems	4
9.	18MCAP105	Operating Systems	4
10.	18MCAP106	Software Engineering	4
11.	18MCAP107	Object Oriented Programming	4
12.	18MCAP108	Full Stack Web Development	4
13.	18MCAP109	Introduction to Machine Learning	4
14.	18MCAP110	Computer Networks	4
15.	18MCAP111	Data Warehousing and Data Mining	4
16.	18MCAP112	Data Structures and Algorithms	4
17.	18MCAP113	Cloud Computing	4
18.	18MCAP114	Agile Software Development Process and DevOps	4
Total			72
Practical Courses			
1.	18ENGP201	English for Communication Laboratory	1.5
2.	18MCAP201	PC Hardware & Networking Laboratory	1.5
3.	18MCAP202	Problem Solving with Python Laboratory	1.5
4.	18MCAP203	Office Automation Laboratory	1.5
5.	18MCAP204	Programming in C Laboratory	1.5
6.	18MCAP205	Database Management Systems Laboratory	1.5
7.	18MCAP206	Unix & Shell Programming Laboratory	1.5
8.	18MCAP207	Object Oriented Programming Laboratory	1.5
9.	18MCAP208	Full Stack Web Development Laboratory	1.5
10.	18MCAP209	Machine Learning Laboratory	1.5
11.	18MCAP210	Data Warehousing and Data Mining Laboratory	1.5
12.	18MCAP211	Data Structures and Algorithms Laboratory	1.5
13.	18MCAP212	Agile Software Development Process and DevOps Laboratory	1.5
Total			19.5
Project & Seminar			

1.	18MCAP601	Technical Seminar	1
2.	18MCAP701	Internship / Mini Project	2
3.	18MCAP702	Major Project	14
Total			17

**List of Mandatory Courses
(No Credits & End Exam – Only Internal Evaluation)**

Sl.No.	Course Code	Course Name	Offered by the Dept. of
1.	18ENG905	Corporate Communication Skills	English
2.	18ENG906	Personality Development and Soft Skills	English



CURRICULUM STRUCTURE
Master of Computer Applications
LIST OF DISCIPLINE ELECTIVES

Discipline Elective - I

S No	Course Code	Course Name	Credits
1	18MCAP401	Mobile Application Development Using Android	03
2	18MCAP402	Web Programming through PHP	03
3	18MCAP403	Web Technologies through Java	03
4	18MCAP404	Software Testing	03
5	18MCAP405	User Interface Design and Multimedia Systems	03

Discipline Elective – I Laboratory

S No	Course Code	Course Name	Credits
1	18MCAP501	Mobile Application Development Using Android Laboratory	1.5
2	18MCAP502	Web Programming through PHP Laboratory	1.5
3	18MCAP503	Web Technologies through Java Laboratory	1.5
4	18MCAP504	Software Testing Laboratory	1.5
5	18MCAP505	User Interface Design and Multimedia Systems Laboratory	1.5

Discipline Elective II and III Cluster

S. No	Course Name	
1	AI and Deep Learning	Deep Learning
		Deep Learning Laboratory
		Artificial Intelligence
2	Data Science	Fundamentals of Data Science
		Fundamentals of Data Science Laboratory
		Big Data Analytics
3	Information and Network Security	Cryptography and Network Security
		Cryptography and Network Security Laboratory
		Cyber Security and Cyber Forensics



CURRICULUM STRUCTURE

Master of Computer Applications

Discipline Elective II

S. No	Course Code	Course Name	Credits
1	18MCAP406	Deep Learning	04
2	18MCAP407	Fundamentals of Data Science	04
3	18MCAP408	Cryptography and Network Security	04

Discipline Elective II Laboratory

S. No	Course Code	Course Name	Credits
1	18MCAP506	Deep Learning Laboratory	1.5
2	18MCAP507	Fundamentals of Data Science Laboratory	1.5
3	18MCAP508	Cryptography and Network Security Laboratory	1.5

Discipline Elective III

S. No	Course Code	Course Name	Credits
1	18MCAP409	Artificial Intelligence	04
2	18MCAP410	Big Data Analytics	04
3	18MCAP411	Cyber Security and Cyber Forensics	04

Discipline Elective – IV

S No	Course Code	Course Name	Credits
1	18MCAP412	Advanced Data Structures and Algorithms	03
2	18MCAP413	.Net Framework and C#	03
3	18MCAP414	Internet of Things	03
4	18MCAP415	Advanced Java Programming	03



CURRICULUM STRUCTURE
Master of Computer Applications
Discipline Elective – IV Laboratory

S No	Course Code	Course Name	Credits
1	18MCAP509	Advanced Data Structures and Algorithms Laboratory	1.5
2	18MCAP510	.Net Framework and C# Laboratory	1.5
3	18MCAP511	Internet of Things Laboratory	1.5
5	18MCAP512	Advanced Java Programming Laboratory	1.5

Software Lab - I

S.NO	Course Code	Course Name	Credits
1	18MCAP513	Data Analysis with SPSS Laboratory	1.5
2	18MCAP514	Data Analysis Using R Laboratory	1.5

Software Lab – II

S.NO	Course Code	Course Name	Credits
1	18MCAP515	C++ Laboratory	1.5
2	18MCAP516	Network Simulator Laboratory	1.5



CURRICULUM STRUCTURE
Master of Computer Applications

LIST OF OPEN ELECTIVES (I, II & III)

Open Elective-I

S.NO	Course Code	Course Name	Credits
1	18MBAP101	Management and Organizational Behavior	03
2	18MBAP102	Managerial Economics	03
3	18MBAP103	Business Environment	03
4	18MBAP111	Human Resource Management	03
5	18MBA302	Introduction to Design Thinking	03

Open Elective – II

S.NO	Course Code	Course Name	Credits
1	18MBAP114	Operations Research	04
2	18MBAP419	Logistics and Supply Chain Management	04
3	18MBAP431	Management of Software Projects	04
4	18MBAP434	E-Business	04
5	18MBAP435	Enterprise Resource Planning	04

Open Elective – III

S.NO	Course Code	Course Name	Credits
1	18MBAP501	Management Information Systems	03
2	18MBAP503	E-Commerce and Digital Markets	03
3	18MBAP504	Managing Digital Innovation and Transformation	03
4	18MBAP509	International Business	03
5	18MBAP513	Entrepreneurship Development and Project Management	03

Programme Core Courses

MCA I Year I Semester

18ENGP104 ENGLISH FOR COMMUNICATION

L	T	P	C
4	0	0	4

Course Prerequisite : None

Course Description:

English for Communication course is designed by keeping the academic and professional needs of the MCA learners. The present course is designed in such way that it covers all the LSRW skills of English language. The first unit of the course mainly focuses on grammar and vocabulary building. This unit provides a learners good exposure of the grammatical and vocabulary aspect of the language. The second unit prepares learners for verbal & non-verbal communication skills. In the third unit learners are exposed to all the LSRW skills. Here learners are taught skill-wise effective strategies of learning English. The fourth unit prepares learners for presentation skills. Here the learners are given practical knowledge in giving effective presentation skills and they are made aware of the dos and don'ts of the effective presentation skills. The last unit technical writing skill enhances learners' technical writing abilities by teaching them business letters and various report writing skills. Overall, the main aim of the course is to develop learners' language proficiency in English.

Course Objectives:

Students will be able to:

1. Enhance students word knowledge and grammatical competency
2. Make students aware of the usage of verbal & Nonverbal communication
3. Enhance learner's ability in LSRW Skills
4. Develop student's skills/ability in giving presentations
5. Improve technical writing competency

UNIT-I

Grammar & Vocabulary Building: Synonyms and Antonyms, Word Roots, One-Word Substitutes, Prefixes and Suffixes, Idioms and Phrases – Grammar- Use of Articles, Prepositions, Tenses & Subject-Verb Agreement – Common Errors

UNIT-II

Verbal & Non-Verbal Communication (Body Language - Kinesics, Proxemics Etc.,) Verbal Communication - Informal Vs Formal Conversation – Features of and Barriers to Effective Communication – Process and Flow of Communication

UNIT-III

Communication Skills (LSRW Skills) Listening (Barriers to & Features of Effective Listening), Speaking (Jargons - Rate of Speech, Pitch, Tone - Clarity of Voice Etc.), Reading (Techniques)

And Writing (Features of Effective Writing Such "As Clarity, Brevity, Appropriate Tone, Balance Etc.) - Differences Between Spoken and Written Communication

UNIT-IV

Presentation Skills: Types of Presentation – Video Conferencing-- Participation in Meetings - Chairing Sessions.

UNIT-V

Technical Report Writing Business Letters - Format - Style – Effectiveness, Promptness- E-Mail, Fax - Types of Reports - Progress Reports, Routine Reports - Annual Reports - Formats - Analysis of Sample Reports from Industry - Synopsis and Thesis Writing

Course Outcomes:

After completion of the above course the student will be able to:

1. Good at Grammar & Vocabulary Usage
2. Learners Will Be Good in Communications in terms of Verbal & Non Verbal communication
3. Learners will improve their proficiency in usage of LSRW skills
4. Improve their presentation skills
5. Learners will enhance their technical writing skills.

Text Books:

1. Technical Communication by Meenakshi Raman and Sangeetasharma, Oxford Univ.Press.
2. Effective Technical Communication, M Ashraf Rizvi, Tata Mc. Graw-Hill Pub, company Ltd.
3. High School English Grammar & Composition by WREN & MARTIN, S.Chand & Company Pvt. Ltd.

References:

1. Communication Skills, Sanjay Kumar & PushpLata, Oxford Univ.Press.
2. Basic Communication Skills for Technology, Andrea J. Rutherford: Pearson Education Asia, New Delhi.
3. GRE and TOEFL; Kaplan and Baron's English in Mind, Herbert Puchta and Jeff Stranks, Cambridge
4. Communication Skills, LenneSen, Prentice –Hall of India Pvt. Ltd., New Delhi.
5. Communicating at work, Ronald B. Adler, Seanne Marquardt Elmhurst, McGraw

Mode of Evaluation: Assignments, Internal Mid Examination, External End Examination

MCA I Year I Semester

18MATP101

DISCRETE MATHEMATICS

L T P C
4 0 0 4

Course Prerequisite : None

Course Description :

This course introduces the applications of discrete mathematics in the field of computer science. It covers algebraic structures, Combinatorics and finite state machines. It also provides insight into the concepts of graph theory and applications.

Course Objectives:

Students will be able to

1. Introduce the concepts of logic, rules of inference and predicates.
2. Concepts of Set Theory, Relations & Functions will be explained.
3. Provides an illustration of problems in graph theory.
4. Explain the concepts of Algebraic Structures and discuss Combinatorics.
5. Apply the principles of Number Theory and Automata Theory.

UNIT I: FOUNDATIONS OF LOGIC

Introduction, truth tables, statements and notations, propositional logic; Connectives, propositional equivalence; predicate and quantifiers; Normal forms; rules of Inference; methods of proofs.

UNIT II: SET THEORY, RELATIONS & FUNCTIONS

Basics of set theory, set operations, Relations and their properties, representing relations, Properties of binary Relations, Equivalence relations, Lattice and its Properties, Partial ordering, Hasse diagram. Composition of functions, Inverse Function, types of functions, Recursive Functions.

UNIT III: GRAPH THEORY

Graphs and graph models, graph terminology and special types of graphs, representing graphs and graph isomorphism, connectivity, Euler and Hamiltonian paths, shortest path problems, planar graphs, graph coloring, Trees: Introduction to trees, Applications of trees, spanning trees & minimum spanning trees.

UNIT IV: ALGEBRAIC STRUCTURES & ELEMENTARY COMBINATORICS

Definition and elementary properties of groups, semigroups, monoids. Elementary Combinatorics: counting techniques, Pigeonhole Principle and its application. Recursion, Recurrence relation.

UNIT V: NUMBER THEORY & AUTOMATA THEORY

Basic Number theory, prime numbers, modular congruence, Integers and algorithms, Applications of number theory- RSA algorithm. Languages and Grammars, Finite state machines with output, Finite state machines with no output, Language recognition, Turing machines.

Course Outcomes:

After completion of the course the student will be able to:

1. Evaluate elementary mathematical arguments and identify fallacious reasoning (not just fallacious conclusions).
2. Use logical notation to define and reason about fundamental mathematical concepts such as sets, relations and functions.
3. Apply graph theory models of data structures and state machines to solve problems of connectivity and constraint satisfaction, for example, scheduling.
4. Synthesize concepts of Algebraic Structures and Combinatorics.
5. Prove elementary properties of modular arithmetic and explain their applications in Computer Science, for example, in cryptography and hashing algorithms.

Text Book:

1. Discrete Mathematics and its applications, Kenneth Rosen, Seventh Edition, TataMcGrawHill Education Private Limited.
2. Discrete Mathematical Structures with Applications to computer science J.P Tremblay, R. Manohar, TMH.

References:

1. “Discrete mathematics for computer scientists and mathematicians”, Mott, Kandel, Baker, PHI
2. Johnson Baugh R, and Carman R, Discrete mathematics, 5th edition, Person Education, 2003.
3. Kolman B, Busoy R.C, and Ross S.C, Discrete Mathematical Structures, 5th edition, Prentice – Hall, 2004.

Mode of Evaluation: Assignments, Internal Mid Examination, External End Examination

18HUMP101 ACCOUNTING AND FINANCIAL MANAGEMENT

L	T	P	C
4	0	0	4

Course Prerequisite: None

Course Description:

This course introduces the accounting system, principles, types of accounts, accounting standards and financial statements etc. Financial management deals with the importance, the role and functions of financial manager in the organization and sources of finance. The ratio analysis and financial analysis are useful to know the position of financial statements. Funds flows statements and cash flow statements are explained to know the analysis of financial matters and explains the need of working capital. Break even analysis concept is necessary to know the profitability position, and useful to know the subject for taking managerial decisions. Capital budgeting explains the need for capital, different investment proposals, and methods to select a proposal which is suitable to the organization.

Course Objectives:

Students will be able to

1. Explain the accounting principles, types of accounting
2. Accounting Books recording procedure and preparation of final accounts.
3. Explain Inventory Valuation Methods.
4. Describe the financial analysis through funds flow and cash flow analysis.
5. Explain the ratio analysis and break-even concepts and problems.

UNIT I: INTRODUCTION TO ACCOUNTING

Definition, Importance, Objectives, Uses of Accounting and Book Keeping Vs Accounting, Single Entry and Double Entry Systems, Classification of Accounts - Rules of Debit & Credit.

UNIT II: THE ACCOUNTING PROCESS

Overview, Books of Original Record; Journal and Subsidiary books, ledger, Trial Balance, Final accounts: Trading accounts - Profit & loss accounts - Balance sheets with adjustments, accounting principles. (Simple Problems)

UNIT III: VALUATION OF ASSETS

Introduction to Depreciation - methods (Simple problems from Straight line method, Diminishing balance method and Annuity method). Inventory Valuation: Methods of inventory valuation

(Simple problems from LIFO, FIFO, and Simple Average & Weighted Average). Valuation of goodwill methods of valuation of goodwill. (Simple Problems)

UNIT IV: FINANCIAL ANALYSIS-I

Objectives of fund flow statement - steps in preparation of fund flow statement, Objectives of cash flow statement- Preparation of cash flow statement - funds flow statement Vs cash flow statement. (Simple Problems)

UNIT V: FINANCIAL ANALYSIS-II

Analysis and interpretation of financial statements from investor and company point of view, Liquidity, leverage, solvency and profitability ratios - Du Pont Chart – BEP Analysis (Simple Problems)

Course Outcomes:

After completion of the above courses the student will be able to:

1. Understand the various type of accounting principles
2. To know how to maintain accounting books and the preparation of final accounts.
3. Assess and apply the inventory valuation methods.
4. Compare and apply the financial analysis through funds flow and cash flow analysis.
5. Understand the concept of ratio analysis and break-even point problems.

Text Books:

1. Financial Accounting, S.N.Maheshwari, Sultan Chand, 2009.
2. Financial Accounting, Tulsian, S Chand, 2009.

Reference Books:

1. Financial Statement Analysis, Khan and Jain, PHI, 2009
2. Financial Management and Policy, Van Horne, James,C., Pearson ,2009

Mode of Evaluation: Assignments, Internal Mid Examination, External End Examination

MCA I Year I Semester

18MCAP101 COMPUTER ORGANIZATION AND ARCHITECTURE

L T P C

4 0 0 4

Course Prerequisite: None

Course Description:

This course introduces internal digital circuits & their operations and basic building block of various computers. Addressing modes and subroutine call are detailed along with memory organization & importance of virtual memory. Importance of I/O interfaces including data transfer.

Course Objectives:

Student will be able to:

1. Understand the concepts and working mechanisms of digital circuits.
2. Analyze the working mechanism of processing unit
3. Understand various instruction sets and exception handling
4. Differentiate various levels in memory hierarchy.
5. Understand the concepts on I/O Organization

UNIT I: OVERVIEW OF BASIC DIGITAL BUILDING BLOCKS

Basic structure of a digital computer, Number system and representation of information, arithmetic and logical operation, hardware implementation, Real numbers - fixed and floating point, IEEE754 representation. Logic gates, Sequential and combinational circuits, flip flop, adders, multiplexers, encoders, decoders, Registers.

UNIT II: BASIC PROCESSING UNIT

Fundamental concepts, Addressing modes, RISC, CISC, Registers, Instruction formats, ALU design Execution of a complete instruction, Multiple bus organization, Hardwired control, Micro programmed control, Nano programming.

UNIT III: PIPELINING

Basic concepts, Data hazards, Instruction hazards, Influence on instruction sets, Data path and control considerations, Performance considerations Exception handling.

UNIT IV: MEMORY SYSTEM

Basic concepts, Semiconductor, Magnetic, Optical memories, Semiconductor RAM, ROM, Speed Size and cost, Cache memories, Improving cache performance, Virtual memory, Memory management requirements, Associative memories, Secondary storage devices.

UNIT V: I/O ORGANIZATION

Accessing I/O devices, Programmed Input/output, Interrupts, Direct Memory Access, Buses Interface circuits, Standard I/O Interfaces (PCI, SCSI, USB), I/O devices and processor

Course Outcomes:

After completion of the courses the student will be able to

1. Explain the use and working procedure of combinational and sequential logical circuits.
2. Explain the internal working concept of processing unit
3. Explore the different instruction sets and exception handling
4. Distinguish the concept of memory mapping techniques within the memory hierarchy.
5. Identify the importance of I/O organization

Text Books:

1. Computer Organization and Architecture by William Stallings 9th Edition,
2. Computer System Architecture by M. Morris Mano, 4th Edition

Reference Books:

1. Computer Organization, Carl Hamacher , Zaky
2. Computer Architecture and Organization, Myles Murudocca, Vincent Heuring , Wiley
3. Computer System Organization and Architecture, M.Usha, T.S. Srikanth

Mode of Evaluation: Assignments, Internal Mid Examination, External End Examination

Course Prerequisite: None

Course Description:

This course introduces problem solving methodologies and discusses how to write algorithms for computational problems. It also discusses the complete overview of python programming from the scratch to the file systems, modules and importing packages in python language.

Course Objectives:

Student will be able to

1. Understand how to write algorithms for the concepts on computational problems
2. Know the basics of python programming such as tokens, data types etc.
3. Get idea on the control statements, how to use functions and the string handling mechanisms in python language
4. Recognize with the concepts of tuples and dictionaries in python language
5. Comprehend file handling concepts, modules and packages of python

UNIT - I COMPUTER PROBLEM SOLVING, FUNDAMENTAL ALGORITHMS

Programs and algorithms, Problem-Solving aspect, top-down design, implementation of algorithms, program verification and efficiency of algorithms and analysis of algorithms. Exchanging the values of two variables, counting, summation of set of numbers, factorial computation, sine function computation, generation of Fibonacci sequence, reversing the digits of an integer, base conversion.

UNIT - II BASICS OF PYTHON PROGRAMMING

Introduction to Python, Python Character set, Tokens, Core Data Types, variables, Assignment of values to variables, Writing simple programs in Python, input() and eval() functions. **Operators and Expressions:** Arithmetic Operators, Operator Precedence and Associativity, Bitwise Operators and Boolean Operators

UNIT - III CONTROL FLOW, FUNCTIONS AND STRINGS

Decision Statements: if statement, if-else statement, nested if statement, multi-way if-elif-else statement; **Loop Statements:** while loop, range() function, for loop, nested loops, break and continue statements; **Functions:** Syntax and basics of a function, parameters and arguments in a

function, return statement, recursive functions, Lambda function; **Strings:** Basic Python functions for String, String operators, String Operations

UNIT - IV LISTS, TUPLES AND DICTIONARIES

Lists: Creating Lists, Slicing Lists, List Methods, Lists and Strings, Lists with Functions, Searching and Sorting in lists; **Tuples:** Creating tuples, inbuilt functions for tuples, Lists and Tuples. **Dictionaries:** Basics of dictionaries, Creating a dictionary, Formatting dictionaries, Methods of Dictionary class

UNIT - V FILE HANDLING, MODULES AND PACKAGES

File Handling: Introduction, Working with text files; Modules: Definition, Creating a Module, Standards modules of Python; Packages: Definition, Importing * from packages, Packages in multiple directories

Course Outcomes:

After completion of the above courses the student will be able to

1. Able to write algorithms for computational problems
2. Work with basic programs in python
3. Write programs in python for looping structures, functions and string handling mechanisms
4. Execute programs on creation of lists, tuples and dictionaries using python language
5. Implement file handling mechanisms, modules and packages of python language

Text Books:

1. How to Solve it by Computer- R G Dromey
2. Programming and Problem Solving with Python by Ashok NamdevKamthane and Amit Ashok Kamthane, McGraw Hill Education; First edition November 2017

Reference Books:

1. Think Python: How to Think Like a Computer Scientist- Allen B. Downey
2. Python Programming – An Introduction to computer science, John Zelle,JimLeisy
3. Programming Python, Mark Lutz, O'Reilly, 3rd Edition, 2006
4. Core Python Programming, Wesley J Chun, PH, 2nd Edition
5. Python Programming: A Compatible Guide for Beginners to Master and Become an Expert in python programming Language, Brain Draper, CreateSpace Independent Publishing Platform, 2016

Mode of Evaluation: Assignments, Internal Mid Examination, External End Examination

MCA I Year I Semester

18ENGP201 ENGLISH FOR COMMUNICATION LABORATORY

L	T	P	C
0	0	3	1.5

Course Prerequisite: None

Course Description:

English Communication Lab is designed to enhance learners' spoken abilities in English. The nature of the course is purely practical in nature and it gives more emphasis on improving speaking skills of the learners. The first unit of the course introduces learners to the sounds of English and gives practice in it. The second unit of the course provides voice accent training to the learners and assists them in developing their own accent. The third and fourth units of the course expose learners to a wide variety of speaking activities. The primary aim of these two units is to enhance learners' day-to-day communication and improve their academic speaking performance. The final unit makes learners ready for the interview. Overall, the primary aim of the course is to enhance learners' spoken proficiency.

Course objectives

To facilitate computer aided multimedia instructions enabling individual listed and independent language learning

- a) To make learners aware of speech mechanism and provide accent training.
- b) To improve learners' day today spoken proficiency and enhance learner's interpersonal skills
- c) To develop learners' ability to face interviews

UNIT I

Introduction to Sounds of Speech, Vowels Consonants Identification of Phonetic Transcription

UNIT II

Introduction, Syllables Accent, Word Stress Rules of Words Stress, Intonation

UNIT III

Role Play, Situational Dialogue, Giving Directions, Describing Objects, Jam

UNIT IV

Group Discussion & Debates

UNIT V

Interview Skills

Course outcomes

After completion of the course:

1. Learners will be good at pronunciation of words
2. Learners will be good at interpersonal skills
3. Learners will be active participants in GD, Debate and ready to face interview

Mode of Evaluation: Continuous Cumulative evaluation of the lab experiments, Record, Viva voce and External Lab Examination.

MCA I Year I Semester

18MCAP201 PC HARDWARE AND NETWORKING PRACTICAL

L	T	P	C
0	0	3	1.5

Course Prerequisite: None

Course Description:

Student can able to get knowledge and working environment about the hardware devices used in the computer like motherboard, RAM, Processor, SMPS and Printers and Memory Hierarchy etc.

Course objectives:

1. Indicate the names and functions of hardware parts.
2. Identify the names and distinguishing features of different kinds of I/O devices and memory storage devices.
3. Understand various hardware and networking components

List of Experiments:

1. List and Identify the various External and Internal components of a Personal Computer (PC)
2. Identify and studying the different parts of a mother board (List the various latest CPUs, MOTHERBOARDS, RAM, MONITER, HARDDISK available in the present market)
3. Study of SMPS and measuring the various output voltages and identifying the cards/components/systems to which they are connected
4. Study of ROM BIOS, Hardware configuration, the CMOS set-up and the effect of various parameters on the working of the system
5. Exercise on disassembling and assembling of PC
6. Installation of WINDOWS XP operating system by partitioning the hard disk (at least two partition) along with device driver's initializations.
7. Installation of LINUX operating system executing Basic LINUX commands and Creating users / group users, Assigning permissions.
8. Installing and using of any application Software, Antivirus software & PC diagnostic tools.
9. Establish LAN between Linux & Windows Systems & Share Printer Connecting & disconnecting computer peripherals and components & driver installation
10. Hard disk formatting, partitioning and initialization.

11. Wire the procedure to connect external devices like printer/scanner/external Hard disk etc.
12. Identify the different browsers with their unique technical features along with text, images, documents with multimedia.
13. Write the procedure of different compression/ conversion tools which is available online/ offline.

Course outcomes:

After the completion of the course, Students will be able to

1. Understand the functions of different hardware components and motherboard
2. Identify the I/O interfaces, primary and secondary storage devices
3. Identify and differentiate various networking components

References:

1. The Complete Reference PC Hardware –Tata McGraw Hill Edition 2007.
2. The Indispensable PC Hardware -Hans-Peter Mesmer 4th edition.

Mode of Evaluation: Continuous Cumulative evaluation of the lab experiments, Record, Viva voce and External Lab Examination.

Course Prerequisite: None

Course Description:

This course provides the students how to write algorithms for computational problems and how to write programs in python language to perform different tasks such as functions, string handling mechanisms and file handling concepts.

Course Objectives:

Student will be able to

1. Learn to write algorithms for different standard tasks
2. Understand the concepts of drawing flow charts using raptor tool and identify how it works
3. Get idea on writing programs for different tasks using different syntax and available functions in python language

LIST OF EXPERIEMENTS:

1. Develop flow charts using Raptor tool
 - a. To exchange the values of two variables
 - b. To find factorial of a given number
 - c. To find the sum of set of numbers
 - d. To count the number of students that passed the examination
2. Develop flow charts using Raptor tool
 - a. To compute sine function
 - b. To generate Fibonacci sequence
 - c. To reverse the digits of a number
 - d. To convert a decimal integer to its octal representation
3. (a) Python installation for WINDOWS/LINUX
(b) Working with interactive and script modes in IDLE
(c) Exploring IDLE(Python Help docs etc.)
4. Write Python scripts
 - (a) To read two numbers and perform an arithmetic operation based on the option chosen by the user.
 - (b) To find the grade of a student based on the marks given for all subjects in a semester
 - (c) To find the GST of a given item
5. (a) Write a program to generate a multiplication table for a given number

- (b) Write a program using loop that asks the user to enter an even number. If the number entered is not even then display an appropriate message and ask them to enter a number again. Do not stop until an even number is entered. Print a Congratulatory message at end.
6. Write Python scripts for the tasks given in Exercise/Week #1
 7. Write Python scripts for the tasks given in Exercise/Week #2
 8. Write functions in Python
 - (a) To find the distance between two points
 - (b) To find the area of a circle
 - (c) To find the factorial of a number using recursion
 - (d) To find the GCD of two numbers using recursion
 9. Consider the string `str="Global Warming"`. Write statements in Python to implement the following
 - (a) To display the last four characters.
 - (b) To display the substring starting from index 4 and ending at index 8.
 - (c) To check whether string has alphanumeric characters or not.
 - (d) To trim the last four characters from the string.
 - (e) To trim the first four characters from the string.
 - (f) To display the starting index for the substring "Wa".
 - (g) To change the case of the given string.
 - (h) To check if the string is in title case.
 - (i) To replace all the occurrences of letter "a" in the string with "*"
 10. Write a program to find the sum of two matrices using lists.
 11. We can use list to represent polynomial. For Example $p(x) = -13.39 + 17.5x + 3x^2 + x^3$ can be stored as `[-13.39, 17.5, 3, 1.0]` Here "index" is used to represent power of "x" and value at the index used to represent the coefficient of the term. Write a function to evaluate the polynomial for a given "x".
 12. Create a list that contains the names of 5 students of your class. (Do not ask for input to do so)
 - (i) Print the list
 - (ii) Ask the user to input one name and append it to the list
 - (iii) Print the list
 - (iv) Ask user to input a number. Print the name that has the number as index (Generate error message if the number provided is more than last index value).
 - (v) Add "Kamal" and "Sanjana" at the beginning of the list by using "+".
 - (vi) Print the list
 - (vii) Ask the user to type a name. Check whether that name is in the list. If exist, delete the name, otherwise append it at the end of the list.
 - (viii) Create a copy of the list in reverse order
 - (ix) Print the original list and the reversed list.
 - (x) Remove the last element of the list.
 13. Write a python program to create telephone directory. Input 10 names and phone numbers to store it in a dictionary and do the following:
 - (a) Input any name and print the phone number of that particular name.
 - (b) Add a new contact
 - (c) Update an existing contact

- (d) Delete a contact
 - (e) Sort the directory according to names.
14. Write a program to input 'n' employees' salary and find minimum & maximum salary among 'n' employees using Tuples.
 15. (a) Write a program to create a file "Story.txt" with the history of MITS
(b) Write a program to read a file 'Story.txt' and create another file, storing an index of Story.txt telling which line of the file each word appears in. If word appears more than once, then index should show all the line numbers containing the word.

Case Study#1/ Creating Modules

Case Study#2/ Working with NumPy package

Course Outcomes:

After completion of the above course student will be able to

1. Draw flow charts for different tasks using raptor tool
2. Write python basic programs using conditional and looping structures
3. Execute programs in python for string handling, functions, create modules and work with packages

Text Books:

1. Python Programming – An Introduction to computer science, John Zelle, Jim Leisy
2. Programming and Problem Solving with Python by Ashok Namdev Kamthane and Amit Ashok Kamthane, McGraw Hill Education; First edition (1 November 2017)

Reference Books:

1. Programming Python, Mark Lutz, O'Reilly, 3rd Edition, 2006
2. Core Python Programming, Wesley J Chun, PH, 2nd Edition
3. Python Programming: A Compatible Guide for Beginners to Master and Become an Expert in python programming Language, Brain Draper, CreateSpace Independent Publishing Platform, 2016

Mode of Evaluation: Internal Practical Examination, External End Semester Practical Exam

Course Prerequisite: None

Course Description

This course gives hands on experience to learn widely used applications including word processing, spreadsheets, databases, presentation

Course Objectives

Students will be able to:

1. Provide the student on hands-on experience and technical training in Word processors, Spreadsheets, MS PowerPoint.
2. Gain knowledge about the Database.
3. Exposed to Latex tool and Google Documents.

List of Experiments:

1. Write procedures to Creating a Newsletter: Features to be covered, Table of Content, Newspaper columns, Images from files and clipart, Drawing toolbar and Word Art, Formatting Images, Textboxes, Paragraphs and Mail Merge in word.
2. Write a procedure to create a worksheet with 4 columns, enter 10 records and find the sum of all columns.
3. Write a procedure to create a report containing the pay details of the employee.
4. Write a procedure to Calculating GPA - .Features to be covered:- Cell Referencing, Formulae in excel – average, std. deviation, Charts, Renaming and Inserting worksheets, Hyper linking, Count function, LOOKUP/VLOOKUP, Sorting, Conditional formatting .
5. Write a procedure to create a simple bar chart to high light the sales of a company for 3 different periods.
6. Write a procedure to create a pie chart for a sample data and give legends
7. Write a procedure to create a worksheet importing data from database and calculate sum of all the columns.
8. Write a procedure to create a worksheet sale and report by using functions.
9. Write a procedure to create a worksheet employee details by using vlookup and hlookup.
10. Write a procedure to create simple table for result processing in MS access.
11. Write a procedure to implementation of project documentation in below mention problems
 - a. Create Sections,
 - b. References,
 - c. Figures
 - d. Tables
 - e. Math Mode

12. Write a Procedure to implementation of different types of data in Google documents by using these operation like create, update, sharing, uploading and delete the documents by using the Google documents
13. Imagine you want a buy a new car from company T. The company's list price leaflet gives the following information.
 - a. List of price of cars below the Rs 10,0000.
 - b. Depost of 25% of list of price is 156,500
 - c. Over 48 months, the monthlyly payment will be rs 10,838
 - d. Over 60 month, rs 8921 per month
 - e. Over 72 months, rs 7734 per month

Use the Goalseek and calculate the rate of interest of car and company charging.

Use the Goalseek and calculate the monthly payment and rate of interest of months 48, 60 and72.

14. Calculate the tax with the help different types of reference functions.

(A)

Employee	Salary	Tax	Tax - 30%
Bill, Carson	\$ 50,784		
Lino, Ramirez	\$ 32,202		
Alex, Rodriguez	\$ 14,237		
Myers, Marc	\$ 22,475		
Lambert, Jody	\$ 63,963		
Larsen, Lara	\$ 61,214		
Mercado, David	\$ 84,121		

(B) **Price per Unit**

		100	200	300	400	500	600	700	800
Unit	10								
	20								
	30								
	40								
	50								
	60								
	70								

80
90
100

15. Fill the column **E** with all the respective closing balances for that day. Write a formula to get the closing balance on **January 22nd** on a practical basis i.e. how it is calculated in a typical Bank statement.

A	B	C	D	E
Date	Particulars	Debits	Credits	Balance
Jan 01, 2014	Salary Accenture for Jan-2014		\$ 78,485	
Jan 02, 2014	POS-Samsung Smartphone	\$ 16,800		
Jan 02, 2014	POS-Joy Alukkas	\$ 29,050		
Jan 02, 2014	POS-KFC Banjara Hills	\$ 800		
Jan 03, 2014	HSBC Credit Card Min Due	\$ 600		
Jan 05, 2014	POS-Woodland Shoes	\$ 2,800		
Jan 07, 2014	NEFT transfer by Best Friend		\$ 25,000	
Jan 08, 2014	Flipkart	\$ 5,235		
Jan 08, 2014	POS Dell	\$ 43,990		

16. 16.

Sno	Acct Name	AccType	A/cOpenDate	Age	Deposits	Bank	City	Demat	AQB
1	Sanders, Troy	Savings	Jan 7, 1988	60	\$ 656,309	Citi	Calcutta	No	\$ 76,159
2	Brian, Rivera	Savings	Jan 11, 1988	87	\$ 137,688	DCB	Mangalore	No	\$ 27,028
3	Roberto, Hernandez	Business	Jan 15, 1988	45	\$ 1,200	Kotak	Hyderabad	Yes	\$ 75,570
4	Lino, Ramirez	Savings	Feb 23, 1988	70	\$ 850,888	ICI CI	Mysore	Yes	\$ 43,858

5	Hancock, Allen	Busin ess	Feb 24, 1988	71	\$ 144,745	Citi	Mysore	No	\$ 49,776
6	Hodge, Craig	Busin ess	Mar 31, 1988	65	\$ 649,318	ICI CI	Mysore	No	\$ 76,808
7	Gabriel, Hierra	Busin ess	Apr 8, 1988	52	\$ 609,169	HS BC	Cochin	Yes	\$ 77,565
8	Joel, Jiminez	Busin ess	Apr 11, 1988	71	\$ 54,233	HD FC	Pune	Yes	\$ 29,083
9	Huffman, Ignacio	Busin ess	Apr 19, 1988	84	\$ 493,309	IDB I	Chennai	Yes	\$ 14,298
10	Combs, Rick	Busin ess	Apr 22, 1988	49	\$ 313,530	IDB I	New Delhi	No	\$ 85,279

A. Calculate below mentioned functionalities

- a. COUNT
- b. COUNTA
- c. COUNTBLANKS
- d. COUNTIF

B. Calculate number of Banks in different branches in the above mentioned table.

Bank	Howmany ?
SBI	
HDFC	
ICICI	
Axis	
DCB	
HSBC	

Bank	City	Howmany?
HSBC	Ahmedabad	
Citi	Ahmedabad	
Citi	Calcutta	
SBI	Jaipur	

17. To create an interactive presentation, read the instructions on the following pages for an own idea. For this exercise, you will create a PowerPoint slide show that includes the following

1. A “homepage” slide with a title and links or buttons to each of your other five pages.
2. A slide that uses arrows (and/or circles) and text to label different aspects of a photograph or graphic Image.
3. A slide that uses no words but is an image collage that builds itself automatically. (In other words, images appear on the page one-at-a-time, automatically timed).
4. A slide that has a bulleted list of items that are animated so that the items appear on the page one-at-a-time. Set it up so that each item dims as the next item appears.
5. A slide that uses a sound effect that is integrally tied to the information on the page (either as an auditory example of something you are talking about, or as an element that helps to create the context).
6. A slide that links to a web page.

Course Outcomes:

After completion of the above course student will be able to

1. To know how prepare the Documents effectively using word processors, spread sheets, presentation.
2. Understanding the MS Access & Database.
3. To know how to implement project documentation by using the latex tool and Google documents.

Text Books

1. Office 2016 All-In-One for Dummies 1st Edition by Peter Weverka .
2. Exploring Microsoft Office 2016 Volume 1 (Exploring for Office 2016 Series) 1st Edition by Mary Anne Poatsy (Author).

Reference Books

1. Shelly Cashman Series Microsoft Office 365 & Office 2016: Introductory (MindTap Course List) 1st Editionby Misty E. Vermaat.
2. Microsoft Office 2016 Essentials Quick Reference Guide - Windows Version (Cheat Sheet of Instructions, Pamphlet – November 4, 2015by Beezix Inc
3. 3.Skills for Success with Microsoft Office 2016 Volume 1 (Skills for Success for Office 2016 Series) 1st Editionby Margo Chaney Adkins

Mode of Evaluation: Continuous Cumulative evaluation of the lab experiments, Record, Viva voice and External Lab Examination.

MCA I year II Semester

18MATP114

PROBABILITY AND STATISTICS

L T P C
4 0 0 4

Course Prerequisite: *Calculus, Linear Algebra*

Course Description:

This course provides an introduction to probability, distributions and statistics with applications. Topics include: Conditional probability, Random variables, Probability distributions, Joint densities, Correlation and linear regression. Hypothesis testing, Confidence intervals, Analysis of Variance.

Course Objectives:

The objectives of this course are

1. To revise the elementary concepts of probability
2. To extend and formalize knowledge of the theory of probability and random variables.
3. To analyze and interpret basic summary and modeling techniques for Multi-variate data
4. To introduce new techniques for carrying out probability calculations and identifying probability distributions.
5. To understand the foundations for statistical inference involving confidence intervals and hypothesis testing.

UNIT- I: Probability and Random variables: [3.1 - 3.8, 4.1, 4.2, 4.4, 4.5, 4.6, 4.8] (12 hours)

Introduction to probability, theorems on probability, conditional probability, multiplication theorem and independence of events, Bayes theorem. Random Variables - types of random Variables - Probability mass Function - Probability density function- Distribution function and its properties- Expectation, Variance and their Properties

UNIT-II Bi-variate random variables [4.3, 4.3.1, 4.3.2, 4.5.1, 4.7, 2.6, 9.1 & 9.2] (12 hours)

Joint Densities and Independence - Marginal Distributions (discrete & continuous)- Transformation of Random Variables - Conditional Distributions and Expectations –Covariance- Correlation - Multiple Linear Regression Models.

UNIT-III Probability Distributions [5.1, 5.1.1, 5.2, 5.2.1, 5.3, 5.4, 5.5, 5.6, 5.7 & 5.8.1] (12 hours)

Discrete Distributions: Bernoulli trail, Binomial distribution, Poisson approximation to the binomial distribution , Poisson distribution and Hyper geometric distribution –properties.

Continuous Distributions: Uniform distribution, Exponential distribution, Gamma distribution, Normal distribution.

UNIT-IV Hypothesis Testing [8.1, 8.2, 8.3, 8.4, 8.6] (12 hours)

Hypothesis Testing- Introduction, Significance Levels, Tests Concerning the Mean of a Normal Population (σ know and unknown), Testing the Equality of Means of Two Normal Populations , Case of Unknown and Unequal Variances , The Paired t -Test , Hypothesis Tests Concerning proportions.

UNIT-V Analysis of Variance (ANOVA) [10.1-10.5] (12 hours)

Analysis of variance (ANOVA); one way classification and two-way classifications. Latin square Design and RBD.

Text Books:

1. Sheldon M. Ross: Introduction to Probability and Statistics for Engineers and Scientists, 4th Edition, Elsevier, Academic Press, 2010.
2. J.S. Milton and J.C. Arnold, Introduction to Probability and Statistics, 4th edition, 2003 Tata McGraw-Hill Publications.

Reference Books:

1. Walpole, R.E., Myers R.H., Myer S.L., Ye. K: Probability and Statistics for Engineers and Scientists, 8th ed., Pearson Education, 2008.
2. Johnson, R.A. Miller Freund's: Probability and Statistics, 7th Edition, PHI, 2005.
3. Sheldon Ross: A First Course in Probability, 6th Edition, Pearson Education, 2002.

Course Outcomes:

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

1. Understand the probability concepts and their importance in engineering.
2. Apply discrete and continuous probability distributions to solve various engineering problems.
3. Get an idea about joint density functions, distribution functions to the random variables and analyze the multivariate problems in engineering
4. Apply the method of least squares to estimate the parameters of a regression model.
5. Perform Test of Hypothesis as well as calculate confidence interval for a population parameter for single sample and two sample cases.

Mode of Evaluation: Assignments, Internal Mid Examinations, External End Examination

MCA I Year II Semester

18MCAP103

PROGRAMMING IN C

L T P C

4 0 0 4

Course Prerequisite: None

Course Description:

The course is designed to provide complete knowledge of C language. Students will be able to develop logics which will help them to create programs, applications in C. Also by learning the basic programming constructs they can easily switch over to any other language in future.

Course Objectives:

Students will be able to

1. Identify, formulate and solve computing problems using C programming language.
2. Understand the various steps in program development
3. Learn how to write modular and readable programs
4. Understand the limitations of basic datatypes and learn the concepts of derived datatypes and user defined data types.
5. Gain the knowledge of record and file manipulation.

UNIT I: C FUNDAMENTALS

The C-Character Set, Identifiers and Keywords, Constants, Data types, Variables, Statements, Symbolic Constants Operators and Expressions, Precedence and Associativity Data Input and Output, Structure of the C Program

UNIT II: CONTROL STRUCTURES

Branching-go to, Simple if, Simple If- else, Nested if-else, if –else ladder Looping- while, do-while, for – Nested Control Structures Switch and Break Statements, continue Statement, comma Operator

UNIT III: ARRAYS AND FUNCTIONS

Arrays: Initialization, Declaration, One dimensional, Two dimensional arrays and Multidimensional arrays, Applications of an array, Matrix operations, Searching techniques

Function: Definition, Declaration, Call by value and Call by reference, Recursion, Storage class.

Strings: Operations, String arrays, Simple programs with and without string functions

UNIT IV: POINTERS, STRUCTURES AND UNIONS

Pointers: Definition, Initialization, Pointers arithmetic, Pointers and arrays, Array of pointers, Dynamic Memory Allocation.

Structures: Introduction, Need for structure data type, definition, declaration, Structure within a structure.

Union: Need for union datatype, Definition, Declaration, Programs using structures and Unions- Typedef, Enum

UNIT V: FILE PROCESSING AND RECENT TRENDS

Files, Types of File Processing, Sequential, Random Access, Opening and Closing a Data file, Creating and Processing a Data file, Command Line Arguments, Low level Programming

Course Outcomes:

After Completion of the Course Students will be able to:

1. Understand the basic concepts of programming techniques.
2. Choose the loops and decision making statements to solve
3. Implement different operations on arrays and use functions to solve the given problems
4. Understand pointers, structures and unions.
5. Implement the file operations in C

Text Books:

1. Programming with C. Byron S Gottfried Schaum's Outlines, Second Edition, Tata McGraw-Hill, 2006.
2. The C Programming language, Kernighan,B.W and Ritchie,D.M, "", Second Edition, Pearson Education, 2006

Reference Books:

- 1 Fundamentals of Computing and Programming in C ,PradipDey, Manas Ghosh, First Edition, Oxford University Press, 2009
- 2 Programming in C ReemaThareja, I, Oxford University Press, Second Edition, 2016.
3. Let Us C ,Yashavant P. Kanetkar, BPB Publications, 2011.
4. Programming in ANSI C ,Balaguruswamy

Mode of Evaluation: Assignments, Internal Mid Examination, External End Examination

MCA I Year II Semester

18MCAP104

DATABASE MANAGEMENT SYSTEMS

L	T	P	C
4	0	0	4

Course Prerequisite: None

Course Description:

This course is designed to investigate how database management system techniques are used to design, develop, implement and maintain database applications in organizations.

Course Objectives:

Students will be able to:

1. Understand data base concepts and applications, data base system structure and ER data modeling.
2. Introduces relational model, integrity constraints, keys, implementation of SQL and Relational Algebra.
3. Provide an over view of schema refinement and Normal Forms.
4. Understand the concept of transaction and its management, concurrency control schemes and crash recovery techniques.
5. Understand the concepts of PL/SQL and NO SQL.

UNIT I: INTRODUCTION

Introduction to data bases, Database system Vs File system, Data abstraction, Instances and Schemas, Database users, Database system structure, Entities, Attributes, Entity sets, Relationships and Relationship sets, Database design and ER diagrams, Specialization and Generalization and participation features of ER model.

UNIT II: THE RELATIONAL MODEL, SQL & RELATIONAL ALGEBRA

The Relational model: Introduction, Codd's rules, Various types of Integrity constraints.

SQL: Structured Query Language, data types, DDL, DML, TCL, views, NULL Values, Set operations, aggregate functions, character functions, Date functions, String functions, Nested queries, correlated nested queries.

Relation Algebra: Fundamental operations, Additional operations.

UNIT III: NORMALIZATION

Problems Caused by redundancy, Decompositions, Reasoning about FDS , FIRST, SECOND,THIRD Normal forms, BCNF, Lossless-join Decomposition, Dependency preserving, Multi valued Dependencies , Fourth Normal Form and Fifth Normal form.

UNIT IV: TRANSACTION MANAGEMENT, CONCURRENCY CONTROL & RECOVERY SYSTEMS

Transaction Management: ACID properties, Transactions and Schedules, Concurrent Execution of transactions, Serializability, Conflict serializability, View serializability, Testing for serializability.

Concurrency Control: Two Phase Locking protocol, Time stamp ordering protocol, Multiple Granularity.

Recovery Systems: Deferred database modification, Immediate data base modification, check pointing and Shadow paging.

UNIT V: PL/SQL & NO SQL

PL/SQL: Functions, procedures, triggers, cursors and exceptional handling in PL/SQL.

NO SQL: No SQL Introduction and properties, NO SQL Columnar families, NOSQL_different NoSQL Systems.

Course Outcomes:

After completion of the above courses the student will be able to

1. Apply the basic concepts and advantages of database systems to an organization.
2. Demonstrate the relational data model, implementation of various SQL queries and relational algebra operations.
3. Deal with the different issues involved in the database design using Normal Forms.
4. Explain the basic issues of transaction processing, concurrency control and crash recovery.
5. Execute a relational database system using PL/SQL.

Text Books:

1. Data Base Management Systems, Raghurama Krishnan, Johannes Gehrke, TMH
2. Data Base System Concepts, 6/e, Silberschatz, Korth, TMH.

References:

1. Data Base Management System, 5/e, ElmasriNavathe, Pearson
2. An Introduction to Database Systems, 8th Edition, C J Date, Pearson

Mode of Evaluation: Assignments, Internal Mid Examination, External End Examination

Course Prerequisite: None

Course Description:

This course introduces the basic concepts of operating systems and their structures. The working concepts of Memory and Process Management are explained. It also covers an introduction to Unix shell scripting and BASH commands.

Course Objectives:

Student will be able to:

1. Understand the basic concept of operating system and types of operating system.
2. Analyze the Basic instructions of Unix Environment.
3. Write the CMD programs using Born Shell.
4. Understand the concept of CPU Scheduling
5. Understand Memory Management and Storage Management

UNIT I: OPERATING SYSTEM INTRODUCTION

Operating Systems objectives and functions, OS Structure, OS Operations, Evolution of Operating Systems - Simple Batch, Multi programmed, time-shared, Personal Computer, Parallel, Distributed Systems, Real-Time Systems, Special -Purpose Systems, Operating System services, User OS Interface, System Calls, Types of System Calls, System Programs, Operating System Design and Implementation, OS Structure, Virtual Machines.

UNIT II: UNIX INTRODUCTION TO SHELL & LINUX UTILITIES

UNIX Features and Environment, UNIX Structure, Accessing UNIX, File handling utilities, Security by file permissions, Process utilities, Disk utilities, Networking commands. Grep command, Vi Editor, SED – scripts, operation, addresses, commands, applications. awk – execution, fields and records, scripts, operation, patterns, actions, functions, uses system commands in awk.

UNIT III: WORKING WITH THE BOURNE AGAIN SHELL (BASH)

Introduction, shell responsibilities, pipes and input Redirection, output redirection, command Execution, command line editing, quotes, command substitution, shell variables, Filters, shell meta characters, shell programming-control structures, arithmetic in shell, shell script examples, interrupt processing, functions, debugging shell scripts.

UNIT IV: PROCESS

Overview of Process, Process Scheduling, scheduling Algorithm, Multiple-Processor Scheduling, The Critical Section Problem, Synchronization Hardware, Semaphores, Classic Problems of Synchronization. Deadlocks - System Model, Deadlock Characterization, Methods for Handling

Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection and Recovery from Deadlock.

UNIT V: MEMORY MANAGEMENT & STORAGE MANAGEMENT

Logical & Physical Address Space, Swapping, Contiguous Allocation, Paging, Structure of Page Table, Segmentation, Segmentation with Paging., File System – File Concept, Access methods, Directory Structure, File System Mounting, File Sharing, Protection. Mass Storage Structure – Overview of Mass Storage Structure, Disk Structure, Disk Attachment, Disk Scheduling, Disk Management, Swap space Management, Free-space Management.

Course Outcomes:

After Completion of the course, the student should be able to

1. Classify the operating systems, general architecture and structures of operating systems.
2. Recognize various levels of user accesses and to control the environment through shell scripting.
3. Implement functions, quoting mechanisms, control structures, substitutions and interrupt processing using BASH.
4. Schedule the processes without dead locks with inter process synchronization.
5. Explain memory management and kernel management.

Text Books:

1. Operating System Principles , Abraham Silberchatz, Peter B. Galvin, Greg Gagne 8th Edition, Wiley Student Edition
2. Unix and Shell programming, B.A. Forouzan and R.F.Gilberg, Cengage Learning

Reference Books:

1. Modern Operating Systems, Andrew S Tanenbaum, 3rd Edition, Pearson/PHI
2. Operating Systems – Internals and Design Principles, W. Stallings, 6th Edition, Pearson Education.
3. Beginning Linux Programming, 4th Edition, N. Matthew, R. Stones, Wrox, Wiley India Edition, rp-2008.

Mode of Evaluation: Assignments, Internal Mid Examination, External End Examination

Course Prerequisite: None

Course Description:

This course provides basics about software engineering process and models. It introduces concepts such as software processes and agile methods, and essential software development activities, from initial specification through to system maintenance. There is a focus on software testing, from unit testing to the testing of software releases. Project, cost, risk and configuration management with professional software engineering practices will also be covered.

Course Objectives:

Students will be able to

1. Gain an insight into the software development processes of models.
2. Understand the importance of requirements and requirement modelling.
3. Gain knowledge about design, measures and metrics to develop the product with quality and reliability.
4. Understand testing, types of testing and quality concepts.
5. Gain the knowledge about project, risk and configuration managements.

UNIT I: SOFTWARE ENGINEERING AND MODELS

Software Engineering Paradigms: Software Characteristics, Software myths, Software Applications, Software Engineering Definitions, Software Process Models, Process iteration, Process activities. Project planning, Project scheduling.

UNIT II: REQUIREMENTS AND MODELING

Understanding Requirements, Requirement Analysis, Scenario Based Modeling, Data Modeling Concepts, Class Based Modeling. Requirement Modeling: Flow based, Creating Behavioural Model, Requirement Modeling for WebApps.

UNIT III: DESIGN AND METRICS

Software Design Concepts: Process, Concepts, Models. Architectural Design, Component level Design, User Interface Design, WebApp Design.

Product Metrics: Framework for Product Metrics, Metrics for Requirement Model, Design Model, Webapps, Source Code, Testing and Maintenance.

Process and Project Metrics: Metrics in Process and Project Domain, Software Measurement, Quality, Integrating metrics with Software Process, Metrics for Small Organization.

UNIT IV: TESTING AND QUALITY ASSURANCE

Software Testing Strategies, Testing Conventional Application, Testing Web Applications
Software Quality Management Quality Concepts Review Techniques, Quality Assurance

UNIT V: SOFTWARE MANAGEMENT

Managing Software Project: Project management Concepts, Estimation for Software Projects, Risk Management, S/w Configuration Management, S/w Maintenance and Re-Engineering

Course Outcomes:

After completion of the above course the student will be able to

1. Understand basics of software development process and various models.
2. Develop SRS and SRS models of software engineering.
3. Model software projects by understanding the concepts of design and can measure the product and process performance using various metrics
4. Evaluate the quality of the system using various testing techniques and strategies
5. Manage project, risk and configuration issues during the software development phases.

Text Books:

1. Software Engineering, Roger S Pressman, TMH.
2. Software Engineering, Gill N.S.: Khanna Book Publishing Co.(P) Ltd, N. Delhi
3. An Integrated Approach to Software Engineering, Jalote, Pankaj: Narosa Publications.

Reference Books:

1. R. Fairley, "Software Engineering Concepts", Tata McGraw Hill, 1997.
2. P. Jalote, "An Integrated approach to Software Engineering", Narosa, 1991.
3. Stephen R. Schach, "Classical & Object Oriented Software Engineering", IRWIN, 1996.
4. James Peter, W Pedrycz, "Software Engineering", John Wiley & Sons
5. Sommerville, "Software Engineering", Addison Wesley, 1999.

Mode of Evaluation: Assignments, Internal Mid Examination, External End Examination

L	T	P	C
0	0	3	1.5

Course Prerequisite: None

Course Description:

This course is an introduction to the theory and practice of computer programming, the emphasis of this course is on techniques of program development within the structure and object-oriented paradigm. Topics include C program basics, control structures, arrays, files, pointers, objects, classes, inheritance, and data structures.

Course Objectives:

Students will be able to

1. Learn C Programming Language
2. Write C programs for real world problems using simple and compound data types
3. Employee good programming style, standards and practices during program development

LIST OF EXPERIMENTS:

1. I/O Statement

a) A person has deposited an amount in a bank. He / She would like to know the total amount as follows

- a. With simple of r% interest after 'n' years
- b. With compound interest of r% after 'n' years if compounded annually

Write a program to implement the above.

b) A General Service company hired you to calculate the labour cost and total charge for the services rendered to their client based on the formula given below:

Rate per hour : Rs.45/-

Labour cost : Rate * hours work

Total Charge : Labour cost + Cost of Materials.

Write a program to implement the above.

c) Mr.Bennet bought a new house and he wishes to paint the both sides of rectangular shaped compound wall of that house. He is having the following details: length, breadth, height and cost per square feet to paint the compound wall. Help him out to calculate the cost for painting work.

d)Write a C program to make the following exchange between the variables a-> b -> c->d -> a

2. I/O Statement

- a) A farm produces several food grains, namely wheat, barley, oats and flax. The monthly production details (in Kg) and price per Kg (in INR) received during the year are recorded. Read the production details and price of the food grains and print its details through your C program.
- b) (BMI) is a measure of body fat based on height (in inches) and weight (in pounds) that applies to adult men and women. Write a C program for the calculation of BMI as $BMI = (\text{weight} * 703) / (\text{height} * \text{height})$ and let the program read your height and weight and display the BMI value.
- c) ABC company Ltd. is interested to computerize the pay calculation of their employee in the form of Basic Pay, Dearness Allowance (DA) and House Rent Allowance (HRA). DA and HRA are calculated as certain % of Basic pay (For example, DA is 80% of Basic Pay, and HRA is 30% of Basic pay). They have the deduction in the salary as PF which is 12% of Basic pay. Propose a computerized solution for the above said problem.

3. Decision making (IF, IF...ELSE)

- a) The Election Commission has decided to organize a special camp to include young people (*age greater than or equal to 16 and less than 18*) in the electoral role. Help the officials to identify the eligible people.
- b) Apoorva super market requires an automated program to print the expenses. Quantity and price per item are the inputs and discount of 10% is offered if the expense is more than Rs. 5000. Automate the above scenario using 'C' program.
- c) Mr. X goes to a dietician for consulting. The dietician measures the weight in kilogram and height in meters. The dietician calculates the BMI using the following formula $BMI = \text{weight} / (\text{height})^2$ Once BMI is calculated the dietician decides whether Mr. X is obese or not. If BMI is greater than 25 then it implies Mr. X is obese else he is Healthy. Write a program depicting the above scenario with displaying the message obese or Healthy.

4. Decision making (ELSE IF Ladder, NESTED IF / Switch ...case)

- a) Given the previous and current month electricity meter readings, calculate the electricity bill amount for that month. The bill is calculated as follows: first 50 units of electricity are charged at Rs2 for each unit, next 50 units are charged at Rs2.50 for each unit, and units above 100 are charged at Rs. 3.50 for each unit. Each bill amount also has surcharge amount of Rs. 20.
- b) The class teacher wants to check the IQ of the students in the class. She is conducting a logical Reasoning, Verbal Reasoning, Arithmetic ability and puzzle logic test. Each of which carries 50 marks. Those who secured 180 and above marks are eligible for taking genius level test. Those who secured below 180 marks are rejected for genius level test. There are two levels of Genius test - Genius level 1 & Genius level 2. Those who secured above 60% marks for all test are eligible for taking Genius level 1 and for remaining students Genius level -2 will be conducted. Automate the task to help the class teacher.
- c) Write a C program to find the roots of the quadratic equation

5. Looping (FOR / WHILE)

- a) Our institution has decided to increase the number of seats from 120 to 180 for the upcoming year. Further, they have also decided to increase the same number of seats every year subsequently. Find out totally how many students will be there, after 'n' years from now.
- b) Write a C program for printing the prime numbers between 1 and n
- c) Write a C program to construct the multiplication table for a given integer.

6. Looping (FOR / WHILE / DO..WHILE)

- a) Write a program to print the calendar for a month given the first Week- day of the month. Input the first day of the month (Sun=0,Mon=1,Tue=2,Wed=3,.....) :: 3 Total number of days in the month : 31
Expected output
- b) A Novice user tries to sign up in Gmail. Gmail does not allow users of age less than 13 years old for sign up. Write a Program to validate the user age and allow the user to sign in Gmail only when the age is above 13. Otherwise display appropriate error message and provide another chance to sign in again.
- c) Write a C program to reverse the digits of a given integer

7. Looping (FOR / WHILE / DO..WHILE)

- a) Write a c program to find the sum of the digits of a positive integer
- b) Fibonacci sequence is defined as follows: the first and second terms in the sequence are 0 and 1. Subsequent terms are found by adding the preceding two terms in the sequence. Write a C program to generate the first n terms of the sequence.
- c) Write a C Program to print the pyramid of numbers
- d) Write a C program to print the pascal triangle

8. Single dimensional arrays

- a) Write a program to store and display the array elements
- b) Write C code to compute the frequency table of survey responses given by 20 users. The survey responses range from 1 to 5 and are stored in an array. For example, 10 responses are stored in the array [1,1,5,2,3,3,3,5,5,2,2]. The frequency table will be as shown below:
 - a. 1 = 2
 - b. 2 = 3
 - c. 3 = 2
 - d. 4 = 0
 - e. 5 = 3
- b) Write C code to reverse the contents of the array. For example, [1,2,3,4,5] should become [5,4,3,2,1]
- c) Write a program that will search and find out the position where the given key element exist in a user chosen array and print it as output.

9) Two dimensional arrays

a) Write a C program that uses functions to perform the following: i) Addition of Two Matrices
ii) Multiplication of Two Matrices

b) A cafe sells two types of cola drinks. The drinks each come in three sizes: small, medium and large. At the beginning of the day the fridge was stocked with the number of units shown in the matrix below. At the end of the day the stock was again counted.

	Start of the day		At the end of the day	
	Brand C	Brand P	Brand C	Brand P
small	→	[42 54]	small	→ [27 31]
medium	→	[36 27]	medium	→ [28 15]
large	→	[34 30]	large	→ [28 22]

The profit matrix for each item is:

	small	medium	large
	[\$0.75	\$0.55	\$1.20]

Implement the program to calculate the total profit made for the day from the sale of these drinks.

10. Sorting

a) Write a program to sort the array elements using exchange sort

b) There are 'n' employees working in an organization. The employer need to allocate the work based on the years of experience of the employees. Use an appropriate logic to help the employer to order the employees according to their years of experience.

11 Programs based on string Manipulations

a) MITS University Academic Section is maintaining student's details with full name (First name and Last name separated by space) in the database. Due to inconvenience in printing the names, they have decided to display it in short, as follows:

Full form: **First Name** **Last Name**

Arunkumar Tiwari

Short Form: Arunkumar T

Write a program to implement the above.

b) Principal of a school has been decided to divide the students into two houses (teams) namely Fire and Water based on their name. If the total count of their name is odd then they will be in the Fire house, else Water house. The total count is calculated by adding ASCII value of individual alphabet in their name. Help the Principal to complete the task.

c) Write a program to check whether the given string is a palindrome or not with or without using string handling functions

d) Write a program to accept a string from user and display number of vowels, consonants, digits and special characters present in each of the words of the given string

12. Program using Functions

- a) An application developer wants to read a list of 'n' fixed-point quantities in a 1-D array. The developer is assigned a task to search out a value in the array. Write a program using functions to search an item in 1-D array.
- b) Develop a C program to implement a calculator. The program should request the user to input two numbers and display one of the following as per the desire of the user:
- i. Sum of the numbers
 - ii. Difference of the numbers
 - iii. Product of the numbers
 - iv. Division of the numbers

Provide separate functions for performing various tasks such as reading, calculating and displaying the results.

- c) Write a program to find the factorial of the function without using recursion and using recursion
- d) Write a program to find the gcd of the two numbers with and without using recursion

13. Program using Functions and pointers

- a) The user-defined function chkprime() accepts an integer as parameter and returns True if the number is Prime. Write a program that prints sum of first 'n' prime numbers using chkprime().
- b) Write a C program to demonstrate on call by value and call by reference functions
- c) Write a program to read and display values of an integer array. Allocate space dynamically for the array using the malloc().

14. Programs using structures

- a) Mr. John runs a pizza-analysis service. For each pizza, he needs to record the following information:** The name of the pizza company, which can consist of more than one word, The weight of the pizza(in gms), price of the pizza **devise a structure in C that can hold this information and write a program to get the information about pizza from the user and then it should display the pizza details.**
- b)** A Departmental store is maintaining sales details (product code, product name, unit-rate, selling-date, quantity-sold and amount) of 'n' customers as a single logic unit (structure). Write a program to generate the bill for the customer.
- c)** The results of a survey of 'n' households in a township have been made available. Each record contains data for one household, including a 4 digit integer identification number, the annual income for the household and the number of family members of the household. Write a program to store the survey results into an array of user-defined structures of type **household**.

Then perform the following:

- i) Print a three column table displaying the data
- ii) Calculate the average household income and list the identification number and income of each household whose income exceeds the average.

iii) Determine the percentage of households having incomes below the poverty level using the formula given below:

$P = \$7500.00 + \$950.00 \times (m - 2)$ where 'm' is the number of members of each household.

15. Files

a) Two text files are given with the names text1 and text2. These files have several lines of text. Write a program to merge (first line of text1 followed by first line of text2 and so on until both the files reach the end of the file) the lines of text1 and text2 and write the merged text to a new file text3.

b) Write a program to split a given text file into n parts. Name each part as the name of the original file followed by .part<n> where n is the sequence number of the part file.

Course Outcomes:

After completion of the above course the student will be able to

1. Write, compile and debug programs in C language.
2. Choose programming components that efficiently solve computing problems in real-world.
3. Design programs involving different data types, control structures, functions and files.

Mode of Evaluation: Continuous Cumulative evaluation of the lab experiments, Record, Viva voce and External Lab Examination.

MCA I Year II Semester

18MCAP205 DATABASE MANAGEMENT SYSTEMS LABORATORY

L T P C
0 0 3 1.5

Course Prerequisite: None

Course Description:

This course provides an executive understanding and approach to the technical subject of database management system.

Course Objectives:

Students will be able to:

1. Provide formal foundation on the relational model of data.
2. Present SQL and procedural interfaces to SQL comprehensively
3. Introduced to systematic database design approaches covering conceptual and physical design.

List of Experiments:

1. Implement DDL Commands. (use constraints while creating tables).
2. Implement DML commands.
3. Implement Unique, NULL, NOT NULL, Primary key, Foreign key, Check constraints.
4. Queries (along with sub Queries) using ANY, ALL, IN, EXISTS, NOT EXISTS, UNION, INTERSET.
5. Queries using Aggregate functions (COUNT, SUM, AVG, MAX and MIN), GROUP BY, HAVING.
6. Queries using Conversion functions (to_char, to_number and to_date), string functions (Concatenation, lpad, rpad, ltrim, rtrim, lower, upper, initcap, length, substr and instr)
7. Queries using date functions (Sysdate, next_day, add_months, last_day, months_between) .
8. Implement all types of joins.
9. Calculate Employee Net salary and Gross salary for the following data using cursors.

If Basic salary \leq 8000 then DA is 8.18% of basic,

HRA is 20% of basic

PF is 20.18% of basic

PT = 60Rs

If Basic salary $>$ 8000 and \leq 14000 then

DA is 32% of basic,

HRA is 7% of basic

PF = 600Rs

PT = 80Rs

If Basic salary $>$ 14000 and \leq 16000 then

DA is 30% of basic

HRA is 6% of basic

PF = 600

PT = 160

10. Implement functions, procedures and triggers .
11. Implement User defined and System defined exceptions.

Case Studies:

- I. Employee management system
- II. Student Result information system
- III. Bank Management system
- IV. Library information system
- V. Hotel management system
- VI. Railway reservation system

Course Outcomes:

At the end of the course the student will be able to:

1. Populate and query a database using SQL DML/DDL commands.
2. Declare and enforce integrity constraints on a database using a state-of-the-art RDBMS.
3. Programming PL/SQL including stored procedures, stored functions, cursors and packages.

Text Books:

1. ORACLE PL/SQL by example, Benjamin Rosenzweig, Elena Silvestrova, Pearson Education 3rd Edition
2. SQL & PL/SQL for Oracle 10g, Black Book, Dr.P.S. Deshpande.

References:

1. Introduction to SQL, Rick F.VanderLans, Pearson Education.
2. Oracle PL/SQL Programming, Steven Feuerstein, SPD.
3. The Database Book, N.Gehani, Universities Press.
4. Database Systems using Oracle: A Simplified Guide to SQL and PL/SQL, Shah, PHI.

Mode of Evaluation: Continuous Cumulative evaluation of the lab experiments, Record, Viva voce and External Lab Examination.

L	T	P	C
0	0	3	1.5

Course Description:

A shell script is a computer program designed to be run by the Unix shell, a command-line interpreter. Typical operations performed by shell scripts include file manipulation, program execution, and printing text, pipes and filters, programming fundamentals.

Course Objectives:

Students will be able to:

1. Develop software in Linux/UNIX environments.
2. Learn the basic operating system concepts, effective command line usage, script and control structures and library functions in Linux/Unix environment.
3. Develop the shell script programming using AWK & SED.

List of Experiments:

1. Count the vowels, words, char, blank lines of a given file
2. To write a shell script to create a two files.
3. Palindrome or not.
4. To write a shell script that will count no. of files in each sub directory.
5. Display odd numbers in given range.
6. Display the sum of the digits.
7. Display the reverse of a number.
8. To write a shell script that accepts a file name, starting and ending line numbers as arguments and displays all the lines between the given line numbers.
9. To write a shell script that displays a list of all the files in the current directory to which the user has read, write and execute permissions.
10. To write a shell script that receives any number of file names as arguments checks if every argument supplied is a file or a directory and reports accordingly. Whenever the argument is a file, the number of lines on it is also reported.
11. To write a shell script to list all files in a directory.

12. To write a shell script to find a factorial of a given integer

13. Write a shell script to manipulate files and directories

Course Outcomes:

After completion of the course students will be able to

1. Implement the basic set of commands and utilities to develop software in Linux/UNIX systems.
2. Write the programs in basic structure and library functions in UNIX/Linux shell script Programming.
3. Write the shell script programming using AWK and SED.

Mode of Evaluation: Continuous Cumulative evaluation of the lab experiments, Record, Viva voce and External Lab Examination.

18MCAP107 OBJECT ORIENTED PROGRAMMING

L T P C

4 0 0 4

Course Prerequisite: Basics of Programming

Course Description:

Basics of Object Oriented Programming - objects, classes, polymorphism, inheritance, static and dynamic binding. Object Oriented Programming using Java-classes, interfaces, inheritance, polymorphism, method dispatch, features for encapsulation and modularity.

Course Objectives:

Students will be able to

1. Understand object oriented programming concepts, and apply them in solving problems.
2. Learn the principles of inheritance and polymorphism; and demonstrate how they relate to the design of abstract classes
3. Introduced to implement the packages and interfaces
4. Learn the concepts of multithreading and I/O streams
5. Learn the design of Graphical User Interface using swing controls.

UNIT I: INTRODUCTION TO OOPS CONCEPTS AND CLASSES

Introduction to Object Oriented Programming, Java Buzzwords, Java Programming Basics, Sample programs, Data types and operators, Control statements, Arrays, Strings, String Handling.

Classes: Classes, Objects, Methods, Constructors, this and static keywords, Method Overloading and Constructor Overloading, Access modifiers, Polymorphism, Inner Class.

Java 8 in Action: Lambdas, Streams and Functional Style Programming

UNIT II: INHERITANCE, INTERFACES, PACKAGES & EXCEPTION HANDLING

Inheritance: Basics, Usage of Super, Multilevel Inheritance, Hierarchical Inheritance, Method overriding, Abstract Class, Wrapper Classes, Final keyword.

Interfaces: Creation and Implementation of Interfaces, Usage of interfaces.

Packages: Defining a Package, Finding and Importing packages, Member Access.

Exception Handling: Fundamentals, Types, Multiple catch clauses, Nested try blocks, Throw Class, Using Finally and Throws, Built-in exceptions, User-defined exceptions.

UNIT III: MULTI-THREADING, STREAMS AND FILES

Multi-threading: Thread Class, Runnable interface, creating multiple threads, life cycle of thread, thread priorities, synchronization, thread communication, suspending, resuming and stopping threads.

Streams and Files: Stream, creating a File using File Output Stream, Reading Data from a File using File Input Stream, Creating a File using File Writer, Reading a file using File Reader, Serialization of Objects, File Copy, File Class, Zip File Class

UNIT IV: COLLECTION FRAMEWORK, NETWORKING

Collection Framework: Collection Objects, Retrieving Elements from Collections, Collection Interfaces, Hash Set Class, Linked Hash Set, Stack, Linked List, Array List, Vector, Hash Map, Arrays, Using Comparator to sort an Array, String Tokenizer, Calendar, Date.

Networking: Basics, Networking classes and Interfaces.

UNIT V: SWINGS AND EVENT HANDLING

Swings– Introduction, limitations of AWT, Components, Containers, Swing Components- JLabel and ImageIcon, JTextField, Swing Buttons, JTabbedPane, JScrollPane, JList, JComboBox, JMenu, JTable, JTree

Event Handling: Events, Event sources, Event classes, Event Listeners, Delegation event model, Handling mouse and keyboard events, Adapter classes.

Course Outcomes:

After completion of the above courses the student will be able to:

1. Solve real world problems using OOP techniques.
2. Implement Interfaces and Packages in Java
3. Work with Collection Framework Classes and Socket Classes
4. Develop multithreaded applications with synchronization.
5. Design GUI based applications

Text Books:

6. , “Java Complete Reference”, Robert Schildt Mc Graw Hill, 9th, 2016
7. Java for Programmers, P.J.Deitel and P.J.Deitel, Pearson education
8. Java: How to Program ,P.J.Deitel and P.J.Deitel, 8th edition, PHI.

Reference Books:

1. K. Somasundaram, “Programming in Java 2”, Jaico Publications, 2005.
2. Programming with Java” T.V.Suresh Kumar, B.Eswara Reddy, P.Raghavan Pearson Edition.

Mode of Evaluation: Assignments, Internal Mid Examination, External End Examination

MCA II Year I Semester

18MCAP108 FULL STACK WEB DEVELOPMENT

L T P C

4 0 0 4

Course Prerequisite: HTML, CSS and Fundamentals of Database

Course Description:

Basics of web design concepts- objects, classes, polymorphism, inheritance, static and dynamic binding. Object Oriented Programming using Java-classes, SQL and Front end and back end development

Course Objectives:

Student will be able to

1. Understand the concepts of Web design process, Visual Design Elements and Agile Web Development
2. Know the designing concepts using HTML5 and CSS3
3. Understand the concepts on Bootstrap frameworks and Javascripts for web designing
4. Know the concepts on databases, SQL queries and the environment of NoSQL and its queries
5. Understand the methodologies of authentication and know the basic idea on NodeJS environment and NPM

UNIT – I WEB DESIGN BASICS

The Design Process- Introduction to Product Design- Introduction to User Experience (UX)- Elements of Visual Design- Building and Using Mockups-Agile Web Development- Pseudo coding- SOLID Design Principles

UNIT – II MARKUP AND STYLING – HTML AND CSS

HTML Basics HTML Elements --HTML Attributes - HTML Formatting - HTML Links and Images HTML Lists, Blocks, Classes - HTML Layout and Forms. CSS: Introduction CSS3

•CSS3 Colours - Backgrounds, Borders, Padding, Height/Width - CSS3 Gradients, Shadows - CSS3 Text, Fonts - CSS3 2D & 3D Transforms - CSS Links – CSS Lists & Tables - C- S Box Model, Outline, Display, Max-width, Position - CSS Float, Inline-block – CSS Align – CSS Pseudo-class, Pseudo-element - CSS Navigation, Dropdowns, Tooltips, Images – CSS Selectors - CSS Forms, Buttons - CSS3 Multiple Columns - CSS3 User Interface: Box Sizing, Filters, Responsive CSS

UNIT III: BOOTSTRAP AND JAVASCRIPT

Introduction to Bootstrap - Bootstrap Basics - Bootstrap Grids - Bootstrap Themes - Bootstrap with CSS Introduction to JavaScript - Java Script Language Basics -Scope - JavaScript Events - Strings JavaScript Math - Arrays – Boolean – Comparisons - JavaScript Loops & Decisions - JavaScript Objects and Method - JavaScript Errors - Debugging - JavaScript Functions - JavaScript Forms - JavaScript DOM

UNIT - IV DATA MODELING AND SQL

Data Relationships-Designing a Data Model-Relational Databases- Alternative Databases- Data Normalization- Entity Relationship Modeling (ERM) SQL- Working with Database Schemas-Create-Read-Update-Destroy (CRUD) -Joins- Aggregate Functions and Groups- Sub Queries-NoSQL- Serialization -Modeling NoSQL data-Document Databases (PostGrey)

UNIT - V

NODE JS , AUTHENTICATION & SECURITY

NodeJS- Server-Side JavaScript- NPM- JavaScript Build Processes-Event Loop and Emitters-File System Interaction-Modules-Intro to Security and Authentication-Basic and Digest Authentication- Session-Based Authentication Document Databases

Course Outcomes:

After completion of the course the student will be able to :

1. Work with the concepts of Web design process, Visual Design Elements and Agile Web Development
2. Develop programs based on the concepts using HTML5 and CSS3
3. Identify and work with the concepts on Bootstrap frameworks and JavaScript's for web designing
4. Execute SQL queries and NoSQL queries
5. Classify the methodologies of authentication and work with the basic concepts on NodeJS environment and NPM

Text Books:

1. Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node 1st ed. Edition, Vasan Subramanian
2. Exploring Visual Design: The Elements and Principles, Joseph A.Gatto, Albert W. Porter, Teachers Edition, Davis publishers
3. HTML & CSS: The Complete Reference, Fifth Edition, Thomas Powell, McGrawHill
4. JavaScript Applications with Node Js, React, React Native and MongoDB, Eric Bush, Blue Sky Production Incorporated, 2018

Reference Books:

1. Complete Bootstrap: Responsive Web Development with Bootstrap 4, Matt Lambert, Bass Johnson, David Cochran, Ian Whitley, PACKT publishing
2. Getting Started with NoSQL, Gaurav Vaish, 2013, PACKT publishing
3. Beginning Node JS, Basarath Ali Syed, Apress, 2014
4. Secure Your Node Js Web Application: Karl Duanna, Pragmatic Bookshelf, 2016

Mode of Evaluation: Assignments, Internal Mid Examinations, External End Examination

MCA II Year I Semester

18MCAP109 INTRODUCTION TO MACHINE LEARNING

L	T	P	C
4	0	0	4

Course Prerequisite: Algorithm Design & Programming, Probability & Statistics

Course Description:

This course provides a concise introduction to the fundamental concepts in machine learning and popular machine learning algorithms. This will cover the standard and most popular supervised learning algorithms including linear regression, logistic regression, decision trees, k-nearest neighbour, an introduction to Bayesian learning and the naïve Bayes algorithm, support vector machines, kernels and neural networks.

Course Objectives:

Students will be able to:

1. Know about overview of machine learning and Bayesian learning concepts.
2. Learn about Decision tree learning and instance based learning.
3. Gain knowledge on Graphical models and Hidden Markov models.
4. Understand Clustering methods for real world problems
5. Learn neural networks for various inputs and expected outputs.

UNIT-I: INTRODUCTION AND BAYESIAN LEARNING

Well posed learning problems, designing a Learning system, Perspectives and issues in Machine Learning, Bayes theorem and Concept learning, Maximum likelihood and least squared Error hypotheses, Maximum likelihood hypotheses for predicting probabilities, Minimum description length principle, Naïve Baye's classifier, Bayesian belief Networks.

UNIT-II: REGRESSION & CLASSIFICATION

Linear Regression, Linear Discriminant Analysis - Logistic regression, Classification Methods- Nearest neighbor- Decision trees- Perceptron- Large margin classification- Kernel methods- Support Vector Machine.

UNIT-III: GRAPHICAL MODELS

Graphical models- Bayesian networks- Conditional independence Markov random fields- Inference in graphical models- Belief propagation- Markov models- Hidden Markov models

UNIT-IV: CLUSTERING

Clustering Methods-Partitioned based Clustering – K means, K medoids; Hierarchical Clustering – Agglomerative, Divisive, Distance measures; Density based Clustering, Gaussian model

UNIT-V: ARTIFICIAL NEURAL NETWORKS

Neural network representation, Appropriate problems in Multilayer perceptron, Multi-Layer Network-Back propagation, Remarks on Back Propagation algorithm, An Illustrative example: Face Recognition, Advance topics in Artificial Neural Networks

Course outcomes

After completion of the course the students will be able to:

1. Select real-world applications that needs machine learning based solutions.
2. Examine the Machine Learning Classification algorithms
3. Acquire knowledge on Graphical models for Machine Learning
4. Apply Clustering methods for real world problems
5. Implement neural networks for various inputs and expected outputs

Text Books :

1. Tom Michel , “Machine Learning”, Mc.Graw Hill, Indian Edition, 1997.
2. Introduction to Machine Learning Edition 2, by Ethem Alpaydin
3. C. Bishop, “Pattern Recognition and Machine Learning, Springer”, 2006.

Reference Book:

1. Mohri, Rostamizadeh and Talwalkar, “Foundations of Machine Learning”, MIT Press, 2012.

Mode of Evaluation: Assignments, Internal Mid Examination, External End Examination

18MCAP110

COMPUTER NETWORKS

L T P C
4 0 0 4

Course Prerequisite: Computer Organization

Course Description:

This course introduces the topologies, models, architecture, various components and protocols of computer networks. It narrates the analog and digital transmission related issues in various networking environments. It explains about the domain naming system and network management protocols in detail. It provides basic information about network security, mechanisms and services provided by various layers.

Course Objectives:

Students will be able to:

1. Get familiar with basics of networking concepts and functions of various layers in networking architecture
2. Know the basics of transmission techniques and media used in networking environment.
3. Understand the underlying components used in networking environments and the different versions of Internet Protocol.
4. Get familiar about DNS and the network management protocols in detail
5. Gain the knowledge about network vulnerabilities and security management.

UNIT I: INTRODUCTION

Uses of Networks, Categories of Networks -Communication model, Data transmission concepts and terminology, Protocol Architecture – Protocols, OSI, TCP/IP, LAN Topology, Transmission media.

UNIT II - DATA LINK LAYER

The Data Link Layer: Design Issues –Error Detection and Correction – Elementary Data Link Protocol – Sliding Window Protocol – HDLC. Medium Access Control Sub Layer: Multiple Access Protocol: CSMA Protocol – Collision Free Protocol – Data Link Layer Switching-Repeaters, Hub, Bridges, Switches, Router and Gateways - Bluetooth.**UNIT III - NETWORK LAYER**

Network layer – Switching concepts – Circuit switching – Packet switching –IP — Datagrams — IP addresses-IPV6– ICMP – Routing Protocols – Distance Vector – Link State- BGP.

UNIT IV - APPLICATION & TRANSPORT LAYER

Transport layer –service –Connection establishment – Flow control – Transmission Control Protocol – Congestion control and avoidance – User datagram protocol. -Transport for Real Time Applications (RTP).

Applications - DNS- SMTP–FTP- WWW–SNMP

UNIT V - SECURITY

Introduction – Substitution and Transposition Cipher–Symmetric Key and Asymmetric Key Cryptography – AES, DES, IDEA, RSA, SHA, Types of Attacks Security Services- Firewall and its types.

Course Outcomes:

After completion of the course the student will be able to:

1. Acquire the basic networking concepts and importance of layered architecture
2. Implement the various techniques and media used for data transmission process
3. Examine the functionalities needed for data communication into layers
4. Utilize the required functionality at each layer for given application
5. Implement the network management protocols, fundamentals of security issues and services available.

Text Books:

1. Andrew S. Tanenbaum, “Computer Networks”, 4th Edition, Eastern Economy Edition.
2. Frozen, “Data Communication and Networking”, Fifth Edition, TMH 2012.
3. William Stallings, “Data and Computer Communications”, Tenth Edition, Pearson Education, 2013

References:

1. Achyut S God bole, AtulHahate, “Data Communications and Networks”, Second edition 2011
2. Douglas E. Comer, —Internetworking with TCP/IP (Volume I) Principles, Protocols and Architecture, Sixth Edition, Pearson Education, 2013.
3. James F. Kurose, Keith W. Ross, “Computer Networking: A Top-down Approach, Pearson Education, Limited, sixth edition,2012.
4. John Cowley, “Communications and Networking : An Introduction”, Springer Indian Reprint, 2010.
5. Larry L. Peterson & Bruce S. Davie, “Computer Networks – A systems Approach”, Fifth Edition, Morgan Kaufmann, 2012
6. Wayne Tomasi, “Introduction to Data communications and Networking”, Pearson 2011

Mode of Evaluation: Assignments, Internal Mid Examination, External End Examination

18MCAP207 OBJECT ORIENTED PROGRAMMING LABORATORY

L T P C
0 0 3 1.5

Course Prerequisite: Basic Programming

Course Description:

In this course, students gain extensive hands-on experience on Java programming. Students learn to create robust console applications using code reusability with multi-threading, applications of exception handling, I/O streams and GUI implementation.

Course objectives:

Students will be able to

4. Write Java programs using classes and objects.
5. Build robust applications using object-oriented features
6. Develop Platform Independent GUIs

List of Experiments:

1.
 - a. Fibonacci sequence is defined by the following rule: The first two values in the sequence are 1 and 1. Every subsequent value is the sum of the two values preceding it. Write a Java program that print the nth value in the Fibonacci sequence.
 - b. Write a Java program that find prime numbers between 1 to n.
2. Write a Java program to perform String Operations.
3.
 - a. Write a Java program for sorting a given list of names in ascending order
 - b. Write a java program to perform linear search.
4. Write a Java program to process the employee retirement schemes. Employee wants to retire from organization. Organization has to give money to employee and employee will deposit that in a Bank. Bank will pay monthly interest if sufficient amount is available in the organization then it will be given to the employee otherwise employee has to wait until the sufficient amount is generator. Use looping concepts to process the above scenario.
5. Write a Java program that uses functions to perform the following operations:
 - a. Inserting a sub-string in to the given main string from a given position.
 - b. Deleting n characters from a given position in a given string.
 - i. Write a Java program that checks whether a given string is palindrome or not.

Ex: MADAM is a palindrome

6. Write a case Study: we will be creating two classes. They are Employee and EmployeeTest. EmployeeTest is the main class which keeps program entry i.e., main () method. First open notepad and the following code. Remember this is the Employee class and the class is a public class. Now save this Employee.java. The Employee class has 4 instance variables name, age, designation and salary. The class has explicitly defined constructor, which takes a parameter.
7. Write a Java Program demonstrates object construction
 - a. Overloaded Constructors
 - b. A call to another constructor with this(...)
 - c. A default constructor
 - d. An object initialization block
 - e. A static initialization block
 - f. An instance field initialization
8. Write a java program to demonstrates parameter passing in Java
9. Write a Java program to Perform Matrix Operations Addition, Subtraction, and Multiplications.
10. Write a Java program to demonstrate Interfaces.
11. Write a java program to demonstrate Packages.
12. 12.
 - a. Write a java program to split a given text file into n parts. Name each part as the name of the original file followed by .part<n> where n is the sequence number of the part file
 - b. Write a java program to convert an ArrayList to an Array.
 - c. Write a Java program to make frequency count of vowels, consonants, special symbols, digits, words in a given text.
13. 13.
 - a. Write a Java program that reads a file name from the user, then displays information about whether the file exists, whether the file is readable, whether the file is writable, the type of file and the length of the file in bytes.
 - b. Write a Java program that reads a file and displays the file on the screen, with a line number before each line.
 - c. Write a Java program that displays the number of characters, lines and words in a text file.
14. 14.
 - a. Write a java program to make rolling a pair of dice 10,000 times and counts the number of

times doubles of are rolled for each different pair of doubles. Hint: Math.random()

b. Write java program that inputs 5 numbers, each between 10 and 100 inclusive. As each number is read display it only if it is not a duplicate of any number already read display the complete set of unique values input after the user enters each new value.

c. Write a java program to read the time intervals (HH:MM) and to compare system time if the system time between your time intervals print correct time and exit else try again to repute the same thing. By using String Tokenizer class.

15

a. Write java program to create a super class called Figure that receives the dimensions of two dimensional objects. It also defines a method called area that computes the area of an object. The program derives two subclasses from Figure. The first is Rectangle and second is Triangle. Each of the sub class overridden area() so that it returns the area of a rectangle and a triangle respectively.

b. Write a Java program that creates three threads. First thread displays —Good Morning every one second, the second thread displays —Hello every two seconds and the third thread displays —Welcome every three seconds

16.

a. Write a Java program that correctly implements producer consumer problem using the concept of inter thread communication.

b. Use inheritance to create an exception super class called EexceptionA and exception sub class ExceptionB and ExceptionC, where ExceptionB inherits from ExceptionA and ExceptionC inherits from ExceptionB. Write a java program to demonstrate that the catch block for type ExceptionA catches exception of type ExceptionB and ExceptionC

17. Write a program that creates a user interface to perform integer divisions. The user enters two numbers in the JtextField, Num1 and Num2. The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a NumberFormatException. If Num2 were Zero, the program would throw an ArithmeticException Display the exception in a message dialog box.

18. Write a Java program that implements a simple client/server application. The client sends data to a server. The server receives the data, uses it to produce a result, and then sends the result back to the client. The client displays the result on the console. For ex: The data sent from the client is the radius of a circle, and the result produced by the server is the area of the circle. (Use java.net)

Course outcomes:

After the completion of the course, Students will be able to

1. Develop Java programs using Classes and Objects
2. Identify the I/O interfaces, primary and secondary storage devices

3.Execute Code for various functionalities of GUI

References:

1. “Programming with Java” T.V.Suresh Kumar, B.Eswara Reddy, P.Raghavan Pearson Edition.
2. “Java Fundamentals - A Comprehensive Introduction”, Herbert Schildt and Dale Skrien, Special Indian Edition, McGrawHill, 2013.

Mode of Evaluation: Continuous Cumulative evaluation of the lab experiments, Record, Viva voce and External Lab Examination.

MCA II Year I Semester

18MCAP208

FULL STACK WEB DEVELOPMENT LABORATORY

L	T	P	C
0	0	3	1.5

Course prerequisite:

Basic knowledge on Java, Java Script and DBMS

Course Description:

Full Stack Web Development course will make students to become master in front-end technology. It provides basic information and experiments to grow to be a Full-Stack web developer. With fast growing technologies, the students can update their knowledge on technologies. This will help the students to learn the complete set of process like designing, development and deployment.

Course objectives:

Students will be able to:

1. Acquire knowledge on web designing using front end tools
2. Develop coding using scripting languages
3. Develop applications using AJAX

List of Experiments:

1. Write a HTML Program to create one Website using all HTML tags.
2. Write a Java Script program using Arithmetic Operators
3. Write a Java Script Program to Sort an Array of Strings
4. Write a Java Script Program
5. To print the array values using for loop
6. To find Fibonacci Series less than of a given number using while loop
7. To print the Even Numbers less than of a given number using do-while loop
8. Write a Java Script Program
9. To check the person is eligible for vote or not using if statement
10. To print the message Good Morning if hours is less than 12 otherwise it prints Good Afternoon using if-else
11. To perform relational operators using else-if
12. Write a Java Script Program
13. To count the Vowels of the given string using functions

14. To find the arithmetic operations using functions
15. Write a Java Script Program to create JS Form Validation Page
16. Write a Java Script Program using DOM Events
17. Create a multiplication table asking the user the number of rows and columns he wants.
18. Create a form that collects the first name, last name, email, user id, password and confirm password from the user. All the inputs are mandatory and email address entered should be in correct format. Also, the values entered in the password and confirm password textboxes should be the same. After validating using JavaScript, display proper error messages in red color just next to the textbox where there is an error.
19. Write a Java Script Program to compute the average and grade of the students
20. Write a Java Script Program
21. To get the name of the Day
22. To get the name of the Month
23. Write an XML file which will display the book information. Validate it using DTD.
24. Implement simple AJAX Application
 - a. Create a simple XMLHttpRequest and retrieve data from a text file
25. Implement AJAX application with a callback function

Course Outcomes:

After the completion of the course, Students will be able to

1. Implement webpages using HTML and CSS
2. Execute dynamic calculation programs using java script
3. Apply the validations for webpage

Reference Book:

1. Java Script for Programmers Paul J. Deitel, Deitel& Associates, Inc. Harvey M. Deitel, Deitel& Associates, Inc.

Mode of Evaluation: Continuous Cumulative evaluation of the lab experiments, Record, Viva voce and External Lab Examination

18MCAP209 MACHINE LEARNING LABORATORY

L T P C
0 0 3 1.5

Course Prerequisite: Programming Basics

Course Description: Machine Learning practical covers how to implement regression, classification, clustering and Neural Networks concepts for real world applications.

Course Objectives:

1. Learn Machine learning classification algorithms and Models
2. Understand clustering methods in different data for real world problems
3. Learn neural networks for various inputs and avail expected outputs in machine learning environment.

List of Experiments:

Implement all the following using Python.

1. Implement simple linear regression to predict profits for a food truck based on the population of the city that the truck would be placed in.
2. We are given both the size of the house in square feet, and the number of bedrooms in the house. Implement multiple linear regression to predict that a house will sell for.
3. Build a classification model that estimates the probability of admission based on the exam scores using logistic regression.
4. Implement un-regularized and regularized versions of the neural network cost function and compute gradients via the backpropagation algorithm.
5. Implement K-means clustering.
6. Take a set of email data and build a classifier on the processed emails using a SVM to determine if they are spam or not.
7. Implement an anomaly detection algorithm using a Gaussian model and apply it to detect failing servers on a network.

Course Outcomes:

After successful completion of this lab, students can able to

1. Utilize the classification algorithms and Models for the given applications.
2. Check clustering methods through real world data
3. Apply Neural networks for known inputs and get unknown outputs

Text Books

1. Programming collective Intelligence: Building Smart Web 2.0 Applications-Toby Segaran
2. Building Machine Learning systems with Python, Willi Richart Luis Pedro Coelho
3. Python Machine Learning by Example, Liu, Yuxi(Hayden), Packt Publishers

Reference Books

1. Machine Learning with Python, Abhishek Vijayvargia, BPB publications
2. Python Machine Learning, Sebastian Raschka, packt publishers

Mode of Evaluation: Continuous Cumulative evaluation of the lab experiments, Record, Viva voce and External Lab Examination.

MCA II Year I Semester

18ENG905

CORPORATE COMMUNICATION SKILLS

L T P C

2 0 0 0

Course Description:

The introduction of the Advance Communication Skills course is considered essential for third year MCA students. At this stage, the students need to prepare themselves for their career which may require them to listen to, read, speak, and write in English both for their professional as well as interpersonal communication in the globalized context.

The proposed course should be a laboratory course to enable students to use good English and perform the following:

- Gathering ideas and information to organize ideas relevantly and coherently.
- Engaging in debates.
- Participating in group discussions.
- Facing interviews.
- Writing project/research reports/ technical reports.
- Making oral presentations.
- Writing formal letters.
- Transferring information from non-verbal to verbal texts and vice-versa.
- Taking part in social and professional communication.
- Attending the highest levels of performance in accordance with the ethical requirement.

Course Objectives:

The lab focuses on using multi-media instructions for language development to meet the following targets:

- To improve the students' fluency in English, through a well-developed vocabulary and enable them to listen to English spoken at normal conversational speed by educated

English speakers and respond appropriately in different socio-cultural and professional contexts.

- Further, they would be required to communicate their ideas relevantly and coherently in writing.
- To prepare all the students for their placements.

SYLLABUS CONTENT

UNIT I: COMMUNICATIVE COMPETENCY

- ❖ Reading comprehension
- ❖ Listening comprehension
- ❖ Vocabulary for competitive purpose
- ❖ Spotting errors

UNIT II: TECHNICAL WRITING

- ❖ Report writing
- ❖ Curriculum vitae
- ❖ E-mail writing
- ❖ Abstract and synopsis writing
- ❖ Reviewing

UNIT III: PRESENTATION SKILLS

- ❖ Oral presentation skills
- ❖ PowerPoint presentation
- ❖ Poster presentation
- ❖ Stage dynamics
- ❖ Body language

UNIT IV: CORPORATE SKILLS

- ❖ Telephonic skills
- ❖ Net etiquettes
- ❖ SMART goal setting
- ❖ Time management
- ❖ Problem solving skills

UNIT V: PROFESSIONAL ETHICS

- ❖ Credibility
- ❖ Professionalism
- ❖ Moral leadership
- ❖ Corporate social responsibility

COURSE OUTCOMES:

After completion of the above course the student will be able to:

1. Accomplish sound knowledge of vocabulary and its usage.
2. Improve their overall language proficiency in English.
3. Enhance their understanding of professional ethics.

SUGGESTED SOFTWARE:

- K – VAN SOLUTIONS
- DELTA'S key to the next generation TOEFL tests: Advance skills practice.
- TOEFL and GRE (KAPLAN, AARCO and BARRONS, USA, Cracking GRE by CLIFFS)
- Train2success.com

Text Books:

1. Technical Communication by Meenakshi Raman and Sangeetasharma, Oxford Univ. Press.
2. Effective Technical Communication, M Ashraf Rizvi, Tata Mc. Graw-Hill Pub, company Ltd.
3. High School English Grammar & Composition by WREN & MARTIN, S.Chand & Company Pvt. Ltd.

References:

1. Communication Skills, Sanjay Kumar & PushpLata, Oxford Univ.Press.
2. Basic Communication Skills for Technology, Andrea J. Rutherford: Pearson Education Asia, New Delhi.
3. Communication Skills, LenneSen, Prentice –Hall of India Pvt. Ltd., New Delhi.
4. Communicating at work, Ronald B. Adler, Seanne Marquardt Elmhurst, McGraw

Mode of Evaluation: Assignments, Internal Mid Examination

L	T	P	C
4	0	0	4

Course Prerequisite: Database Concepts

Course Description:

This course introduces data warehousing concepts and various data mining techniques. Data Mining is the process of extracting the hidden predictive information from large data sets. Data warehousing involves data pre-processing, data integration and providing on-line analytical processing (OLAP) tools for the interactive analysis of multidimensional data. It also includes the various processes such as spatial mining, text mining and web mining.

Course Objectives:

Students will be able to:

1. Know the fundamentals of Data mining and Data pre-processing techniques.
2. Understand the data warehouse architecture & OLAP and discover the frequent Item sets for the given data set.
3. Predict the knowledge from data sets using Classification techniques.
4. To understand the various clustering methods.
5. Learn the mining concepts for complex data objects.

UNIT- I: INTRODUCTION

Fundamentals of data mining, Data Mining Functionalities, Classification of Data Mining systems, Data Mining Task Primitives, Integration of a Data Mining System with a Database or a Data Warehouse System, Major issues in Data Mining

Data Pre-processing: Need for Pre-processing the Data, Data Cleaning, Data Integration and Transformation, Data Reduction, Discretization and Concept Hierarchy Generation.

UNIT- II: DATA WAREHOUSE AND OLAP TECHNOLOGY FOR DATA MINING

Data Warehouse, Multidimensional Data Model, Data Warehouse Architecture, Data Warehouse Implementation, Further Development of Data Cube Technology, From Data Warehousing to Data Mining

Mining Frequent Patterns, Associations and Correlations: Basic Concepts, Efficient and Scalable Frequent Item set Mining Methods, mining various kinds of Association Rules, From Association Mining to Correlation Analysis, Constraint-Based Association Mining

UNIT- III: CLASSIFICATION AND PREDICTION

Issues Regarding Classification and Prediction, Classification by Decision Tree Induction, Bayesian Classification, Rule-Based Classification, Classification by Back propagation, Other Classification Methods, Prediction, Accuracy and Error measures, Evaluating the accuracy of a Classifier or a Predictor.

UNIT- IV: CLUSTER ANALYSIS

Types of Data in Cluster Analysis, A Categorization of Major Clustering Methods, Partitioning Methods, Hierarchical Methods, Density-Based Methods, Grid-Based Methods, Model-Based Clustering Methods, Clustering High-Dimensional Data, Constraint-Based Cluster Analysis, Outlier Analysis.

UNIT- V: MINING OBJECT, SPATIAL, MULTIMEDIA, TEXT, WEB AND TIME SERIES DATA

Multidimensional Analysis and Descriptive Mining of Complex Data Objects, Spatial Data Mining, Multimedia Data Mining, Text Mining, Mining the World Wide Web, Mining Time Series data.

Course outcomes:

After the completion of the course, Students will be able to

1. Explore the knowledge about fundamentals of Data Mining and implement the methods of data pre-processing.
2. Generate the frequent item sets for the given data set using association rule mining.
3. Extract the knowledge by classification techniques using data mining tool.
4. Perform cluster analysis for the given data sets.
5. Apply the data mining algorithms to advanced database applications.

Text Book:

1.Data Mining – Concepts and Techniques - Jiawei Han &MichelineKamber, Morgan Kaufmann Publishers, Third edition,2014.

Reference Books:

1. Data Mining Introductory and advanced topics –Margaret H Dunham, Pearson education
2. Data Mining Techniques – Arun K Pujari,2nd edition, Universities Press.
3. Introduction to Data Mining – Pang-Ning Tan, Michael Steinbach and Vipin Kumar, Pearson education.
4. Insight into Data Mining, K.P.Soman,S.Diwakar, V.Ajay,PHI,2008.
5. Data Mining,V.Pudi and P.Radha Krishna, Oxford University Press.

Mode of Evaluation: Assignments, Internal Mid Examination, External End Examination

MCA II Year II Semester

18MCAP112

DATA STRUCTURES AND ALGORITHMS

L	T	P	C
4	0	0	4

Course Prerequisite: Any Programming language

Course Description:

The purpose of this course is to impart knowledge on various data structures and design and analyse the algorithm concepts

Course Objectives:

Students will be able to

1. Introduces the notations for analysis of the performance of algorithms and learn basic data structures stack, queue and linked list.
2. Understand Sorting and Searching techniques and divide and conquer strategy.
3. Learn the concepts of Trees and Graphs.
4. Know disjoint sets and greedy method for problem solving.
5. Introduces Dynamic programming and backtracking concepts for problems.

UNIT-I: INTRODUCTION, BASIC DATA STRUCTURES AND HASH TABLE REPRESENTATION

Introduction: Algorithm, Performance Analysis-Space complexity, Time complexity, Asymptotic Notations

Basic Data Structures: Stack, Queue, Applications of stacks- Infix to postfix conversion, Evaluation of postfix expression, linked list, doubly linked list, Circular list representation- operations - insertion, deletion and searching.

Hash Table representation: Hash functions, collision resolution-separate chaining, open addressing-linear probing, quadratic probing

UNIT-II: SEARCHING, SORTING & DIVIDE AND CONQUER STRATEGY

Searching: Linear Search and Binary Search

Sorting: Bubble sort, Selection sort, Insertion sort, Radix sort, Heap sort

Divide and conquer: General method, Applications- Quick sort, Merge sort

UNIT-III: TREES AND GRAPHS

Trees: Definition, Types of trees, Binary Trees, Binary Tree Traversal Methods, Binary search tree operations, operations of AVL tree, B-Tree

Graphs: Definition, Representation of graphs, Graph Traversal Methods.

UNIT-IV: DISJOINT SETS AND GREEDY METHOD

Disjoint Sets: Disjoint set operations, Union and Find algorithms

Greedy Method: General method, Applications- knapsack problem, Minimum cost spanning trees, Prim's algorithm, Kruskal's algorithm, Single source shortest path problem

UNIT-V: DYNAMIC PROGRAMMING AND BACKTRACKING

Dynamic Programming: General method, Applications- Optimal binary search trees, knapsack problem, all pair's shortest path problem, Travelling Sales Person Problem Backtracking: General method, Applications- N queen's problem, Graph colouring.

Course Outcomes:

After completion of the above course the student will be able to:

1. Analyse the performance of algorithms and Select the data structures that efficiently model the information in a problem
2. Implement various Sorting and Searching Techniques.
3. Apply Trees and Graph theory techniques for the problems.
4. Implement greedy technique for problems
5. Apply Dynamic programming and Backtracking problem solving strategies.

Text Books:

1. Fundamentals of Data structures in C, 2nd Edition, E. Horowitz, S. Sahni and Susan Anderson-Freed, Universities Press.
2. Fundamentals of Computer Algorithms, Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekharan, Galgotia publications.

Reference Books:

1. Data structures A Programming Approach with C, D. S. Kushwaha and A.K. Misra, PHI.
2. Data Structures using C, R. Thareja, Oxford University Press.
3. Data Structures, S. Lipschutz, Schaum's Outlines, TMH.
4. Data Structures and Algorithms ,Alfred Aho, John E. Hopcroft, Ullman, Addison_Wesley.
5. Data Structures using C & C++, R. Shukla, Wiley India.
6. Classic Data Structures, D. Samanta, 2nd edition, PHI.

Mode of Evaluation: Assignments, Internal Mid Examination, External End Examination

MCA II Year II Semester

18MCAP113

CLOUD COMPUTING

L T P C

4 0 0 4

Course Prerequisite: Computer Fundamentals, Computer Networks

Course Description

Cloud computing is a key distributed systems paradigm that has grown popular in the last few years. Cloud technologies are pervasive, touching our daily lives any time we access the World Wide Web, use a mobile app, or make a retail purchase.

Course Objectives

Students will be able to

1. Explain the evolving computer model called cloud computing.
2. Introduce the various levels of services that can be achieved by cloud.
3. Describe the different types of Migration services & Data Storage.
4. Explain the various types of Cloud Architectures & Applications.
5. Describe the cloud security and legal issues

UNIT- I SYSTEMS MODELING, CLUSTERING AND VIRTUALIZATION

Distributed System Models and Enabling Technologies. Computer Clusters for Scalable Parallel Computing. Virtual Machines and Virtualization of Clusters and Data centers.

UNIT-II FOUNDATIONS

Introduction to Cloud Computing, migrating into a Cloud, Enriching the 'Integration as a Service' Paradigm for the Cloud Era. The Enterprise Cloud Computing Paradigm.

UNIT- III IAAS, PAAS & SAAS

Virtual machines provisioning and Migration services, On the Management of Virtual machines for Cloud Infrastructures, Enhancing Cloud Computing Environments using a cluster as a Service.

UNIT- IV MONITORING, MANAGEMENT AND APPLICATIONS

Architecture for Federated Cloud Computing, SLA Management in Cloud Computing, Performance Production for HPC on Clouds, Best Practices in Architecture Cloud Applications in the AWS cloud.

UNIT – V GOVERNANCE AND CASE STUDIES

Organizational Readiness and Change management in the Cloud age. Data Security in the Cloud, Legal issues in Cloud computing. Achieving Production Readiness for Cloud Services

Text Books

1. Cloud Computing: Principles and Paradigms by Rajkumar.
2. Distributed and Cloud Computing. Kal Hwang. GeoffeyC.Fox. Jack J. Dongarra. Elsevier. 2012.

Reference Books

1. Cloud Computing: A Practical Approach. Anthony T.Velte. Toby J.VeFte, Robert Elsenpeter. Tata McGraw Hill. rp2011.
2. Enterprise Cloud Computing Gautam Shroif, Cambridge University Press. 2010.
3. Cloud Computing: Implementation, Management and Security, John W. Rittinouse, James F Ransome. CRC Press, rp2012.
4. Cloud Application Architectures: Building Applications and Infrastructure in the Cloud. George Reese, O'RedI SPD, rp2011.
5. Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance, Tim Mather, SubraKtriaraswamy, ShahedLatif, O'Redç SPD, rp2011.

Course Outcomes

After completion of the course the student will be able to

1. Ability to understand the system models, virtualization and cloud computing concepts.
2. Analyze the Cloud computing setup with its vulnerabilities and applications using different architectures.
3. Design different workflows according to cloud services requirements and apply real time applications.
4. Apply and design suitable Virtualization concept, Cloud Resource Management and design scheduling algorithms for suitable architecture in Cloud Applications.
5. Assess cloud Storage systems and to explore the impact of engineering on legal and societal issues involved in addressing the security issues of cloud computing.

Mode of Evaluation: Assignments, Internal Mid Examinations, External End Examination.

MCA II Year II Semester

18MCAP210 DATA WAREHOUSE AND DATA MINING LABORATORY

L	T	P	C
0	0	3	1.5

Course Prerequisite: Knowledge on Databases

Course Description:

This course introduces Data mining tool WEKA is used to apply machine learning algorithms.

Course Objectives:

Students will be able to:

1. Understand the concepts of mining rules from the given data.
2. Develop skills for using data mining tool to solve the practical problems.
3. Know the various techniques to extract the hidden data.

LIST OF EXPERIMENTS

1. Write the procedure to implement the data pre- processing using own dataset.
2. Implement association rule mining using your own data set.
3. Implement classification techniques.
4. Write the procedure for rep tree and implement rep tree using own dataset preparation in weka.
5. Implement J48 algorithm with own dataset.
6. Write the procedure for multi-layered perception and cross validation and test set.
7. Write the procedure for selecting attribute and prepare graphical representation using weka
8. Write the procedure for visualizing in weka and explain different constraint based visualization with graphs
9. Write the procedure for buys computer, peps dataset and association rules.
10. Implement k-means clustering algorithm with dataset.
11. Implement single link, complete link and average link in clustering method.

References:

1.Data Mining – Concepts and Techniques - Jiawei Han &Micheline Kamber, Morgan Kaufmann Publishers, Third edition,2014.

Mode of Evaluation: Continuous Cumulative evaluation of the lab experiments, Record, Viva voce and External Lab Examination

Course Prerequisite: Programming Basics

Course Description:

This course introduces on the practical part of Data Structures and Algorithms using C language. This course allows students to implement linear and nonlinear data structures. It also gives practical exposure on Sorting and searching algorithms.

Course Objectives:

Students will be able to:

1. Design, implement and document linear and Non-linear data structure programs.
2. Choose appropriate data structures for solving typical searching and sorting problems.
3. Learn Divide & Conquer, Greedy method and Backtracking strategies.

List of Experiments:

1. Create a Stack and do the following operations.
 - a. Push b. Pop c. Peep
2. Create a queue and do the following operations.
 - a. Add b. Delete c. Display
3. Implement the operations on singly linked list.
4. Implement the operations on circular linked list.
5. Implement the operations on doubly linked list.
6. Write a C program to convert from infix to postfix expression.
7. Write a C program to evaluate postfix expression.
8. Write C program that use both recursive and non-recursive functions to perform the following
 - a. Linear search b. Binary search
9. Implement Insertion and Selection sort.
10. Implement Bubble sort technique.
11. Implement Quick sort using Divide and Conquer strategy.
12. Implement Merge sort using Divide and Conquer strategy.
13. Write a C program for traversing tree in

- a. Pre order b. In order c. Post order
14. Implement the operations on Binary Search tree.
15. Implement knapsack problem using greedy method.
16. Calculate minimum spanning tree using Prim's method.
17. Calculate minimum spanning tree using Kruskal's method.
18. Implement single source shortest path problem.
19. Implement all pairs shortest path problem.
20. Implement N-queen's problem using backtracking.

Course Outcomes:

After completion of the course the student will be able to:

1. Select the data structures that efficiently model the information in a problem.
2. Assess efficiency trade-offs among different data structure implementations.
3. Implement and know the application of algorithms for Sorting and Searching.

Text Books:

1. C & Data Structures by V.V. Muniswamy. I K International Publishing House Pvt. Ltd
2. Data Structures through C in depth ,S.K. Srivastava, BPB publications
3. Data Structures and Algorithms made Easy by Narasimha Karumanchi.

Reference Books:

1. Data Structures, S. Lipschutz, Schaum's Outlines, TMH.
2. Data Structures and Algorithms ,Alfred Aho, John E. Hopcroft, Ullman, Addison_Wesley.
3. Data Structures using C & C++, R. Shukla, Wiley India.

Mode of Evaluation: Continuous Cumulative evaluation of the lab experiments, Record, Viva voce and External Lab Examination.

MCA II Year II Semester

18ENG906 PERSONALITY DEVELOPMENT AND SOFT SKILLS

L	T	P	C
2	0	0	0

Course Description:

This course intends and aims to enhance the confidence of the students by exposing them to various situations and contexts they face in their career. It is imperative for MCA students to start preparing for the ever growing competition in the Job market. This course focuses on the practical aspects of soft skills relevant to the requirements of the prospective employers in view of globalization.

Course Objectives:

1. To expose the students to those soft skills which are crucial to an employee's ability to work smarter.
2. To enhance Art of Communication, Team Skills, GD handling skills and preparing resume & Interview Skills

UNIT I: VERBAL COMMUNICATION

Active listening - Non Verbal Communication - Body Language.

UNIT II: DEVELOPING EMOTIONAL INTELLIGENCE

Importance of Team work - Leadership skills, self-realization (Identifying strengths and weaknesses).

UNIT III: TIME MANAGEMENT

GD skills – Roles in a GD – Do's & Don'ts – Mock GD.

UNIT IV: RESUME PREPARATION

Tips in writing resume - Interview Handling Skills – Do's & Don'ts – mock interview - Goal setting.

UNIT V:

Grooming etiquette, Professional Electronic Communication-Telephone etiquette, Email etiquette.

Course Outcomes:

After the completion of the course, Students will be able to

1. Communicate effectively and enhance their interpersonal relationship and building skills with renewed self-confidence.
2. Work together in teams and accomplish objectives in a cordial atmosphere.
3. Face Group Discussion with confidence
4. Prepare resume and face interviews.
5. Understand and develop the etiquette necessary to present oneself in a professional setting.

Text Books:

“Soft Skills”. Dr K Alex. S Chand Publications, New Delhi

References Books:

1. The Seven Habits Of Highly Effective People By Stephen R. Covey, Covey Leadership Center, 2005.
2. Negotiate To Close By Gary Karnass, Simon And Schuster, 1987.
3. The Greatest Miracle In The World – Ogmandino, Random House Publishing Group, 2009.
4. Working With Emotional Intelligence - Daniel Goleman, A&C Black, 2009.
5. Developing Communication Skills By Krishna Mohan And Meera Banerji; Macmillan India Ltd., Delhi, 2000.
6. Essentials Of Effective Communication, Ludlow And Panthon; Prentice Hall Of India, 1993.
7. Effective Presentation Skills (A Fifty-Minute Series Book) By Steve Mandel, Crisp Publications, 1996.
8. “Effective Group Discussion: Theory And Practice” By Gloria J. Galanes, Katherine Adams, John K. Brillhart, Tata Mcgraw-Hill, 2010.

Mode of Evaluation: Assignments, Internal Mid Examinations

AGILE SOFTWARE DEVELOPMENT PROCESS & DEVOPS

L T P C

4 0 0 4

Course Prerequisite: Software Engineering life cycle development

Course Description:

Object oriented analysis, UML, Agile software development and Frameworks of Agile development. DevOps is fundamental for developers and application oriented personalities. Improved workflows and faster deployment will be helpful for developers in their project implementation.

Course Objectives:

Students will be able to

1. Analyze the object oriented techniques used in the real-world software industries and acquire knowledge about the classes, objects and UML diagrams.
2. Compare agile with traditional software development and Contrast different agile methodologies.
3. Analyze the DevOps relationship to Agile, Lean and ITSM.
4. Improve workflows and speed up deployment.
5. Improve feedback and fine tune continual learning activities.

UNIT-I: OBJECT ORIENTED ANALYSIS AND UML

Introduction: System development life cycle & variations, Approaches to system development, SSAD Vs OOAD, Overview of OOAD methodologies, The Unified Process and its characteristics, Unified Process life cycle, What is Object Orientation? What is OO development? OO themes; Evidence for usefulness of OO development; OO modeling history.

Object oriented modeling: Capturing, modeling and documenting system requirements, Identifying object oriented basic components and their relationships-Dynamic modeling-Use case modeling-Various UML diagrams.

UML: Introduction to UML, UML Building Blocks. Conceptual model of UML, Architecture, Classes, Relationships, Common Mechanisms, Class-Object-Sequence-Activity-Use Case Diagrams.

UNIT-II: INTRODUCTION TO AGILE AND ITS SIGNIFICANCE

Introduction to Software Development Process. Iterative and Evolutionary Methods. What is agile software development? How agile methods are different than traditional methods and when to use agile methods. Agile Manifesto: Principles, and Benefits of agile development, User Stories, Generating User Stories. Agile Story: Evolutionary delivery, Scrum Demo, Planning game, Sprint back log, adaptive planning. Agile Motivation – Problems with the Waterfall - Research Evidence. Agile roles: how does agile redefine traditional software jobs and tasks, in particular the manager's role?

UNIT-III: AGILE FRAMEWORK

Scrum: Method Overview, Life cycle phases and Work product roles and practices. **Extreme Programming:** Method Overview, Life cycle phases and Work product roles and practices. **Unified process:** Method Overview, Life cycle phases and Work product roles and practices. **EVO:** Method Overview, Life cycle phases and Work product roles and practices.

UNIT-IV: INTRODUCTION OF DEVOPS

Continuous Delivery, and the three ways – The Principles of Flow – Feedback – Continual Learning and Experimentation. Selecting which value stream to start with – Understanding the Work in our value stream – Making it visible – Expanding - Design Organization and architecture-Outcomes by Integrating Operations.

UNIT-V: TECHNICAL PRACTICES OF FEEDBACK AND CONTINUAL LEARNING

Technical Practices of Feedback: Creating telemetry for solving problems –analyze Telemetry - Feedback for safety code deployment – integrate Hypothesis driven development and A/B testing – creating reviews to increase quality of work.

Technical Practices of Continual Learning: Enable and inject learning – local discoveries into global improvements – reserve time for organizational learning and improvement.

Course Outcomes:

After completing the course, the student is expected to be able to:

1. Design the UML diagrams for real-world project components.
2. Identify the significance of agile development over the traditional approaches.
3. To examine the different frameworks of agile software development approaches.
4. To analyze DevOps and its relationship with agile development.
5. To Recognize how to improve feedback and fine tune continual learning activities.

Text Books:

1. Object Oriented Analysis and Design with Applications by Grady Booch Robert, Third Edition.
2. Agile Software Development Best Practices for Large Software Development Projects by Thomas Stober, and Uwe Hansmann.

3. Shore, James. *The Art of Agile Development: Pragmatic guide to agile software development*. "O' Reilly Media, Inc.", 2007.
4. The DevOps Handbook by Gene Kim, Jez Humble, Patrick Debois and John Willis.

Reference Books:

1. Object - Oriented Modeling and Design with UML 2nd Edition.
2. Agile Software Development, Principles, Patterns, and Practices by Robert C. Martin
3. Effective DevOps, by Jennifer Davis, Ryn Daniels.
4. DevOps for Web Development by MiteshSoni.

Mode of Evaluation: Assignments, Internal Mid Examination, External End Examination.

Course Prerequisite: Software Engineering, OOAD, Use Case Models

Course Description:

The primary goal is to enhance the stream of value from idea to end user. Obviously, there's a traditional change that must happen for a company to be successful with DevOps, so culture is a big focus, but the DevOps goal is to make the delivery of project to be more efficient and effective. Also introduces new tools in the Industry for implementing project development activities.

Course Objectives:

Students will be able to:

1. Use communicative visual modelling language so that they can design various meaningful models.
2. To install and configure the latest DevOps tools.
3. To develop and implement the project management activities using DevOps tools.

List of Experiments:

1. To develop a problem statement.
2. Develop an IEEE standard SRS document, and also develop risk management and project plan (Gantt Chart).
3. Identify the use cases and develop the Use Case model.
4. Identify the business activities and develop an UML Activity diagram.
5. Identify the conceptual classes and develop a domain model with UML class diagram.
6. Using the identified scenarios, find the interaction between Objects and represents them using UML Interaction diagrams.
7. Draw the State Chart diagram.
8. Draw the Component and Deployment diagrams.

9. Install Git on Windows/Mac/Linux/Unix.
10. Understand Git file life cycle.
11. Create Git repository - Local repository and configure it to GitHub.
12. Create a repository on GitHub and clone it.
13. Perform Git operations on the remote repository
14. Performing different Branching and Merging operations
15. Using Git Tags
16. How to use Git stash
17. Installing the Jenkins setup.
18. Create a simple Jenkins job, delete, fail, disable the job.

Course Outcomes:

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

1. Design the various meaningful project models.
2. Install and configure the basic components of DevOps tools.
3. Get exposure of developing and implementing the project management activities.

Text Books:

5. “Hands-on DevOps”, by Sricharan Vadapalli, Kindle Edition
6. “Practical DevOps”, by Joakim Verona, Kindle Edition
7. “Mastering Jenkins”, by Jonathan McAllister, Kindle Edition
8. “GIT Essentials”, by Ferdinando Santacroce, Kindle Edition.

Reference Books:

1. “The DevOps Handbook”, by Gene Kim et.al, Kindle Edition.
2. “Jenkins Essentials”, by Mitesh Soni, Kindle 2nd Edition.
3. “GIT”, Learn Version Control with GIT, by Dennis Hutten, Kindle Edition

Mode of Evaluation: Continuous Cumulative evaluation of the lab experiments, Record, Viva voce and External Lab Examination.

Discipline Elective - I

MCA II Year II Semester

18MCAP401 MOBILE APPLICATION DEVELOPMENT USING ANDROID

L	T	P	C
3	0	0	3

Course Prerequisite: Java Programming and Basics of XML

Course Description:

This course is concerned with the development of applications on Android platform. Android is used as a basis for the development of mobile applications. This course starts with the basic concepts of Java, history of android and architecture. It introduces the major building blocks that are used to develop an android application with examples. It also covers the development of applications using widgets, events, networking. It provides ideas on sensors, their types and writing programs based on sensor classes for application development.

Course Objectives:

Students will be able to

1. Understand Android history and its fundamentals and know the building blocks of android
2. Get idea on the creation of android user interface and its testing mechanisms
3. Identify the usage of threads, broadcast receivers, intents, services and their working methodology
4. Know about the storage mechanism in android using SQLite and the usage of content providers
5. Recognize the usage of android widgets and sensors in android based applications

UNIT- I INTRODUCTION AND INSTALLATION OF ANDROID TOOLS

Android Overview – History – Android Versions - Android Flavors. **Android Stack:** Linux, Native Layer and Hardware Abstraction Layer (HAL) – ART - Application Framework: Native C++ Library – Applications: System and User Applications - **Installation and Use of Android Tools:** Installing the Android SDK - Anatomy of an Android Project - Drawable Resources – XML Introduction - Creating user interface using XML – Overview of Android Building Blocks – Logging Messages in Android

UNIT- II USER INTERACTION

Example. Input Components – Text View – Image View – List View and Alert Dialogues – Menus: Popup, Options and Context Menus – Screen Navigation through App Bar – RecyclerView – Material Design – Testing the User Interface: Espresso – Screen Navigation using Intents: Definition – Usage of Intents – Creation of Intents with example program – Lists and Adapters – Types of Adapters – Examples using Adapters

UNIT- III THREADS, LOADERS AND ASYNCTASK LOADER, BROADCAST RECEIVERS, SERVICES

Threading in Android – AsyncTask – Loaders – AsyncTask Loader – Connecting to Internet: JSON - HTTP API, Apache HTTP Client, HTTP URL Connection - Broadcast Receivers: Custom Broadcasts – Broadcasting Intents and their related API - Boot Receiver - Alarms and system services – Examples on alarms and services – Services: Services Life Cycle – Intent Service – Implementing Intent Service – Notifications: Managing Notifications

UNIT – IV SAVING, RETRIEVING AND LOADING DATA

Android File systems and Files - Action Bar: Preferences and Action Bar - Shared Preferences – App Settings - Databases on Android - SQLite - Status Contract Class, Update Refresh Service – Cursors – Backups - Content Providers: Overview – Role of Content Providers - Content Provider Example Program – Content Resolver

UNIT-V APPLICATIONS WIDGETS, INTERACTION AND SENSORS

App Widgets: Creation of Application Widgets - Interaction and Animation: Live Wallpaper and Handlers - Sensors: Sensor API in Android - Motion Sensor, Position Sensor, Environmental Sensor, Sensor Values, Sensor Manager Class, Sensor Class, Sensor Event class, Sensor Event Listener interface, Compass Accelerometer and orientation Sensors, Sensor Examples

Course Outcomes:

After completion of the above course the student will be able to

1. Work on android basic components and Install android
2. Create User Interfaces with various Layouts and views using android building blocks
3. Work with Broadcast Receivers and Services
4. Create Database in Android, Store and Retrieve data using SQLite and Content Providers
5. Develop widgets, Wall papers for an android application and write programs based on Sensors

Text Books:

1. Android Programming-The Big Nerd Ranch Guide, Bill Philips, Christ Stewart, Kristin Mariscano, Big Nerd Ranch publishers, 3rd Edition
2. Android Programming for Beginners, John Horton, PACKT publishers

Reference Books:

1. Learning Android , By Marko Gargenta& Masumi Nakamura, O'Reilly, II Edition
2. Android Application Development All in One for Dummies, Barry Burd, Wiley, 2nd Edition
3. Android application Development-Black Book, Pradeep Kothari, dreamtech
4. Android Programming - Unleashed, B.M.Harwani, Pearson Education, 2013
5. Head First Android Development: A Brain-Friendly Guide, Dawn Griffiths and David Griffiths, O'Reilly, 2nd Edition
6. Android System Programming, Roger Ye, PACKT publishers
7. Programming Android,By Zigurd Mednieks, Laird Dornin, G.Blake Meike & Masumi Nakamura, O'Reilly

Mode of Evaluation: Assignments, Internal Mid Examination, External End Examination

MCA II Year II Semester

18MCAP402 WEB PROGRAMMING THROUGH PHP

L	T	P	C
3	0	0	3

Course Prerequisite: Basic HTML and Advanced HTML Tags, CSS Concepts, OOPs concepts, Basics of SQL

Course Description:

This course enables students to have strong fundamentals in web programming using PHP, This course helps students to understand various concepts of web page designing, and to establish connection between the front end PHP forms with the backend database. This course helps the students to develop dynamic websites for different applications

Course Objectives:

Students will be able to

1. Understand HTML Tags – Both basics and Advanced, PHP basics, Operators, Control structures and arrays
2. Learn String manipulation, Function concepts and Regular Expressions
3. Know Object oriented concepts and their implementation PHP.
4. Learn Database concepts, and analyze different types of Queries in MySQL
5. Know advanced PHP concepts, to develop application using PHP

UNIT – I INTRODUCTION TO HTML TAGS, PHP BASICS AND ARRAYS

Basic Tags in HTML, Advanced HTML Tags, CSS Concepts, Linking CSS and HTML.

PHP Introduction: Evolution of PHP, Static Vs Dynamic Web pages, PHP servers for different Operating Systems, PHP Data types, Variables in PHP, User defined variable declaration, Scope of variables, Operators and their precedence in PHP, Conditional and Looping/Iterative constructs in PHP. **Arrays:** Types, Array Operators, Array Manipulation, Useful Library functions in Arrays

UNIT - II STRINGS, REGULAR EXPRESSIONS & FUNCTIONS

Strings: Creating a string and accessing, Formatting Strings, Searching and Replacing String, String Functions. **Regular Expressions:** Basics, Types, Pattern matching, Replacing text, Splitting strings using Regular Expressions. **Functions:** Built in functions, User defined functions, Returning values from function, Call by Value and Call by Reference, Recursion)

UNIT– III OBJECT ORIENTED CONCEPTS AND EXCEPTION HANDLING

Basic concepts in OOPs, Object Oriented PHP, Creation of Classes and Operations in PHP using OOPs, Implementing different types of Inheritance using PHP, Advanced object oriented functionality. **Exception Handling:** Exceptions Vs Errors, Implementing Try, Catch and Throw in PHP.

UNIT – IV DATABASE AND MYSQL

RDBMS concepts, Database Architecture and Web Database Architecture, **MySQL:** Creating tables, Usage of Primary keys and Foreign Keys, Inserting data into tables, Data manipulation with MySQL databases using join queries and sub queries.

UNIT - V ADVANCED PHP

PHP Forms and validation checking, **PHP Sessions:** Starting PHP Session, Session variables, Modifying Session variables, Destroying Session. PHP Cookies, Connecting PHP with MySQL, Authentication using PHP and MySQL. Secure data Transaction using PHP and MySQL. Developing applications in PHP by connecting with MySQL.

Course Outcomes:

After completion of the above course the student will be able to:

1. Work with basic and advanced tags in HTML & CSS and basic programs in PHP using arrays and embedding PHP tags with HTML tags.
2. Write programs in PHP using Functions, strings and regular expressions.
3. Apply object oriented approach in PHP programming
4. Develop different MySQL Queries for data manipulation and mechanism of RDBMS to connect with PHP.
5. Develop a complete end-to-end web application using PHP

Text Books:

1. The Complete Reference – PHP, Steven Holzner, McGrawHill, 2007
2. Programming PHP - Creating dynamic web pages, RasmusLerdorf, KevinTatroe and Peter B. MacIntyre, O'Reilly, 2006

Reference Books:

1. The Joy of PHP, Alan Forbes, Copyright Material, 2017
2. Programming PHP MySQL- Let's Explore, HirdeshBharadwaj, Educreation Publishing, 2nd Edition
3. PHP and MySQL web development, Luke Weiling& Laura Thomson, Addison Wesley, 5th Edition

4. Head First PHP & MySQL, Lynn Beighley, Michael Morrison, "O'Reilly Media, Inc.", 2008
PHP /MySQL Programming for the Absolute Beginner, Andy Harris, Cengage Learning, 2009

Mode of Evaluation: Assignments, Internal Mid Examination, External End Examination

MCA II Year II Semester

18MCAP403 WEB TECHNOLOGIES THROUGH JAVA

L	T	P	C
3	0	0	3

Course Prerequisite: Basic Knowledge on HTML & CSS

Course Description:

This course enables students to have strong fundamentals in web programming using HTML, CSS, Java Script and JSP. This course helps students to understand various concepts of web page designing, and to establish connection between the Client and Server with the backend database. This course helps the students to develop dynamic websites for different applications.

Course Objectives:

Students will be able to

1. Know the basic concepts of web and java, HTML, CSS and Java Script
2. Understand DHTML and XML concepts and creation of XML file
3. Get idea on Java Beans and their implementation in web
4. Learn about Java Server Pages and their implementation
5. Understand the database connectivity in Java using JDBC and its implementation

UNIT I - WEB INTRODUCTION, HTML, and CSS & JAVASCRIPT

Introduction, World Wide Web, Web Browsers, Web Page, Java and its Web Applications
– an Overview, Search Engines HTML: Basic and Advanced Tags. CSS, Java Script:
Introduction to Java Script – Objects in Java Script – Dynamic HTML with Java Script

UNIT II - EXTENDED MARKUP LANGUAGE

Introduction to Dynamic HTML – Introduction to XML – Document type definition – XML
Schemas - Document Object Model – Presenting XML – XML processors

UNIT III - JAVA BEANS

Introduction to Java Beans, Advantages of Java Beans, JDK Introspection, Using Bound
properties, Bean Info Interface, Constrained properties, Persistence, Customizes, Java Beans API

UNIT IV - JSP APPLICATION DEVELOPMENT

Generating Dynamic Content, Using Scripting Elements Implicit JSP Objects, Conditional
Processing – Displaying Values Using an Expression to Set an Attribute, Declaring Variables and
Methods Error Handling and Debugging Sharing Data Between JSP pages, Requests, and Users

Passing Control and Data between Pages – Sharing Session and Application Data – Memory Usage Considerations

UNIT V - DATABASE ACCESS

Database Programming using JDBC, Studying Javax.sql.* package, Accessing a Database from a JSP Page, Application – Specific Database Actions, Deploying JAVA Beans in a JSP Page, Introduction to struts

Course Outcomes:

After Completion of the Course the students will be able to

1. Write programs using HTML, CSS and Javascript
2. Work on DHTML concepts and create XML files for required applications
3. Execute programs based on Java Beans
4. Implement Java Server Page concepts by writing relevant programs
5. Develop a web application using JDBC connectivity and other concepts

Text Books:

1. Web Programming, building internet applications, Chris Bates 2nd edition, WILEY Dreamtech
2. The complete Reference Java 2 Fifth Edition, Patrick Naughton and Herbert Schildt. TMH
3. Java Server Pages –Hans Bergsten, SPD O'Reilly

Reference Books:

1. Programming world wide web-Sebesta, Pearson Education
2. Core Servlets And Java Server Pages Volume 1: Core Technologies, Marty Hall and Larry Brown Pearson
3. Internet and World Wide Web – How to program, Dietel and Nieto PHI/Pearson Education Asia.

Mode of Evaluation: Assignments, Internal Mid Examinations, External End Examination.

MCA II Year II Semester

18MCAP404 SOFTWARE TESTING

L	T	P	C
3	0	0	3

Course Prerequisite: Software Engineering concepts and OOAD Concepts.

Course Description:

This course is providing students to have a clear understanding and sound knowledge in foundations, techniques & tools and their usage in the area of software testing in industries. It makes students be competent in software testing domain, whether as a developer or a tester. It also helps students to gain in-depth knowledge in strengths and weakness in software testing techniques.

Course Objectives:

Students will be able to

1. The need of software testing, practices and their applicability in various types of software.
2. The relevance of transaction flow and dataflow in software testing.
3. Classify the domains; apply the testing strategies in domain and interfaces.
4. Identify various paths for data and instruction flow, regular expressions.
5. Get an overview about decision tables, charts, graphs and testing tips.

UNIT- I BASICS OF SOFTWARE TESTING

Introduction: Purpose of testing, Dichotomies, model for testing, consequences of bugs, taxonomy of bugs. Flow graphs and Path testing: Basics concepts of path testing, predicates, path predicates and achievable paths, path sensitizing, path instrumentation, application of path testing.

UNIT- II TRANSACTION FLOW TESTING

Transaction flows, transaction flow testing techniques. Dataflow testing:-Basics of data flow testing, strategies in data flow testing, application of dataflow testing.

UNIT-III DOMAIN TESTING

Domains and paths, Nice & ugly domains, domain testing, domain and interface testing, domains and testability.

UNIT-IV PATH AND REGULAR EXPRESSIONS

Paths, Path products and Regular expressions: path products & path expression, reduction procedure, applications, regular expressions & flow anomaly detection.

UNIT –V LOGICBASED TESTING

Overview, decision tables, path expressions, kv charts, specifications. State, State Graphs and Transition testing: state graphs, good & bad state graphs, state testing, Testability tips.

Course Outcomes:

After completion of the above course the student will be able to:

1. Identify and apply suitable testing practices in various software.
2. Draw attention towards transaction flow and data flow in various software.
3. Split the entire software into different domains; apply the testing strategies in domain and their interfaces.
4. Increase the efficiency of software by concentrating on data and instruction flow, regular expressions.
5. Identify the usage of decision tables, charts, graphs and testing tips.

Text Books:

1. Software testing techniques –Baris Beizer, Dreamtech, second edition

Reference Books:

1. Software Testing Tools–Dr.K.V.K.K.Prasad, Dreamtech.
2. The craft of software testing- Brian Marick, Pearson Education.
3. Software Testing, 3rd edition, P.C. Jorgensen, Aurbach Publications(Dist.bySPD).
4. Software Testing in the Real World–EdwardKit,Pearson.
5. Effective methods of Software Testing, Perry, John Wiley,2nd Edition,1999.

Mode of Evaluation: Assignments, Internal Mid Examination, External End Examination

MCA II Year II Semester

18MCAP405 USER INTERFACE DESIGN AND MULTIMEDIA SYSTEMS

L T P C

Course Prerequisite: Computer Basics

3 0 0 3

Course Description:

The students are provided with the basic user interface and multimedia tools along with simple animations

Course Objectives:

Students will be able to:

1. Identify the human factors of UID, controlled and psychologically oriented experiments
2. Work with latest software tools, commands and natural languages
3. Learn and understand the fundamental components of multimedia systems such as audio, video and images.
4. Understand the need of Animations, data compression and various compression techniques.
5. Acquire knowledge on fundamentals of storage mechanism using optical storage media like Video disks, CD-ROM and Multimedia workstations.

UNIT- I INTERACTIVE SOFTWARE GOALS AND USER-INTERFACE DESIGN

Theories principles and guidelines of object-action interface model - Principle 1.2 and 3 - data display and data entry, balance of automation and human control - Managing design processes - Usability, design pillars, development methodologies, ethnographic observation, usability testing, surveys, and continuing assessments – expert reviews, usability testing and laboratories - surveys acceptance tests - evaluation on active use and controlled psychologically oriented experiments

UNIT- II SOFTWARE TOOLS

Specification methods, interface- building tools and evaluation and critiquing tools -Direct manipulation and Virtual environments – Visual thinking and Visual environments. Tasks and Menu - form filling and dialog boxes – Event sequence - response time and display - Command and natural languages – Functionality to support users tasks, command –organization strategies, the benefits of structure, naming and abbreviations, command menus, natural language in computing.

UNIT- III INTRODUCTION TO MULTIMEDIA:

media and Data Streams: Medium Main Properties of Multimedia System-Multimedia-Traditional data streams Characteristics-Data Streams Characteristics for continuous Media – Information Units-Sound/Audio: Basic Concepts-Computer Image Processing.

UNIT- IV VIDEO AND ANIMATION:

Basic Concepts-Television-Computer based Animation-Data Compression:Storage Space-Coding Requirements-Source, Entropy and Hybrid coding-some Basic Compression Techniques-JPEGH.261-MPEG_DVI.

UNIT- V OPTICAL STORAGE MEDIA

Basic Technology-Video Disks and other WORMs Compact Disk Read Only Memory-CD-ROM Extended Architecture-Further CD-ROM Technologies-Computer Technology: Communication Architecture-Multimedia Workstation.

Course Outcomes:

After the completion of this course Students can able to

1. Get familiar with the human factors of UID, controlled and psychologically oriented experiments
2. Apply their knowledge on latest software tools, commands and natural languages
3. Exposed to the fundamental components of multimedia systems such as audio, video and images.
4. Implement Animations, data compression and various compression techniques.
5. Exposed to fundamentals of storage mechanism using optical storage media like Video disks, CD-ROM and Multimedia workstations.

Text Book:

1. Ben Shriderman, "Designing the user Interface, strategies for effective human- Computer, Pearson Education Asia.
2. Ralf Steinmetz and KlaraNahrstedt, Multimedia: Computing, Communications andApplications,Pearson Education Asia.

Reference Books:

1. Hix, Deborah and Hartgon, H.RR X; Developing use Interfaces, John Wiley, 1993.
2. Galitz, Wilbert O., It's Time to Clear Your Windows: Designing GUIs that Work,John Wiley and Sons, New York(1994)
3. Tay Vaughan, Multimedia Making it work, Tata McGraw-Hill, Edition, 2001
4. Jeffcoate, Multimedia in practice Technology and Application, Prentice Hall, 1995
5. John F. Koeel Buford, Multimedia systems, Addison Wesley, 1994.
6. Fred Halsall, Multimedia communications, Pearson Edition 2001.
7. Prabhat K Andleigh and KiranThatkar, Multimedia systems Design, PHI 2005.

Mode of Evaluation: Assignments, Internal Mid Examination, External End Examination

MCA II Year II Semester

18MCAP501

MOBILE APPLICATION DEVELOPMENT USING ANDROID LABORATORY

L T P C

0 0 3 1.5

Course Prerequisites: CoreJava, XML

Course Description:

This course gives the practical experience to the students to develop android applications. It provides the methodology of creating user interface for different applications using different components in android and testing the interface properly. This course also gives idea on creating applications using Sensor Classes to work with different android sensors.

Course Objectives:

Students will be able to

1. Understand the installation of Android SDK
2. Get idea on the User Interface Design and their testing methodologies
3. Know the components of Android Building Blocks and how to use them for different application developments

LIST OF EXPERIMENTS:

1. Develop an android application to display a simple text in the emulator
2. Develop an android application to display the internal keyboard in the emulator
3. i. Write an android program to display a message in the toast
 - i. Write an android program to input a text through a text and the same must be displayed in the toast when a button is clicked on the screen
4. i. Develop an application to perform 5 arithmetic operations: Addition, Subtraction, Multiplication, Division and Modulo operation with necessary user interface creation
 - ii. Develop an android application to process a student mark list by creating proper UI using the necessary controls
5. Write an android application to create a calculator
6. Create an android UI that consists of Different Departments of a company namely Production, Finance, Marketing and HR. If the user clicks on any department it should show details of that department. Use indents.
7. Design an android application to display a list of items on the android screen. If the user clicks any one of the list items a dialogue box should show that the user has clicked that particular item (Use array adopters)

8. Develop an android application to show some categories such as education, entertainment, health, provisions etc., If the user clicks on any one of the items it should show the sub categories of the category and if is again clicked it should the details of those items. (Use indents and lists)
9. i. Design an android application to create a service that shows the service is running in the background in the form of a toast
ii. Develop an android application to create an alarm using the concept of service
10. Develop an android application to demonstrate the database connectivity with the SQLite database to post and retrieve data through the User Interface(Example: Student mark list processing, Email Registration and Login, Products and sales)
11. Demonstrate the usage of Sensors in android by developing proper application.

Course Outcomes:

After completion of the above course, the student will be able to

1. Do the installation of Android SDK and write simple programs
2. Write programs for different applications by creating different user interfaces and testing them
3. Work on Sensor programs and network connectivity applications

Text Books:

1. Android Programming-The Big Nerd Ranch Guide, Bill Philips, ChristStewart, Kristin Mariscano, Big Nerd Ranch publishers, 3rd Edition
2. Android Programming for Beginners, John Horton, PACKT publishers

Reference Books:

1. Android Application Development All in One for Dummies, Barry Burd, Wiley, 2nd Edition
2. Android application Development-Black Book, Pradeep Kothari, dreamtech
3. Android Programming - Unleashed, B.M.Harwani, Pearson Education, 2013
4. Head First Android Development: A Brain-Friendly Guide, Dawn Griffiths and David Griffiths, O'Reilly, 2nd Edition
5. Android System Programming, Roger Ye, PACKT publishers
6. Programming Android, By Zigurd Mednieks, Laird Dornin, G.Blake Meike & Masumi Nakamura, O'Reilly

Mode of Evaluation: Internal Lab Examination, External Lab End Examination

MCA II Year II Semester

18MCAP502 WEB PROGRAMMING THROUGH PHP LABORATORY

L T P C

0 0 3 1.5

Course Prerequisite: Basic HTML and Advanced HTML Tags, CSS Concepts, OOPs concepts, SQL Basics

Course Description:

This course enables students to develop web application using PHP. This course helps students to create web pages using HTML and CSS and to establish connection between the front end PHP forms with the backend database. This course helps the students to develop dynamic websites for different applications.

Course Objectives:

Students will be able to

1. Understand HTML Tags – Both basics and Advanced, PHP basics, Programs using control structures and arrays in PHP
2. Know Object oriented concepts and their implementation in PHP.
3. Learn Database concepts, and work with different types of Queries in MySQL and their connectivity with PHP

LIST OF EXPERIMENTS:

1. Programs that demonstrate the usage of advanced HTML tags and CSS.
 - i. Create a Registration form with necessary fields
 - ii. Create a Login form with username and password fields with a button
 - iii. Create a table with proper borders and table heading row with a background color. Use CSS.
2. Programs using control structures in PHP by embedding with HTML
 - i. Develop a php program to generate 100 numbers in the form of a table using HTML table tag
 - ii. Develop a php program to display a triangle using * symbol upto n rows inside a table using <tr>and <td> tags
 - iii. Develop a program to display the multiplication values of the I row and I column in the corresponding table cells by using php loop constructs and html tags.
3. Programs based on Arrays

- i. Write a php program to demonstrate the usage of arrays by creating 5 data records in the array and display the records in a table.
 - ii. Write a php program to display 10 string data values in the form of an option using for loop in php and <option> tag in HTML.
 - iii. Write a program to create a table that shows the employees' salary details in the form rows and columns using the concept of php multidimensional arrays.
4. Develop a php program that performs the string manipulation operations using string functions in php.
5. Create and develop a php program / programs to demonstrate the usage of the regular expressions.
6. Develop a php program to demonstrate the usage of functions. (Use call by value and call by reference)
7. Demonstrate the usage of exception handling using try. Catch and throw in php.
8. Programs on databases:
 - i. Create a database table in MySQL and perform basic query operations
 - ii. Create a database table to demonstrate the usage of sub queries and join queries
9. Programs on HTML Forms and Get & Post Methods. Design a php program to create a
 - i. Form and apply proper validation mechanisms for a registration form. Values should be checked based on regular expressions. (Use Post Method)
 - ii. Develop a php program using Form object, Text objects and button objects in HTML to perform the arithmetic operations by getting the input from the user. (Use Post Method)
10. Programs on PHP and MySQL
 - i. Design and Create a Registration form with all necessary fields using PHP and MySQL to demonstrate the storage of data in the database.
 - ii. Create a Login form to demonstrate the user name and password authentication using PHP and MySQL
11. Applications on PHP
 - i. Develop a PHP application for a book store (Create design using HTML and PHP and connect it with MySQL for data manipulation)\
 - ii. Develop a PHP application for a home appliances shop based on the sales. Generate invoice report and weekly sales report based on products.

Course Outcomes:

After completion of the above course the student will be able to:

1. Work with basic and advanced tags in HTML & CSS and basic programs in PHP
2. Write programs in PHP Functions, Strings and Regular expressions.
3. Develop different MySQL Queries for data manipulation and mechanism of RDBMS to connect with PHP.

Text Books:

1. The Complete Reference – PHP, Steven Holzner, McGrawHill, 2007
2. Programming PHP - Creating dynamic web pages, RasmusLerdorf, KevinTatroe and Peter B. MacIntyre, O'Reilly, 2006

Reference Books:

1. The Joy of PHP, Alan Forbes, Copyright Material, 2017
2. Programming PHP MySQL- Let's Explore, HirdeshBharadwaj, Educreation Publishing, 2nd Edition
3. PHP and MySQL web development, Luke Weiling& Laura Thomson, Addison Wesley, 5th Edition
4. Head First PHP & MySQL, Lynn Beighley, Michael Morrison, "O'Reilly Media, Inc.", 2008
PHP /MySQL Programming for the Absolute Beginner, Andy Harris, Cengage Learning, 2009

Mode of Evaluation: Internal Model Practical Examination, External Lab End Examination

MCA II Year II Semester

18MCAP503 WEB TECHNOLOGIES THROUGH JAVA LABORATORY

L	T	P	C
0	0	3	1.5

Course Prerequisite: Basic Knowledge on web programming.

Course Description:

In this course, students gain extensive hands-on experience on Java programming. Students learn to create web based applications using code reusability with, applications of JSP.

Course objectives:

1. Learn to create simple webpages using HTML and CSS
2. Get idea to develop XML pages and JSP
3. Understand the JDBC connectivity concepts to develop web application

List of Experiments:

1. Design a web page on your own to demonstrate the usage of Basic and Advanced HTML Tags
2. Create a web page for a shopping portal using CSS and HTML tags
3. Creating animations using CSS3
4. Create a registration form for an examination portal and validate the fields using JavaScript
Validation criteria:
 - i. Name should be in capital letters only and length must be of at least 10 characters
 - ii. Username must be the same name in the name field along with the date of birth
 - iii. Password must be of at least 10 characters and must contain atleast two special symbols with first character must be capital letter only
 - iv. Email must satisfy the email criteria
 - v. Phone number should contain only 10 digits
5. Write a JavaScript program to perform the arithmetic operations
6. Developing application using DHTML Filters
7. Create an XML page to display the username and password of the registered users
8. Create a web page for book information. Write an XML page to display the book information created using the web page
9. Implement Client and Server Socket program to send and receive messages

10. Create a simple visual bean with a area filled with a color. The shape of the area depends on the property shape. If it is set to true then the shape of the area is Square and it is Circle, if it is false. The color of the area should be changed dynamically for every mouse click. The color should also be changed if we change the color in the “property window “
11. Designing a scientific calculator using java script
12. Install a database (Mysql or Oracle). Create a table which should contain at least the following fields: name, password, email-id, phone number Write a java program/servlet/JSP to connect to that database and extract data from the tables and display them. Insert the details of the users who register with the web site, whenever a new user clicks the submit button in the registration page.
13. Write a JSP which insert the details of the 3 or 4 users who register with the web site by using registration form. Authenticate the user when he submits the login form using the user name and password from the database

Course outcomes:

After the completion of the course, Students will be able to

1. Develop simple web pages using HTML and CSS and Java Script
2. Implement Java Server Page concepts in web application
3. Create a full pledged web application based on database connectivity using JDBC

References:

1. HTML & CSS-Design and Build Websites, John Duckett, John Wiley,
2. An Introduction to XML and Web Technologies, Anders Mallar, Michal Schwartzback, Pearson Education, 2005
3. Speaking JavaScript-An indepth guide for Programmers, Axel Rauschmayer, 2013
4. Mastering JSP; A Server side Technology, Anil Barnwal, VS. Dixit, Alpha Science International Limited, 2015
5. JDBC – Practical Guide for Programmers, Gregory D.Speegle, Morgann Kaufmann Publishers, 2002

Mode of Evaluation: Continuous Internal Evaluation, Practical Examination.

MCA II Year II Semester

18MCAP504

SOFTWARE TESTING LABORATORY

L	T	P	C
0	0	3	1.5

Course Prerequisite: Software Engineering concepts and OOAD Concepts

Course Description:

This course is providing students to have a clear understanding and sound knowledge in foundations, techniques & tools and their usage in the area of software testing in industries. It makes students be competent in software testing domain, whether as a developer or a tester. It also helps students to gain in-depth knowledge in strengths and weakness in software testing techniques.

Course Objectives:

Students will be able to

1. Win Runner/QTP for functional testing.
2. LoadRunner for Load/Stress testing.
3. Test Director for test management.

LIST OF EXPERIMENTS:

1. Write programs in 'C' Language to demonstrate the working of the following constructs:
 - a. do...while
 - b. while....do
 - c. if...else
 - d. switch
 - e. for
2. "A program written in 'C' language for Matrix Multiplication fails" Introspect the causes for its failure and write down the possible reasons for its failure.

3. Take any system (e.g. ATM system) and study its system specifications and report the various bugs.
4. Write the test cases for any known application (e.g. Banking application)
5. Create a test plan document for any application (e.g. Library Management System)
6. Study of any testing tool (e.g. Winrunner)
7. Study of any web testing tool (e.g. Selenium)
8. Study of any bug tracking tool (e.g. Bugzilla, bugbit)
9. Study of any test management tool (e.g. Test Director)
10. Study of any open source-testing tool (e.g. Test Link)
11. Take a mini project (e.g. University admission, Placement Portal) and execute it. During the
12. Lifecycle of the mini project create the various testing documents* and final test report document
13. Test the following using JUnit and CPPUNIT:
 - a. Sorting problems
 - b. Searching problems
 - c. Finding gcd of two integers
 - d. Finding factorial of a number.
14. Test web based forms using HTML Unit.
15. Test database stored procedures using SQL Unit.

Course Outcomes:

After completion of the above course the student will be able to:

1. Understand and apply the working mechanism of Win Runner/QTP for functional testing.
2. Understand and apply the working mechanism of LoadRunner for Load/Stress testing.
3. Design and solve the Test Director for test management and apply the automated testing procedures for JUnit, HTML Unit, CPP Unit

Reference Books:

1. Software Testing Tools–Dr.K.V.K.K.Prasad, Dreamtech.
2. The craft of software testing- Brian Marick, Pearson Education.
3. Software Testing, 3rd edition, P.C. Jorgensen, Aurbach Publications(Dist.bySPD).
4. Software Testing in the Real World–EdwardKit,Pearson.
5. Effective methods of Software Testing, Perry, John Wiley,2nd Edition,1999.

Mode of Evaluation: Continuous Internal Evaluation, Practical Examination.

MCA II Year II Semester

18MCAP505

USER INTERFACE DESIGN AND MULTIMEDIA SYSTEMS LABORATORY

L	T	P	C
0	0	3	1.5

Course Prerequisite: Computer Basics

Course Description:

The students are provided with the basic user interface and multimedia tools along with simple animations

Course Objectives:

After the completion of this course Students are able to

1. Understand and design drawing pad and create calculator application and the usage of ADO control
2. Understand and design Digital clock and web browser application and Develop an application like Photo slideshow and to show boat sailing in water using Macromedia Flash.
3. Design an application for creating an advertisement using Layers, Motion Tween & Shape, between and also importing an objects from the library and apply zoom in /zoom out effects using Macromedia Flash. And Create Publishing Banner Ads and Quiz on C-Language or Java.

List of Experiments:

1. Write a program to calculate Maximum Number from the given numbers

Description:

This program allows the user to enter three hidden numbers and the program can calculate the maximum number among the three numbers. In order to allow the user to enter the password in hidden mode, you have to set the Password Char property to alphanumeric symbols such as *

2. Write a program in Visual Basic to evaluate Simultaneous Equations Solver

Description: Simultaneous equations are equations that involves two or more unknown variables. There must be as many equations as the number of unknown variables in order for us to solve the problem.

3. Write a program in Visual Basic to calculate Prime Number

Description: This program can test whether a number entered by the user is a prime number or not. A prime number is a number that cannot be divided by other numbers other than by itself. Examples are 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37 and more.

4. Write a program in Visual Basic to Simple Harmonic Motion

Description: Simple harmonic motion is the motion of a simple harmonic oscillator. The motion is periodic, as it repeats itself at standard intervals in a specific manner with constant amplitude. It is characterized by its amplitude, its period (the time for a single oscillation), its frequency (the number of cycles per unit time), and its phase (which determines the starting point on the sine wave). The period, and its inverse the frequency, are constants determined by the overall system, while the amplitude and phase are determined by the initial conditions (position and velocity) of that system.

5. Write a program in Visual Basic to Simple Harmonic Motion Boogle

Description: Boogle is a type of words puzzle game where the players can form as many words as possible from the characters displayed on an $n \times n$ square. Words can be formed in many ways, from left to right, from right to left, top to bottom, bottom to top, diagonal, zigzag manner and etc.

6. Write a program in Visual Basic to Drawing pad

Description: the user needs to fill in all the coordinates and selects a color before proceeding to draw the required shape. If the user forgets to fill in the coordinates or selects a color, he or she will be prompted to do so.

7. Write a program in Visual Basic to Calculator

8. Write a program in Visual Basic to using ADO control

9. Write a program in Visual Basic to create digital clock

10. Write a program in Visual Basic to create Web Browser

11. Design and develop a software for Photo slideshow using Macromedia Flash.

12. Design and develop a software to show boat sailing in water using any Multimedia S/W

13. Design and develop a software for creating an advertisement using Layers, Motion Tween & shape, tween in Macromedia flash.

14. Design and develop a software for importing an object from the library apply the zoom in effect and zoom out effect using Macromedia flash.

15. Design and develop a software for creating a Publishing Banner ads using Macromedia Flash.

Course Outcomes:

After the completion of this course Students are able to

1. Drawing pad and creating calculator applications and Utilize the ADO control

2.. Design Digital clock and web browser applications and Develop an application like Photo slideshow and to show boat sailing in water using Macromedia Flash.

3.Creating an advertisement using Layers, Motion Tween & Shape, tween and also importing an objects from the library and apply zoom in /zoom out effects using Macromedia Flash and Publishing Banner Ads and Quiz on C-Language or Java.

Text Book:

1. Ben Shneiderman, "Designing the user Interface, strategies for effective human- Computer, Pearson Education Asia.
2. Ralf Steinmetz and KlaraNahrstedt, Multimedia: Computing, Communications and Applications, Pearson Education Asia.

Reference Books:

1. Hix, Deborah and Hartson, H.R.R X; Developing use Interfaces, John Wiley, 1993.
2. Galitz, Wilbert O., It's Time to Clear Your Windows: Designing GUIs that Work, John Wiley and Sons, New York(1994)
3. Tay Vaughan, Multimedia Making it work, Tata McGraw-Hill, Edition, 2001
4. Jeffcoate, Multimedia in practice Technology and Application, Prentice Hall, 1995
5. John F. Koeel Buford, Multimedia systems, Addison Wesley, 1994.
6. Fred Halsall, Multimedia communications, Pearson Edition 2001.
7. Prabhat K Andleigh and KiranThatkar, Multimedia systems Design, PHI 2005.

Mode of Evaluation: Continuous Cumulative evaluation of the lab experiments, Record, Viva-voce and External Lab Examination.

Discipline Elective - II

MCA II Year II Semester

18MCAP406 DEEP LEARNING

L	T	P	C
4	0	0	4

Course Prerequisite: Algorithms, Networks, Machine Learning

Course Description:

This course will cover Imaging, Enhancement, Restoration and Filtering. Deep Learning in Imaging will construct an expert Computer vision and also a form of artificial intelligence that automates Image analysis and data analysis. Neural networks will facilitate computers to learn and adapt through experience to do specific tasks without explicit programming.

Course Objectives:

Students will be able to

1. Know the basic concepts of Images and its transformations
2. Learn Image Enhancement, Restoration and Filtering
3. Gain knowledge on boundary detection and clustering
4. Understand CNN and NLP associated with Deep learning
5. Learn Neural Networks in Deep Learning

UNIT I: DIGITAL IMAGE FUNDAMENTALS

Image Formation and types – Basic geometric transformations – Fourier Transforms – Walsh – Hadamard – Discrete Cosine Transforms.

UNIT II: IMAGE ENHANCEMENT AND RESTORATION

Histogram Modification Techniques – Image Smoothing – Image Sharpening – Image Restoration – Degradation Model – Noise Models – Spatial Filtering – Frequency Domain Filtering.

UNIT III: DEEP LEARNING IN COMPUTER VISION

Detection of Discontinuities – Edge Linking and Boundary Detection – Thresholding – Region Based Segmentation – Morphology operations - Pattern classification - Clustering and Matching - Knowledge representation in scene analysis - image understanding (2D and 3D) - Object recognition and identification – Case study of various applications.

UNIT IV: DEEP LEARNING IN CNN

Introduction to NLP - Classification using Convolution Neural Networks – Convolution and Pooling– Deep supervised Learning - Natural Language Processing using Deep Learning - Applications of Dynamic Memory Networks in NLP

UNIT V: DEEP LEARNING IN RNN

Parsing – Sentiment Analysis Using Recursive Neural Networks –LSTM - Multiclass discrimination- Language modeling –Soft Attention - Training procedures- localized network structure- dimensionality reduction interpretation.

Course Outcomes:

After completion of the above courses the student will be able to:

1. Acquire knowledge of Images and transformations.
2. Examine the Image Enhancement, Restoration and Filtering
3. Implement algorithms for boundary detection and Clustering
4. Acquire knowledge on CNN and NLP
5. Implement Neural networks for real world inputs through Deep learning

Text Books:

1. Milan Sonka, Vaclav and Roger Boyle, “Image Processing, Analysis and Machine Vision”, Thomson, 2012.
2. Bengio, Yoshua, Ian J. Goodfellow, and Aaron Courville. "Deep learning." An MIT Press book in preparation. 2015.
3. Aaron Courville, Ian Goodfellow, and Yoshua Bengio “Deep Learning”, MIT Press, 2017.

Reference Books:

1. R. Szeliski, “Computer Vision: Algorithms and Applications”, Springer 2011.
2. AP Dawan, “Medical Image Analysis”, Wiley 2011.

Mode of Evaluation: Assignments, Internal Mid Examination, External End Examination

Course Prerequisites: Statistics Basics and Basic Concepts on Big Data Analytics

Course Description:

This course will introduce students to this rapidly growing field and equip them with some of its basic principles and tools as well as its general mindset. Students will learn concepts, techniques and tools they need to deal with various facets of data science practice, including data collection and integration, exploratory data analysis, predictive modeling, descriptive modeling, data product creation, evaluation, and effective communication.

Course Objectives:

Students will be able to :

1. Have an idea on data science and the methods of distributions and the concepts of R Language
2. Understand the Machine learning concepts and the algorithms
3. Know about Data Wrangling concepts and Feature Algorithms
4. Learn about text mining techniques and graph databases
5. Analyze data visualization concepts and know ethics on data science

UNIT – I INTRODUCTION

What is Data Science? – Big data and Data Science –Facets of Data – Data fication – Current landscape of perspectives – Statistical Inference: Population – Samples – Statistical Modelling – Probability: Discrete and Continuous, Distributions: Binomial, Poison and Normal distributions – Fitting a Model – Introduction to R Language: Data Types – Vectors – Sorting – Indexing - R Markdown

UNIT – II EXPLORATORY DATA ANALYSIS AND DATA SCIENCE

Basic Tools: Plots, Graphs, The Data Science Process: Goals, Data Retrieval, Cleansing, Integrating and Transforming Data – Machine Learning: Introduction to Machine Learning – Applications of Machine Learning in Data Science - Machine Learning Algorithms: Linear Regression – k-Nearest Neighbor (k-NN) – k –means – Filtering Span – Drawbacks of Linear Regression and k-NN in Filtering Span - Naïve Bayes and Filtering Span

UNIT – III DATA WRANGLING, FEATURE GENERATION AND SELECTION, RECOMMENDATION SYSTEMS

Data Wrangling: API's and other tools for Web scrapping – Feature Generation: Role of domain expertise – Imagination – Feature Selection – Feature Selection Algorithms: Filters, Wrappers, Decision Trees, Random Forests – Recommendation Systems: Algorithmic Ingredients of Recommendation Engine – Dimensionality Reduction – Singular Value Decomposition – Principal Component Analysis

UNIT – IV TEXT MINING AND TEXT ANALYTICS, GRAPH DATABASES

Text mining in the real world – Text Mining Techniques: Stemming and lemmatization – Decision Tree Classifier – Case Study: Classifying Reddit Posts - Graph Databases: Introducing connected data and Graph databases – Neo4j Graph database – Case Study: A recipe recommendation engine example

UNIT – V DATA VISUALIZATION, DATA SCIENCE AND ITS ETHICS

Basic Principles – Ideas and Tools for Data Visualization – Crossfilter, JavascriptMadReduce Library – Interactive dashboard – Dashboard development tools – Privacy, Security, Ethics – Next Generation Data Scientists.

Course Outcomes:

After completion of the above courses the student will be able to

1. Identify the methods of distribution and work with the concepts of R-Language
2. Apply Machine Learning algorithms for a particular scenario
3. Work on Data Wrangling and Analyze Feature Algorithms
4. Implement Text Mining Techniques and work with graph databases
5. Use Data Visualization concepts based on requirement and apply data science ethics in real situation

Text Books:

1. Introduction to Data Science, Davy Cielen, Arno D.B. Meysman, Mohamed Ali, Manning publications, 2016
2. Doing Data Science, Straight Talk From The Frontline. Cathy O'Neil and Rachel Schutt. O'Reilly. 2014.

Reference Books:

1. Jure Leskovek, AnandRajaraman and Jeffrey Ullman. Mining of Massive Datasets. v2.1, Cambridge University Press. 2014. (free online)
2. Kevin P. Murphy. Machine Learning: A Probabilistic Perspective. ISBN 0262018020. 2013.

3. Foster Provost and Tom Fawcett. Data Science for Business: What You Need to Know about Data Mining and Data-analytic Thinking. ISBN 1449361323. 2013.
4. Trevor Hastie, Robert Tibshirani and Jerome Friedman. Elements of Statistical Learning, Second Edition. ISBN 0387952845. 2009. (free online)
5. Avrim Blum, John Hopcroft and RavindranKannan. Foundations of Data Science.
6. Mohammed J. Zaki and Wagner Miera Jr. Data Mining and Analysis: Fundamental Concepts and Algorithms. Cambridge University Press. 2014.
7. Jiawei Han, MichelineKamber and Jian Pei. Data Mining: Concepts and Techniques, Third Edition. ISBN 0123814790. 2011.

Mode of Evaluation: Assignments, Internal Mid Examination, External End Examination

MCA II Year II Semester

18MCAP408 CRYPTOGRAPHY AND NETWORK SECURITY

L T P C

4 0 0 4

Course Prerequisite: Computer Networks

Course Description:

Network Security introduces techniques for protecting information and network components against attacks. It highlights the core cryptographic mechanisms and their implementation procedure to provide security for the data. Investigates various networking security standards and methods for enforcing and enhancing those standards. It also covers the electronic mailing system and supporting protocols.

Course Objectives:

Students will be able to

1. Gain knowledge about symmetric key encryption techniques
2. Study of asymmetric key cryptographic algorithms and encryption methods for security.
3. Understand the message authentication codes and hashing algorithms.
4. Understand the method of public, private key distribution and applications related to authentication.
5. Know about IP security and Transport layer security with supporting protocols

UNIT – I SYMMETRIC CIPHERS

Introduction to security attacks, services and mechanisms, Classical Encryption Techniques – Substitution Ciphers and Transposition Ciphers, cryptanalysis, Steganography, Stream and Block Ciphers. Modern Block Ciphers: Block cipher principles, Shannon's theory of confusion and diffusion, feistel structure, modular arithmetic, Data encryption standard (DES), Advanced Encryption Standard (AES).

UNIT – II ASYMMETRIC CIPHERS

Prime and Relative Prime numbers, Extended Euclidean Algorithm, Fermat's and Euler's theorem, primality testing, Chinese Remainder Theorem, Discrete Logarithmic Problem, Principle of Public key crypto systems, RSA algorithm, security of RSA.

UNIT – III MESSAGE AUTHENTICATION

Message Authentication Codes: Authentication requirements, authentication functions, message authentication code, hash functions, birthday attacks, security of hash functions, Secure hash algorithm

(SHA). Digital Signatures: Digital Signatures, Elgamal Digital Signature Techniques, Digital signature standards (DSS).

UNIT – IV KEY DISTRIBUTION & AUTHENTICATION APPLICATIONS

Key Management and distribution: Symmetric key distribution, Diffie-Hellman Key Exchange, Public key distribution, X.509 Certificates, Public key Infrastructure (PKI). Authentication Applications: Kerberos, Electronic mail security: pretty good privacy (PGP), Secure Multipurpose Internet Mail Extensions (S/MIME).

UNIT – V NETWORK & TRANSPORT LAYER SECURITY

IP Security: Architecture, Authentication header, Encapsulating security payloads, combining security associations, key management. Introduction to Secure Socket Layer, Secure electronic transaction (SET) System Security: Introductory idea of Intrusion, Intrusion detection systems, Viruses and related threats.

Course Outcomes:

After completion of the above courses the student will be able to

1. Acquire knowledge about block ciphering and symmetric ciphering techniques.
2. Implement RSA algorithm
3. Execute Secure Hashing and Digital Signature Algorithms.
4. Implement authentication applications Kerberos and MIME.
5. Apply Knowledge on IP security, Transport layer security Protocols and system security

Text Books:

6. William Stallings, “Cryptography and Network security Principles and Practices”, Pearson/PHI, 4th Edition.
7. Behrouz A. Frouzan, “Cryptography and Network Security”, Tata McGraw Hill, 2nd Edition.

References Books:

1. Bruce Schneier, “Applied Cryptography”. John Wiley & Sons
2. Bernard Menezes, “Network Security and Cryptography”, Cengage Learning.
3. Atul Kahate, “Cryptography and Network Security”, TMH

Mode of Evaluation: Assignments, Internal Mid Examination, External End Examination

MCA II Year II Semester

18MCAP506 DEEP LEARNING LABORATORY

L	T	P	C
0	0	3	1.5

Course Prerequisite: Algorithms, Image Processing, Machine Learning

Course Description:

This course will cover Imaging, Enhancement, Restoration and Filtering. Deep Learning in Imaging will construct an expert Computer vision and also a form of artificial intelligence that automates Image analysis and data analysis. Neural networks will facilitate computers to learn and adapt through experience to do specific tasks without explicit programming.

Course Objectives:

Students will be able to:

1. Learn real world problems conversion into AI
2. Know the clustering algorithms segment objects.
3. Understand Neural networks for known inputs and get unknown outputs

List of Experiments:

1. Collect the number of words in a paragraph and find out positive and negative terms in it.
2. Read a curriculum vitae and find whether the candidate completed post-graduation degree
3. Read an Image and change its dimensions
4. Read an Image and change it to Binary Image
5. Read an Image and Find the Histogram for it
6. Read an Image and cluster different colors available in it
7. Read an Image and detect boundaries in it
8. Read an Image and detect face in it
9. Read an Image and detect specific shapes in it
10. Read a paragraph identify and count the alphabets and numerical values using NLP
11. Read an audio file find the smallest and highest frequencies in them
12. Read known inputs and practice an output format. Apply Neural Network and do the same for unknown inputs
13. Read marks of few students and practice a grade value. Apply Neural Network and try for new grade values in them

Case Studies:

- I. Deep Learning in Text

- II. Deep learning in Number System
- III. Deep learning in Image and formats
- IV. Deep learning in Clustering
- V. Deep learning in Natural Language Processing
- VI. Deep Learning in Neural networks

Course Outcomes:

After successful completion of this lab, students can be able to

- 1. Apply real world problems into AI
- 2. Utilize the clustering algorithms to segment objects.
- 3. Apply Neural networks for known inputs and get unknown outputs

Text Books:

- 1. Phil Kim, “Matlab Deep Learning: with Machine Learning, Neural Networks and Artificial Intelligence” APress, 2017.
- 2. AP Dawan, “Medical Image Analysis”, Wiley, 2011.

Reference Books:

- 1. R. Szeliski, “Computer Vision: Algorithms and Applications”, Springer 2011.
- 2. Valentino zocco, “Python Deep Learning”, PackT press, 2017.

Mode of Evaluation: Continuous Cumulative evaluation of the lab experiments, Record, Viva voce and External Lab Examination.

MCA II Year II Semester

18MCAP507 FUNDAMENTALS OF DATA SCIENCE LABORATORY

L T P C

0 0 3 1.5

Course Prerequisites: Concepts of Statistics and probability

Course Description:

This course helps the students to work on different sets of data and analyze those using different techniques. Also this course gives an idea on working on R Language. Moreover, it introduces the concepts of MongoDB to work on them.

Course Objectives:

Students will be able to

1. Understand different data sets and their analysis using various techniques
2. Know how to do simple programming concepts in R and plotting techniques using R
3. Get knowledge on MongoDB queries

LIST OF EXPERIMENTS:

1. Take a set of Data and use chi square test to analyze the data
2. Taka a set of Data and analyze and apply T-test and F-test for the same
3. Demonstrate the concept of if ... else constructs and for Loop structures using R Language
4. Demonstrate the concepts of Argument matching in R
5. Take a data set and perform regression analysis
6. Generate a histogram for the rainfall for every 3 days in a winter season and plot this in a DataFrame
7. Perform the following operations in PLSQL: Creating tables, Distribution and Partitioning, Indexes and Creation of External Tables.
8. Take a set of Time Series data (for example, No. of packets arrival per second from the server and number of packets sent from server) of a computer center and perform the regression time series analysis for high data rates and low data rates
9. Prepare a report on the orders of 5 years of data of any organization using RandomForest
10. Perform some basic operations on MongoDB Query Language

Course Outcomes:

After completion of the above courses the student will be able to:

1. Analyze and find the result based on the output for different sets of data using different statistical techniques
2. Work on simple concepts of R Language and analysis parts using R Language techniques and working with DataFrames
3. Execute and write queries on MongoDB

Text Books:

1. Mining of Massive Datasets. v2.1, Cambridge University Press. 2014, Jure Leskovek, AnandRajaraman and Jeffrey Ullman.
2. Elements of Statistical Learning, Second Edition. ISBN 0387952845. 2009, Trevor Hastie, Robert Tibshirani and Jerome Friedman

Reference Books:

1. Data Mining and Analysis: Fundamental Concepts and Algorithms. Cambridge University Press. 2014. Mohammed J. Zaki and Wagner Miera Jr.
2. NoSQL For Dummies, Adam Fowler, Wiley, 2015
3. Big data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's businesses, Wiley, 2013

Mode of Evaluation: Continuous Cumulative evaluation of the lab experiments, Record, Viva voce and External Lab Examination.

Course Prerequisite: Networking Concepts, C and Java Programming

Course Description:

This course provides technical grounding in network security concepts and algorithms. It deals with the cryptographic algorithms like DES, AES, RSA and Message Digest etc. In essence, it helps the user to secure the data confidentially. It provides the platform to work with open source networking tools like Honeypot and Snort.

Course Objectives:

The students will be able to:

1. Implement the characters handling and converting ASCII values. And Apply the substitution and transposition ciphering Techniques.
2. Gain Knowledge on algorithms like DES, RSA, Diffie-Hellman and Implement the hashing algorithms MD5,SHA-1.
3. Implement the open source tools for network security and analysis.

LIST OF EXPERIMENTS

1. Write a C Program to find the frequency of occurrence of characters
2. Write a C Program to reverse a given without using built in function
3. Write a C Program to read a given string and print the ASCII value of each character
4. Write a C program to implement the concept of Ceaser Cipher
5. Write a C program to implement the concept of Playfair Cipher
6. Write a C program to implement the concept of Hill Cipher
7. Write a C program to implement the concept of Vigenere Cipher
8. Write a C program to implement the concept of Rail fence – row & Column Transformation
9. Write a Java program to implement the concept of DES algorithm
10. Write a Java program to implement the concept of AES algorithm
11. Write a Java program to implement the concept of RSA algorithm
12. Write a Java program to implement the concept of Diffie- Hellman Key Exchange Algorithm
13. Write a Java program to implement the concept of MD5

14. Write a Java program to implement the concept of SHA-1
15. Write a C program to implement the signature scheme named digital signature standard
16. Setup a honey pot and monitor the honeypot on network (KF Sensor)
17. Demonstrate intrusion detection system (ids) using any tool (snort or any other s/w)
18. Select a Network device, Analyze and Capture the network packets using Wireshark tool.
19. Identify the MAC Addresses, IPV4 and Port Addresses using Wireshark tool.

Course Outcomes:

After successful completion of the course, students will be able to

1. Implement the looping and string handling concepts and Develop the various types of ciphering techniques
2. Implement the symmetric and asymmetric key cryptographic algorithms and the data integrity algorithms MD5, SHA-1
3. Implement the network security using open source tools like Wireshark, Honeypot

Reference Books:

1. Bruce Schneier, "Applied Cryptography". John Wiley & Sons
2. Bernard Menezes, "Network Security and Cryptography", Cengage Learning.
3. Atul Kahate, "Cryptography and Network Security", TMH

Mode of Evaluation: Continuous Cumulative evaluation of the lab experiments, Record, Viva voce and External Lab Examination.

Discipline Elective - III

MCA III Year I Semester

18MCAP409 ARTIFICIAL INTELLIGENCE

L	T	P	C
4	0	0	4

Course Prerequisite: Algorithms, Networks, Operating Systems

Course Description:

This course aims at providing advanced knowledge in the field of Industrial Robotics and the associated artificial intelligence.

Course Objectives:

Students will be able to

1. Comprehend the basic concepts of AI, Problems and algorithms in it.
2. Categorize various procedures in game search algorithms
3. Define the knowledge of various knowledge representation schemes
4. Analyze various Expert System tools and applications
5. Summarize the AI on Autonomous Systems

UNIT I: INTRODUCTION TO AI:

AI Problem formulation, Foundations of AI - Problem definition – Production systems - Graph search - of A* search algorithm - AO* algorithm.

UNIT II: SEARCHING AND GAME TREES:

Optimal decisions in Games - Minimax procedure - Alpha-beta pruning – Alternative approaches

UNIT III: KNOWLEDGE REPRESENTATION:

Proportional Theorem – First order Inference rule – Baye’s Rule in knowledge representation - Representing knowledge in Uncertain Domain.

UNIT IV: INTRODUCTION TO EXPERT SYSTEMS:

Inference - Forward chaining - Backward chaining - Knowledge acquisition – Dynamic Bayesian Networks – Robotic Perception

UNIT V: AI ON AUTONOMOUS SYSTEMS:

Early Systems and Current Systems – Automatic Systems to Autonomous Systems – Imaging and Intelligence in Autonomous Systems - Ethical principles and democratic prerequisites in AI

Course Outcomes:

After the successful completion of this course

1. Acquire the basic concepts of AI, Problems and algorithms in it.
2. Distinguish various procedure in game search algorithms
3. Justify the knowledge of various knowledge representation schemes
4. Implement various Expert System tools on real world applications
5. Convert the AI concepts into Autonomous Systems

Text Books:

1. Stuart J.Russell and Peter Norvig, "Artificial Intelligence", Third Edition, Pearson, 2015.
2. G. Luger, W. A. Stubblefield, "Artificial Intelligence", Third Edition, Addison-Wesley Longman, 1998.
3. Kevin, Elaine and Nair, "Artificial Intelligence", Third Edition, McGraw Hill, 2017

Reference Books:

1. David L Pool and Alan K Mackworth, "Artificial Intelligence", Cambridge University Press, 2017
2. Wolfgang Ertal, "Introduction to Artificial Intelligence", Springer, 2017.

Mode of Evaluation: Assignments, Internal Mid Examination, External End Examination.

MCA III Year I Semester

18MCAP410

BIG DATA ANALYTICS

L T P C

4 0 0 4

Course Prerequisites: Data Mining Concepts

Course Description:

This course provides fundamental knowledge on Big Data Analytics, Big Data Platform, Intelligent data analysis, Analytic Processes and Tools. Also it guides through the Hadoop platform and its environment. Next it introduces the concepts of MongoDB and NoSQL, Working with MongoDB and NoSQL. This course also discusses about Mining Data Streams and provides knowledge on Hive Environment, Pig and Hbase applications.

Course Objectives:

Students will be able to

1. Understand Types of Data, Big Data Concepts and Tools for analyzing Big Data
2. Learn all concepts in Hadoop and its architecture, Algorithms related to Hadoop and Hadoop Yarn concept
3. Know the importance of MongoDB and NoSQL and their applications.
4. Gain knowledge on Mining Data Streams and its applications
5. Recognize the usage of HIVE, Pig and Hbase in Big Data Analytics

UNIT – I INTRODUCTION TO BIG DATA

Different types of Data: Structured, Semi structured, Unstructured and Metadata – Definition and Evolution of Big Data – Elements of Big Data - What is Big Data Analytics – Differences between Traditional Analysis and Big Data Analytics, Advantages of Big Data Analytics –Classification of Analytics - Big Data Analytics Life Cycle–Tools and Methods of Analytic Processing – Analysis Vs Reporting – Modern Data Analytic Tools

UNIT – II UNDERSTANDING HADOOP

Introduction to Hadoop – History of Hadoop – RDBMS Vs Hadoop – Functionality of Hadoop – Hadoop Ecosystems – Hadoop Distributed File Systems: HDFS Architecture and Features – MapReduce Features - MapReduce Execution – Algorithms using MapReduce–Matrix Vector Multiplication – Relational Algebra Operations - Hadoop YARN

UNIT – III MONGODB AND NOSQL

MongoDB Introduction – Necessity of MongoDB – Datatypes – MongoDB Query Language – Introduction to NoSQL – NoSQL Data Architecture Pattern – ACID and SQL – The BASE property of NoSQL- The CAP Theorem - NoSQL to Manage Big Data –Big Data Problems using NoSQL– NoSQL Databases

UNIT IV MINING DATA STREAMS

Stream Data Model and Architecture - Data Stream Management Systems – Data Stream Mining and Examples of Data Stream Applications - Mining Time Series Data –Stream Queries – Issues in Data Stream Query Processing – Sampling in Data Streams –Filtering Streams – Counting Distinct Elements in a Stream. Counting Ones in a Window

UNIT V HIVE AND PIG, HBASE

Introduction to Hive – Hive Data Types – File Formats – Query Language – User Defined Functions – Introduction to Pig – Anatomy of Pig – Pig Data Models - Hadoop on Pig – Data Types – HDFS Commands – Operators –Piggy Bank - User Defined Functions – Pig Vs Hive – Hbase Introduction – Hbase Clients

Case Studies

1. Product Placements and Store layout

Business Challenge:

A major omni-channel retailer knew that in-store layout, merchandising and product placement affected sales. Yet the company’s brick-and-mortar stores lacked “pre-cash register” visibility into how its customers shopped before they made decisions. The company wanted the same level of customer path visibility and analysis that its clickstream data gave for customers visiting its website. In-store sensors, RFID tags and QR codes could fill that data gap, but those technologies generate data in formats and volumes that the company’s legacy systems were illequipped to handle. The retailer became a relatively early adopter of Hadoop because the platform did not enforce a schema-onload paradigm that would have hampered ingestion and storage of the location data needed for the program.

Solution:

The company began testing iBeacon technology in its flagship stores. iBeacons capture in-store location data from the shoppers’ iPhones and Android devices. The data then streams into Big data, revealing how customers move through the retail stores (which can be compared to the location of particular product categories). As the iBeacon program grows, Big data can store and process that huge volume of sensor and micro-location data.

Results:

Though the results of this specific pilot are not public, the retailer’s big data analytics program boosted store sales by 10 percent. As data in Big data helps the company optimize its in-store

experience, it sees the potential for additional programs that reduce unnecessary inventory and improve customer satisfaction through smarter product placement and updates to store layouts.

2. Product Recommendation Engine

Business Challenge:

A major specialty department store wanted to improve its product marketing precision. The marketing team wanted to roll out personalized promotions, coupons and product recommendations over multiple customer touch points: in-store, kiosk, web and mobile apps. The company was particularly interested in enabling in-store, real-time product promotion among its shoppers. But the company's customer data was fragmented, and this prevented it from developing those data-driven marketing promotions. For example, the website or a kiosk should not recommend a product that the same shopper had already purchased in the store. Financial obstacles blocked proposals for an IT project to modernize the data architecture

Solution:

Now a Big data data lake integrates all the raw data from customers across different product lines. The company ingests and integrates data in real-time and batch, in both structured and unstructured formats. An ETL process transform the raw data, which is then consumed by learning algorithms. The retailer can now deliver real-time recommendations and promotions through all channels, including its website, store kiosks and mobile apps

Results:

This retailer built an omni-channel recommendation engine similar to what Amazon does online. Thirty-five percent of what consumers purchase on Amazon and seventy-five percent of what they watch on Netflix comes from such product recommendations based on that type of analysis. This retailer can vary recommendations based on weather, loyalty, purchase history, abandoned carts or life stage triggers—and deliver those to shoppers in its stores

Course Outcomes:

After completion of the above course the student will be able to

1. Classify necessary tools for analyzing Big Data
2. Work with the concepts in Hadoop, algorithms related to Hadoop and how to apply them in respective areas.
3. Execute MongoDB and NoSQL commands and apply them in applications.
4. Apply the ideas on Mining Data Streams
5. Identify the usage of HIVE, Pig and Hbase, classify them to do analysis based on given applications

Text Books:

1. Hadoop – The Definitive Guide, .Chris Eaton, Dirk DeRoos, Tom Deutsch, George Lapis, Paul Zikopoulos,3rd Edition, O'Reilly Media, 2012

2. Big Data Analytics, Radhashankarmani, M.Vijayalakshmi, Wiley

Reference Books:

1. Big Data – Black Book, DT Editorial Services, Dreamtech Press, 2015
2. Big Data Analytics, VenkatAnkarm, Packt publishers
3. Big Data Analytics, Seema Acharya, SubhashiniChellappan, Wiley
4. Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics, Bill Franks, Wiley, 2012
5. Big data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's businesses, Wiley, 2013
6. Big Data Analytics – Made Easy, Y.Lakshmi Prasad, Notion Press, 2016
7. NoSQL For Dummies, Adam Fowler, Wiley, 2015
8. NoSQL Distilled – A Brief Guide to the Emerging World of Polyglot Persistence, P.J.Sadalage and M.Fowler, Addison Wesley, 2012
9. Intelligent Data Analysis, Michael Berthold, David J. Hand, Springer 2007
10. Practical Big Data Analytics, NatarajDasgupta, Packt publishing, 2018

Mode of Evaluation: Assignments, Internal Mid Examination, External End Examination

MCA III Year I Semester

18MCAP411 CYBER SECURITY AND CYBER FORENSICS

L	T	P	C
4	0	0	4

Course Prerequisites: Computer Networks, Cryptography and Network Security Basics.

Course Description:

This course provides importance of Cyber Security, its objectives and Policies. It also focuses on Cyber Forensics investigation process and Cyber Forensics Laws and legal considerations.

Course Objectives:

Students will be able to

1. Know the need of Cyber Security
2. Understand the Objectives of cyber security
3. Emphasis on cyber security policies
4. Understand the cyber forensics investigation process
5. Get familiar with cyber forensics laws

UNIT –I INTRODUCTION

Cyber Attacks, Viruses, Worms and other Malware, Intrusion Detection and Prevention Introduction to Cyber Security, Domains of Cyber Security Policies, Cyber Security Evolution-Productivity, Internet, E-commerce, Counter Measures, Challenges

UNIT-II CYBER SECURITY OBJECTIVES AND DECISION MAKING

Cyber Security Objectives – Cyber Security Management – Metrics, Security Management goals, Counting Vulnerabilities, Security Frame work, Security Policy Objectives, Guidance for decision Makers, Catalogue Approach

UNIT-III CYBER SECURITY POLICIES

Cyber Governance Issues – Cyber user Issues –Cyber Conflict Issues –Cyber Infrastructure Issues - Cyber security’s role in National Security

UNIT-IV INTRODUCTION TO CYBER FORENSICS

Cyber Forensics, Computer Forensics and Investigations as a Profession- Understanding Computer Forensics-Preparing for Computer Investigations, Understanding Computer Investigations-Preparing a Computer Investigation-Taking a Systematic Approach-Procedures for Corporate High-Tech Investigations-Understanding Data Recovery Workstations and Software,

Data Acquisition-Understanding Storage Formats for Digital Evidence-Determining the Best Acquisition Method-Validating Data Acquisitions-Performing RAID Data Acquisitions

UNIT – V CYBER FORENSICS INVESTIGATION AND LAW

The Complex World of Corporate CyberForensics Investigations, Investigating Large-Scale Data Breach Cases, Insider Threat Investigations, Accounting Forensics, Cyber Law and Crime

Course Outcomes:

After completion of the course the student will be able to

1. Understand the importance of cyber security
2. Obtain knowledge on objectives of cyber security
3. Become familiar with cyber security policies
4. Explain the cyber forensics investigation process
5. Understand the laws of cyber forensics

Text Books:

1. Cyber Security Policy Guidebook, Jennifer L. Bayuk, Jason Healey, Paul Rohmeyer and Marcus Sachs John Wiley & Sons, Kindle Edition, 2012. References
2. Cyber Forensics, Albert J Marcella, Jr. Doug Menendez, AuerbachPublicaitons, Second Edition

Reference Books:

1. Cyber Security and Cyberwar, P.W.Singer and Allan Friedman Oxford University Press, Kindle Edition, 2014.
2. Cyber Security Essentials, James Graham, Ryan Olson and Rick Howard , CRC Press, Kindle Edition, 2014.
3. Network Security and Cryptography, Bernard Menezes , Cengage Learning

Mode of Evaluation: Assignments, Internal Mid Examination, External End Examination

Discipline Elective - IV

MCA III Year I Semester

18MCAP412 ADVANCED DATA STRUCTURES AND ALGORITHMS

L	T	P	C
3	0	0	3

Course Prerequisites:

Any Programming Language, Data Structures and Algorithms

Course Description:

This course helps the students to understand the concepts and applications of data structures. It gives an in depth working mechanism of various data structure algorithms

Course Objectives:

Students will be able to

1. Understand and the usage of algorithms in computing
2. Learn and use of trees and graphs in data structures and its operations
3. Learn the usage of graphs and its applications.
4. Select and design data structures and algorithms that is appropriate for problems.
5. Study about NP Completeness of problems.

UNIT- I ROLE OF ALGORITHMS IN COMPUTING

Algorithms, Algorithms as a Technology Insertion Sort, Analysing Algorithms , Designing Algorithms, Growth of Functions: Asymptotic Notation , Standard Notations and Common Functions-, Recurrences: The Substitution Method , The Recursion-Tree Method

UNIT -II TREES AND GRAPHS

Basics of trees and binary trees, Representation of trees and Binary trees, Binary tree Traversals, Threaded binary trees, Graphs, representation and traversals. Binary Search Trees, AVL Trees and B Trees - Binary Search Trees: Definition, Operations and applications. AVL Trees: Definition, Operations and applications. B Trees: Definition, Operations and applications.

UNIT -III RED- BLACK TREES, SPLAY TREES AND HASHING

Red-Black Trees, Splay Trees and their applications, Hash Tables, Hash Functions, Collision Resolution Techniques, Double Hashing, File Organizations

UNIT-IV ALGORITHM DESIGN TECHNIQUES

Dynamic Programming: Matrix-Chain Multiplication, Elements of Dynamic Programming, Longest Common Subsequence, Greedy Algorithms: An Activity-Selection Problem, Elements of the Greedy Strategy, Huffman Codes

UNIT-V NP COMPLETE AND NP HARD

NP-Completeness: Polynomial Time, Polynomial-Time Verification, NP-Hard Graph Problems, NP- Completeness and Reducibility, NP-Completeness Proofs, NP-Complete Problems

Course Outcomes:

After the completion of the course the students will be able to:

1. Identify and analyse the data structures and algorithms to solve computing problems
2. Design algorithms using trees and graph structure for solving various algorithms
3. Implement various trees structures to solve real-life problems
4. Apply suitable design strategy for problem solving
5. Identify and understand the problems belongs to NP- HARD and NP- COMPLETE

Text Books:

1. Fundamentals of Computer Algorithms by Ellis Horowitz, Sartaj Sahni andSanguthevar Rajasekaran, 2nd edition, University Press.
2. Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman, —Data Structures and Algorithms, Pearson Education, Reprint 2006

Reference Books:

1. Design and Analysis of Algorithms, First Edition,S.SridharOxford University Press. 2014
2. Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, , Third Edition, Prentice-Hall, 2011.

Mode of Evaluation: Assignments, Internal Mid Examination, External Mid Examination

MCA III Year I Semester

18MCAP413

.NET FRAMEWORK AND C#

L	T	P	C
3	0	0	3

Course Prerequisite:

Basic Programming Knowledge like C/C++

Course Description:

The course is designed to provide complete knowledge of C# language. Students will be able to develop logics which will help them to create programs, applications in C#. Microsoft has developed C# with features of popular languages to develop different types of .net applications. It has simplicity of Java and power of C++.

Course Objectives:

Students will be able to:

1. Get complete knowledge of MS.NET Framework and its internals.
2. Learn the basing programming concepts using C#
3. Build strong concepts of OOP's and implement the same in C#.
4. Learn to identify and usage of various libraries in C#
5. Learn features required to develop a web application.

UNIT -I THE .NET FRAMEWORK

Introduction, The Origin of .Net Technology, Common Language Runtime (CLR), Common Type System (CTS), Common Language Specification (CLS), Microsoft Intermediate Language (MSIL), Just-In -Time Compilation, Framework Base Classes and Library, Introduction to Windows Form

UNIT- II FUNDAMENTALS OF C- SHARP LANGUAGE (C#)

Data Types, Identifiers, Variables, Constants, Literals, Conditionals and Loops: Enumerations and Data Types Conversion, Arrays and Dynamic Arrays, Operators Decision Making statements, select case, Switch and choose statement, Looping Statements. Procedures, Scope and Exception Handling

UNIT-III OBJECT ORIENTED PROGRAMMING

Classes and Objects, Fields, properties, methods, Abstraction, Encapsulation Inheritance, Polymorphism, Overloading, Overriding, Shadowing, Constructors and Destructors, Interfaces.

UNIT -IV C# USING LIBRARIES

Multi-Threading, Networking and sockets, Managing Console I/O Operations, File Handling, Error Handling. Delegates and Events

UNIT -V ADVANCED FEATURES USING C#

Web Services, Window Services, Asp.net Web Form Controls, ADO.Net. Distributed Application in C#, Unsafe Mode, Graphical Device interface with C#.

Course Outcomes:

After Completion of the Course, Students will be able to:

1. Use .NET framework and the technologies that constitute the framework.
2. Implement the basic programming concepts using C#
3. Implement the object oriented programming skills in C#
4. Identify and use various libraries in C#
5. Design and Develop a web based application.

Text Books:

1. Herbert Schildt (2009), C# 3.0: The Complete Reference, McGraw-Hill, New Delhi
2. Wiley, "Beginning Visual C# 2008", Wrox
3. Balagurusamy, "Programming with C#", (TMH)
4. ShibiParikkar, "C# with .Net Framework", Firewall Media.

Reference Books:

1. Jesse Liberty (2002), Programming C#, Second edition, O'Reilly Media Inc, Cambridge, USA
2. Paul Deitel, Harvey Deitel (2011), C# 2010 For Programmers, Deitel Developer Series, Fourth Edition, Pearson Education, New Delhi.
3. Fergal Grimes, "Microsoft .Net for Programmers". (SPI)
4. Mark Michaelis, "Essential C# 3.0: For .NET Framework 3.5, 2/e, Pearson Education

Mode of Evaluation: Assignments, Internal Mid Examination, External End Examination

MCA III Year III Semester

18MCAP414

INTERNET OF THINGS

L T P C

Course Prerequisite:

3 0 0 3

Computer Networks, Python and C Programming

Course Description:

This course introduces the Machine to Machine Communication, Sensors, Actuators, IoT Architecture and IoT Protocols. It gives the exposure for building IoT with Arduino and Raspberry PI with Python Programming. It explains about IoT real world Constraints and various IoT applications in detail. It also includes Amazon Web Services for IoT.

Course Objectives:

Students will be able to:

1. Understand the fundamentals of Internet of Things
2. Study the IoT architecture and design models
3. Learn about the basics of IOT protocols
4. Build a small low cost embedded system using Raspberry Pi.
5. Study the concept of Internet of Things in the real world scenario.

UNIT-I INTRODUCTION TO IoT

Internet of Things - Physical Design- Logical Design- IoT Enabling Technologies - IoT Levels & Deployment Templates - Domain Specific IoTs - IoT and M2M - IoT System Management with NETCONF-YANG- IoT Platforms Design Methodology

UNIT-II IoT ARCHITECTURE

M2M high-level ETSI architecture - IETF architecture for IoT - OGC architecture - IoT reference model - Domain model - information model - functional model - communication model - IoT reference architecture

UNIT -III IoT PROTOCOLS

Protocol Standardization for IoT – Efforts – M2M and WSN Protocols – SCADA and RFID Protocols – Unified Data Standards – Protocols – IEEE 802.15.4 – BACNet Protocol – Modbus– Zigbee Architecture – Network layer – 6LowPAN - CoAP – Security

UNIT- IV BUILDING IoT WITH RASPBERRY PI & ARDUINO

Building IOT with RASPBERRY PI- IoT Systems - Logical Design using Python – IoT Physical Devices & Endpoints - IoT Device -Building blocks -Raspberry Pi -Board - Linux on Raspberry Pi - Raspberry Pi Interfaces -Programming Raspberry Pi with Python - Other IoT Platforms - Arduino.

UNIT- V CASE STUDIES AND REAL-WORLD APPLICATIONS (12)

Real world design constraints - Applications - Asset management, Industrial automation, smart grid, Commercial building automation, Smart cities - participatory sensing - Data Analytics for IoT – Software & Management Tools for IoT Cloud Storage Models & Communication APIs - Cloud for IoT - Amazon Web Services for IoT.

Course Outcomes:

After Completion of the Course, Students will be able to:

1. Classify the necessary fundamental concepts of IoT
2. Develop the IoT architecture and design models
3. Identify the usage of various IoT protocols
4. Analyze low cost embedded system using Raspberry Pi
5. Analyze applications of IoT in real time scenario

Text Books:

1. Arshdeep Bahga, Vijay Madiseti, “Internet of Things – A hands-on approach”, Universities Press, 2015
2. Dieter Uckelmann, Mark Harrison, Michahelles, Florian (Eds), “Architecting the Internet of Things”, Springer, 2011.

Reference Books:

1. Jan Hoeller, Vlasios Tsiatsis, Catherine Mulligan, Stamatis, Karnouskos, Stefan Avesand. David Boyle, "From Machine-to-Machine to the Internet of Things - Introduction to a New Age of Intelligence", Elsevier, 2014.
2. Honbo Zhou, “The Internet of Things in the Cloud: A Middleware Perspective”, CRC Press, 2012.
3. Olivier Hersent, David Boswarthick, Omar Elloumi, “The Internet of Things – Key applications and Protocols”, Wiley, 2012
4. The DevOps Handbook by Gene Kim, Jez Humble, Patrick Debois and John Willis.
5. Agile Software Development, Principles, Patterns, and Practices by Robert C. Martin
6. Effective DevOps, by Jennifer Davis, Ryn Daniels.
7. DevOps for Web Development by MiteshSoni.

Mode of Evaluation: Assignments, Internal Mid Examination, External End Examination

18MCAP415 ADVANCED JAVA PROGRAMMING

L	T	P	C
3	0	0	3

Course Prerequisite: Basic HTML, CSS Concepts, OOPs, Basic of SQL.

Course Description:

This course enables students to have strong fundamentals in web programming using Servlets, JSP and Spring Framework. This course helps students to understand various concepts of web page designing, and to establish connection between the Client and Server with the backend database. This course helps the students to develop dynamic websites for different applications.

Course Objectives:

Students will be able to

1. Construct Database Queries and understand the mechanism of JDBC
2. Learn the concepts of Servlets
3. Know about web servers and Java Server Page
4. Understand how to develop applications using JSP
5. Gain knowledge on Spring Core Framework

UNIT -I JAVA DATABASE CONNECTIVITY

JDBC Fundamentals, Database server, Database Client, working with Oracle Database, working with MySQL Database, Stages in a JDBC program, Types of JDBC Drivers, Retrieving Data from MySQL Database, Retrieving Data from MS Access Database, Improving the Performance of a JDBC Program, Types of Result Sets, storing a file into Database, Retrieving a File from the Database, Types of JDBC Drivers.

UNIT-II SERVLETS

Introduction to Servlets: Lifecycle of a Servlet, JSDK, The Servlet API, The javax.servlet Package, Reading Servlet parameters, Reading Initialization parameters. The javax.servlet HTTP package, Handling Http Request & Responses, Using Cookies-Session Tracking, Security Issues

UNIT- III WEB SERVERS & INTRODUCTION TO JSP

The Problem with Servlet. Introduction to Java Server Pages(JSP), Advantages of JSP, Life Cycle of JSP , JSP Processing. JSP Application Design with MVC Setting Up and JSP Environment: Installing the Java Software Development Kit, Tomcat Server & Testing Tomcat

UNIT-IV JSP APPLICATION DEVELOPMENT

Components of a JSP: Expressions, Scriptlets, Comments, Declaratives, Directives, (Page, Include, Taglib) Implicit Objects, JSTL, JSP Standard Actions (usebean, setproperty, getproperty, param)

UNIT-V JAVA SPRING FRAMEWORK

Spring Introduction: What is Spring-It's Features, How Spring Fits in to Enterprise Edition,

Spring Beans: What is Spring Bean-Bean Scope-Bean Lifecycle, IOC Containers :Core Container- J2EE Container-Web Container, Dependency Injection -Setter DI and Constructor DI,

Auto wiring: ByType- ByName

Course Outcomes:

After Completion of the Course the students will be able to

1. Execute SQL Queries and connect with different databases using JDBC.
2. Write programs in Java for Client/Server Communication and connect with specific database for data manipulation
3. Execute programs on basic concepts of JSP
4. Build applications using JSP and deploy the project using Tomcat Server
5. Work on concepts of Spring

Text Books:

1. J2EE – The Complete Reference- Jim Keogh, Tata McGraw Hill, 2007
2. Mahesh P. Matha JSP and Servlets, 1st Edition, PHI, 2013.
3. Java Servlet Programming, Second Edition by Jason Hunter, William Crawford, O'Reilly
4. Spring in Action –Craig walls 4th edition, Dreamtech publisher

Reference Books:

1. The J2EE Tutorial – Stephanie Bodoff et al, 2nd Edition, Pearson Education, 2004.
2. Java Server Pages –Hans Bergsten, SPDO"Reilly ,2008
3. Learning Spring 5.0: Build enterprisegrade applications using Spring MVC, ORM Hibernate and RESTful APIs – Tejaswini Mandar Jog, Packt Publisher
4. Spring in Action –Craig walls 4th edition, Dreamtech publisher

Mode of Evaluation: Assignments, Internal Mid Examinations, External End Examination.

MCA III Year I Semester

18MCAP509

ADVANCED DATA STRUCTURES AND ALGORITHMS LABORATORY

L	T	P	C
0	0	3	1.5

Course Prerequisites:

Any Programming Language, Data Structures and Algorithms

Course Description:

This course helps the students understand the concepts and applicability of available programming paradigm to solve the advanced data structure related problems

Course Objectives:

Students will be able to:

1. Acquire the knowledge of using advanced tree structures.
2. Learn the usage of heap structures.
3. Usage of graph structures and trees.

List of Experiments

1. Implementation of Merge Sort and Quick Sort-Analysis
2. Multi Stack
3. Double ended Queues and Circular Queues
4. Implementation of a Binary Search Tree
5. Min Heap and Max Heap
6. AVL Trees
7. B- Trees
8. Red-Black Tree Implementation
9. Heap Implementation
10. Fibonacci Heap Implementation
11. Graph Traversals
12. Spanning Tree Implementation
13. Shortest Path Algorithms (Dijkstra's algorithm, Bellmann Ford Algorithm)

14. Implementation of Matrix Chain Multiplication
15. Activity Selection and Huffman Coding Implementation.

Course Outcomes:

After completion of the course the Students will be able to:

1. Design and implement basic and advanced data structures extensively.
2. Develop algorithms using tree and graph structures
3. Design and develop efficient algorithms with minimum complexity using design techniques

Text Books:

1. Fundamentals of Computer Algorithms by Ellis Horowitz, Sartaj Sahni and Sanguthevar Rajasekaran, 2nd edition, University Press.
2. Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman, —Data Structures and Algorithms, Pearson Education, Reprint 2006

Reference Books:

1. Design and Analysis of Algorithms, First Edition, S.Sridhar Oxford University Press. 2014
2. Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, , Third Edition, Prentice-Hall, 2011.

Mode of Evaluation: Continuous cumulative evaluation of the Lab Experiments, Record, Viva Voce and External Examinations

18MCAP510 .NET FRAMEWORK AND C# LABORATORY

L	T	P	C
0	0	3	1.5

Course Prerequisite:

Basic Programming Knowledge

Course Description:

The course is designed to develop implementation knowledge to the students in the .NET environment using C# language. Students will be able to develop windows and console applications using various object-oriented concepts and advance features like multithreading, File input-output and Error Handling etc.

Course Objectives:

Student will be able to

1. Understand use of C# basics, OOPS Concepts like Encapsulation, Inheritance, Polymorphism
2. Implement and develop and Windows Applications with C#.
3. Implement Multi-Threading, Sockets Programming, File Handling, Error Handling etc.

List of Experiments:

1. Write a C# Sharp program to read age of a candidate and determine whether it is eligible for casting his/her own vote.
2. Write a C# Sharp program to accept a coordinate point in an XY coordinate system and determine in which quadrant the coordinate point
3. Write a C# Sharp program to calculate root of Quadratic Equation.
4. Write a program in C# Sharp to calculate and print the Electricity bill of a given customer. The customer id., name and unit consumed by the user should be taken from the keyboard and display the total amount to pay to the customer. The charge are as follow :

Unit	Charge/unit
upto 199	@1.20
200 and above but less than 400	@1.50
400 and above but less than 600	@1.80
600 and above	@2.00

If bill exceeds Rs. 400 then a surcharge of 15% will be charged and the minimum bill should be of Rs. 100/-
5. Write a program in C# Sharp to make such a pattern like a pyramid with numbers increased by 1.

The pattern is as follows:

1
2 3
4 5 6
7 8 9 10

6. Write a C# program to implement Encapsulation concept using public, private and protected access specifiers
7. Write a C# program to implement function overloading and function overriding
8. Write a C# program to implement Multiple Interfaces having same method name in .NET environment
9. Write a C# program to sort the elements in the following List elements using console application
 - a. 1,5,6,2,4,3
 - b. A,I,G,B,H,F,C,
10. Write a program to simple calculator using windows application.
11. Write a windows forms application for creating a login form. After the successful login user can perform addition, multiplication, subtraction and division by the user. Controls can be added to the Windows forms via the Toolbox in Visual Studio. Controls such as labels, checkboxes, radio buttons, etc. can be added to the form via the toolbox
12. Write a C# program to implement the concept of Single threaded model
13. Write a C# program to implement the concept of Multi-threaded model
14. Write a C# program to implement Socket Programming. In this process, create a socket and setup a listener server node that starts listening to any messages coming to it via the predefined IP and protocol. Create a client application that will send message to the listener server and read it.
15. Write a program in C# to create and copy the file to another name and display the content.
16. Write a program in C# program to implement the concept of Exception Handling in the following case:

Let's assume that we are calculating the average grades for students. Further, we'll assume that for a particular subject not a single student sat for the exam. In this case, the divisor would become zero. If this situation occurs and there is no handler, the program would crash. However, developers usually foresee this possibility and check for zero divisors. A developer would enter code to handle the error by displaying an error message and bringing the program to a logical end.

Course Outcomes:

After Completion of the Course, Students will be able to

1. Write Program in .NET framework using C# language both in basic and advanced levels.
2. Develop real time Windows based and console-based Applications
3. Build careers in software industries

Text Books:

1. Herbert Schildt (2009), C# 3.0: The Complete Reference, McGraw-Hill, New Delhi
2. Wiley," Beginning Visual C# 2008", Wrox
3. Balagurusamy," Programming with C#", (TMH)
4. ShibiParikkar, " C# with .Net Frame Work" , Firewall Media.

Reference Books:

1. Jesse Liberty (2002), Programming C#, Second edition, O'Reilly Media Inc, Cambridge, USA
2. Paul Deitel, Harvey Deitel (2011), C# 2010 For Programmers, Deitel Developer Series, Fourth Edition, Pearson Education, New Delhi.
3. Fergal Grimes," Microsoft .Net for Programmers". (SPI)
4. Mark Michaelis, "Essential C# 3.0: For .NET Framework 3.5, 2/e, Pearson Education

Mode of Evaluation: Continuous Cumulative Evaluation, Observation, Record, Viva Voce and External Lab Examination

18MCAP511 INTERNET OF THINGS LABORATORY

L T P C
0 0315

Course Prerequisite: C Programming

Course Description:

The student should have hands on experience in using various sensors like temperature, humidity, smoke, light, etc. and should be able to use control web camera, network, and relays connected to the Pi.

Course Objectives:

Students will be able to :

1. Implement the Linux commands in command terminal window
2. Run some python programs on Rasperry Pi
3. Implement the functionalities of Actuators using python program on Rasperry Pi and Use the sensors & actuators for IoT applications

LIST OF EXPERIMENTS

1. Start Rasperry Pi and try various Linux commands in command terminal window: ls, cd, touch, mv, rm, man, mkdir, rmdir, tar, gzip, cat, more, less, ps, sudo, cron, chown, chgrp, ping etc.
2. Read your name and print Hello message with name
3. Read two numbers and print their sum, difference, product and division.
4. Word and character count of a given string
5. Area of a given shape (rectangle, triangle and circle) reading shape and appropriate values from standard input
6. Print name 'n' times, where name and n are read from standard input, using for and while loops.
7. Print current time for 10 times with an interval of 10 seconds.
8. Read a file line by line and print the word count of each line.
9. Light an LED through Python program
10. Get input from two switches and switch on corresponding LEDs
11. Flash an LED at a given on time and off time cycle, where the two times are taken from a file.
12. Flash an LED based on cron output (acts as an alarm)
13. Switch on a relay at a given time using cron, where the relay's contact terminals are connected to a load.

14. Access an image through a Pi web cam.
15. Control a light source using web page.
16. Implement an intruder system that sends an alert to the given email.
17. Get the status of a bulb at a remote place (on the LAN) through web.
18. Get an alarm from a remote area (through LAN) if smoke is detected.

Course Outcomes:

After completion of the above courses the student will be able to

1. Execute the various Linux commands in command terminal window
2. Gain real time IoT experience by implementing python programs on Rasperry Pi platform
3. Implement the functionalities of various Actuators on Rasperry Pi and integrate the sensors & actuators in Rasperry Pi using Python Programming

Reference Books:

1. Dieter Uckelmann, Mark Harrison, Michahelles, Florian (Eds), “Architecting the Internet of Things”, Springer, 2011.
2. Jan Ho" ller, Vlasios Tsiatsis , Catherine Mulligan, Stamatis , Karnouskos, Stefan Avesand. David Boyle, "From Machine-to-Machine to the Internet of Things - Introduction to aNew Age of Intelligence", Elsevier, 2014.
3. Olivier Hersent, David Boswarthick, Omar Elloumi , “The Internet of Things – Key applications and Protocols”, Wiley, 2012

Mode of Evaluation: Continuous Cumulative evaluation of the lab experiments, Record, Viva voce and External Lab Examination.

18MCAP512 ADVANCED JAVA PROGRAMMING LABORATORY

L	T	P	C
0	0	3	1.5

Prerequisite: Practical knowledge on OOPs concepts.

Course Description:

In this course, students gain extensive hands-on experience on Java programming. Students learn to create robust web based applications using code reusability with, applications of I/O streams and GUI implementation, Servlets and JSP.

Course Objectives:

Students will be able to :

1. Apply SQL DDL and DML commands to create tables, select, insert, update and delete
2. Develop Web Based Applications using Servlets and JSP
3. Develop Simple Applications using Spring

List of Experiments

1. Write a java program to connect oracle database and perform basic SQL operations.
2. Write a java program to connect with MS-Access database and perform basic Query operations.
3. Write a java program to create Callable Statement to call the stored procedure and retrieve the result from oracle Server.
4. Write a java program to connect with MySQL database and perform basic MySQL operations.
5. Install Tomcat Web Server and Apache
6. Design and create registration form with all necessary fields using Servlets and JDBC to demonstrate the storage of data in the database.
7. Develop a web application to process student information.
8. Write a JSP program for arithmetic calculations
9. Write a JSP program to create checkboxes
10. Develop a web application using Servlet, JSP and JDBC
11. Implement Spring Dependency Injection via setter and constructor
12. Develop simple application using spring

Course Outcomes:

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

1. Differentiate between DDL and DML
2. Implement Database connectivity and web based applications
3. Develop Simple Application using Spring

Reference Books:

1. The J2EE Tutorial – Stephanie Bodoff et al, 2nd Edition, Pearson Education, 2004.

Java Servlet Programming, Second Edition by Jason Hunter, William Crawford, O'Reilly

Mode of Evaluation: Continuous Cumulative evaluation of the lab experiments, Record, Viva voce and External Lab Examination

SOFTWARE LAB – I

MCA I Year II Semester

18MCAP513 DATA ANALYSIS WITH SPSS LABORATORY

L	T	P	C
0	0	3	1.5

Course Prerequisite: None

Course Description: This course is intended for students with no prior training in statistical methods using SPSS. It introduces fundamental concepts and methods of statistics and mathematics, and specifically develops students' knowledge and skills in probability distributions and collecting, describing, summarizing, and presenting statistical data using SPSS.

Course Objectives:

Students will be able to

1. Familiarize the student with data presentation techniques.
2. Equip students with the methods of social science statistics.
3. Develop application of statistics in accordance with social research objectives.

LIST OF EXPERIMENTS

GRAPHS & DIAGRAMS OF STATISTICAL DATA

1. Frequency Distributions
2. Bar Diagrams, Pie-Diagram, and Histograms.

DESCRIPTIVE STATISTICS

3. Computing Measures of central tendencies
4. Computing Measures of dispersion
5. Computing Measure of Skewness
6. Computing Correlation Coefficient for Bivariate Distribution.
7. Computing Simple Linear Regression.
8. Computing Multiple Linear Regression

TESTS OF SIGNIFICANCE

9. t-Test for Single Mean
10. t-Test for Difference of Means

11. Paired t-Test
12. Chi-Square Test for Goodness of Fit
13. Chi-Square Test for Independence of Attributes
14. F-Test for Equality of two Population Variances
15. Analysis of Variance

Course Outcomes:

After completion of the above courses the student will be able to

1. The Students will be able to use MS Excel and other Statistical Software to employ statistical techniques for data analysis.
2. Students will be able to analyze business data and work together to statistically analyze and infer from the data.
3. Understanding the layout and interface of SPSS.

Text Books:

1. Practical Statistics, Shiv Kumar, Sultan Chand & Co.
2. Statistical Methods, S.P.Gupta and M.P.Gupta, Sultan Chand &Co.

Reference Books:

1. Fundamental of Applied Statistics, S.C.Gupta&V.K.Kapoor, Sultan Chand & Co.
2. Statistical Methods, G.W. Snedcor, W.G. Cochran: 6 Edn, East West Press.
3. Fundamentals of Mathematical Statistics, S.C.Gupta and V.K.Kapoor, Sultan Chand & Sons New Delhi.

Mode of Evaluation: Continuous Cumulative evaluation of the lab experiments, Record, Viva voce and External Lab Examination

Course Prerequisite: Basic Computer Science, Basic Mathematics, Arithmetic and Reasoning Knowledge requires

Course Description:

This course describes show to use R for effective data analysis. The course covers practical aspects of statistical computing in the data science which includes programming in R, reading data into R, accessing R packages, writing R functions, debugging, and establishing R code. Various statistical computations on data is introduced. In addition to this, drawing Graph and Chart through R has also been included.

Course Objectives:

Student will be able to

1. Understand how to write program for data analysis
2. Do the basics of R programming concepts like R-Basic Syntax, R- Decision Making, R- Objects (i.e. Vector, List, Matrix, Array, Factor, Data Frame etc.)
3. Get an idea to plot various types of Charts and Graphs for data analyses.

List of Experiments:

1. Write R- Script to determine the classes of the following:
a) FALSE b) 48.59 c) "TRUE" d) 'T' e) "T" f) 50+9i g) 485
h) 10L
2. Write a R-Script to create vectors with the following specification

Vector Name	Vector Elements
apple	red, green, yellow
ColorImage	Red, Green, Blue
Value	20,10,13
Value2	20L, 10L, 13L

Create and display the vectors and their corresponding classes.
3. (i) Create two vectors of 6 numeric elements each then perform the following:
a) Addition b) Multiplication c) Subtraction d) Division

(ii) Create two vectors of 6 and 2 numeric elements respectively then perform the following:
a) Addition b) Multiplication c) Subtraction d) Division

(iii) Create vector with elements from 5 to 9 incrementing by 0.4 and calculate its length.

(iv) Concatenate the following vectors

Vector Name	Vector Elements
Fruits	Apple, Orange, Banana
Vegetables	Cabbage, Potato, Tomato

4. (i) Create a list containing 'vector of days', 'vector of numbers' and a logical value.
(ii) Give name to the vectors available in the list.
(iii) Print the list.
(iv) Access the particular vector of the list.
5. Create a matrix taking a vector of numbers as input with 4 rows and 3 columns.
(a) Print the elements row wise and column wise
(b) Give the column and rows names
(c) Print element of the matrix at position of 3rd row and 2nd column
(d) Print only 2nd column elements
6. Create two 3*4 matrix and perform the following matrix operations
(a) Addition (b) Subtraction (c) Multiplication (d) Division
7. Write a R-Program to create a matrix taking a given vector of numbers as input. Display the matrix.
8. Write a R-Program to find row and column index of maximum and minimum value in a given matrix.
9. Write a R program to convert a given matrix to a 1-dimensional array.
10. Write a R program to create an array of two 3x3 matrices each with 3 rows and 3 columns from two given two vectors. Print the second row of the second matrix of the array and the element in the 3rd row and 3rd column of the 1st matrix.
11. Write a R program to combine three arrays so that the first row of the first array is followed by the first row of the second array and then first row of the third array.
12. Write a R program to create a two-dimensional 5x3 array of sequence of even integers greater than 50.
13. Write a R program to find the levels of factor of a given vector.
14. Write a R program to change the first level of a factor with another level of a given factor.
15. Write a R program to convert a given pH levels of soil to an ordered factor.
16. Write a R program to extract the five of the levels of factor created from a random sample from the LETTERS (Part of the base R distribution.)
17. Write a R program to create a factor corresponding to height of women data set, which contains height and weights for a sample of women.
18. Write a R program to create a data frame from four given vectors.
19. Write a R program to get the structure of a given data frame.
20. Write a R program to get the statistical summary and nature of the data of a given data frame.
21. Write a R program to extract 3rd and 5th rows with 1st and 3rd columns from a given data frame.

22. Write a R program to add new column(s), new row(s), drop new column(s), drop new row(s), in a given data frame.
23. Write a R program to sort a given data frame by multiple column(s).
24. Write a R program to create inner, outer, left, right join(merge) from given two data frames.
25. Write a R program to save the information of a data frame in a file and display the information of the file.
26. Write a R program to call the (built-in) dataset airquality. a) Check whether it is a data frame or not? Order the entire data frame by the first and second column. b) Remove the variables 'Solar.R' and 'Wind' and display the data frame.
27. Write a R program to compute mean, median, mode of the given vector.
28. Write a R program to Create Relationship Model between two vectors.
29. Write a R program to Create and plot the binomial distribution of given set of data.
30. Write a R program toplot 2-D and 3-D pie-chart with the title and color using input vector data and labels. Display Slice Percentages and Chart Legend.

Course Outcomes:

1. Students can be able understand and analyze the basic problems related to Data Science
2. Problem solving skill of students can be enhanced.
3. Decision making knowledge can be enhanced for solving real time problems.

Text Books:

1. The Art of R Programming - Norman Matloff
2. Hands-On Programming with R - Garrett Golemund

Reference Books:

1. Exploratory Data Analysis with R, Roger D Peng.
2. Data Visualization: A practical introduction, by Kieran Healy.

Mode of Evaluation: Continuous Cumulative evaluation of the lab experiments, Record, Viva voce and External Lab Examination

SOFTWARE LAB - II

MCA II Year I Semester
18MCAP515

C++ LABORATORY

L T P C
0 0 3 1.5

Course Prerequisite: C Programming

Course Description:

This course is an introduction to the theory and practice of computer programming, the emphasis of this course is on techniques of program development within the structure and object-oriented paradigm. Topics include C++ program basics and Object Oriented Concepts.

Course Objectives:

Student will be able to

1. Learn C++ Programming Language
2. Write C++ programs for real world problems using simple and compound data types
3. Make good programming style, standards and practices during program development

List of Experiments:

Conditional constructs:

1. Write a C++ program to check whether the given number is even or odd.
2. Write a C++ program to calculate the grade points when the user enter a grade using switch. Case constructs.
3. Write a C++ program to swap two given integers without using third variable
4. Write a C++ program to find whether the entered character is a capital letter or small letter or a digit or a special character.
5. Develop a C++ program to find the largest of 4 numbers without using array
6. Write a C++ program to find whether the given person is eligible for vote or not by getting the age as the input

Iterative Constructs:

7. Write a C++ program to find if the given number is perfect number or not.
(Perfect number ex: $6 = 1+2+3= 6$)
8. Write a C++ program to find the last prime number before the number which is given as input.
Ex: Input 10 - Last prime number: 7. Input 14-Last prime number 13. Input 15-Last prime number 13
9. Write a C++ program to convert the decimal integer into binary number
10. Write a C++ program to find the LCM and HCF of the given numbers

11. Write a C++ program to find whether the given number is a palindrome or not without using arrays

Playing with arrays:

12. Write a C++ program to perform the matrix operations (Addition, Subtraction and Multiplication using arrays)
13. Write a C++ program to find whether the given string and the given number is a palindrome or not using arrays (Palindrome example: Madam I'm Adam)
14. Write a C++ program to find whether the given number is an Armstrong number or not
15. Write a C++ program to sort the given array using any one of the sorting techniques
16. Develop a C++ program to search the given element in an array using Binary search and Linear search using recursion.

Object oriented paradigm:

17. Write a C++ program to perform the arithmetic operations using classes and objects
18. Write a C++ program to demonstrate the usage of friend function
19. Write a C++ program to find the square and cube of the given number using hierarchical inheritance
20. Develop a C++ program to read the employee information and print with PF details using hierarchical inheritance
21. Write a C++ program to find the areas of different shapes such as square, rectangle, triangle using inheritance

Files and its use:

22. Develop a C++ program to write a data into a text file
23. Write a C++ program to append data into a file with the existing data
24. Demonstrate the file manipulation operations such as deletion and reading data from a file using C++

Course Outcomes:

After completion of the above course the student will be able to

1. Write, compile and debug programs in C language.
2. Choose programming components that efficiently solve computing problems in real-world.
3. Design programs involving different data types, control structures, functions and files.

Text Books:

1. The C++ Programming Language, 4th edition by Bjarne Stroustrup

Reference Books:

1. C++ Primer (5th Edition), by Stanley B. Lippman, Josée Lajoie, Barbara E. Moo.

Mode of Evaluation: Continuous Cumulative evaluation of the lab experiments, Record, Viva voce and External Lab Examination

MCA II Year I Semester

18MCAP516

NETWORK SIMULATOR LABORATORY

L T P C
0 0 3 1.5

Course Prerequisite: Networking Concepts and C++ Programming

Course Description:

This course provides to get exposure of the most useful tool in Network research and development. Network Simulator tool deals with the design principles of wired and wireless network. It helps to study the performance of network protocols and it provides the platform to work in simulation environment.

Course Objectives:

Students will be able to:

1. Understand and design network topology using NS2
2. Understand and design wireless and wired network using NS2
3. Understand the scenario and study the performance of various network protocols through simulation.

LIST OF EXPERIMENTS

1. Installation of Network Simulator (NS-2) in Linux Environment
2. Basic TCL Scripting Language Commands
3. Initialization and Termination of TCL Script in NS-2
4. Setup a CBR over UDP connection.
5. Read a data from file and draw graph using XGRAPH program.
6. Process the data from Trace Files using AWK scripts.
7. Implement three nodes point-to-point network with duplex links between them. Set the queue size, vary the bandwidth and find the number of packets dropped.
8. Implement transmission of ping messages/trace route over a network topology consisting of 6
9. nodes and find the number of packets dropped due to congestion.
10. Implement an Ethernet LAN using n nodes and set multiple traffic nodes and plot congestion window for different source / destination.

11. Implement simple ESS and with transmitting nodes in wire-less LAN by simulation and determine the performance with respect to transmission of packets
12. Implement and study the performance of GSM on NS2/NS3 (Using MAC layer) or equivalent Environment.
13. Implement and study the performance of CDMA on NS2/NS3 (Using stack called Call net) or equivalent environment

Course Outcomes:

After completion of the above courses the student will be able to:

1. Learn the basic idea about open source network simulator NS2 and how to download, install and work with NS2 using TCL programming.
2. Defining the different agents and their applications like TCP, FTP over TCP, UDP, CBR and CBR over UDP etc.
3. Identifying and solving the installation error of NS2.

Reference Books:

1. Introduction to Network Simulator NS2, Ekram Hossain and TeerawatIssariyakul, Second Edition, Springer.
2. Computer Network Simulation Using NS2, Ajit Kumar Nayak, First Edition, CRC Press.

Mode of Evaluation: Continuous Cumulative evaluation of the lab experiments, Record, Viva voce and External Lab Examination