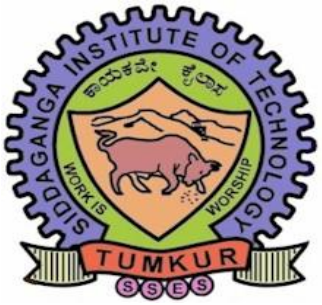
SCHEME & SYLLABUS

OF

I & II SEMESTERS

2019 Batch



Department of Master of Computer Applications
Siddaganga Institute of Technology
B.H. Road, Tumakuru

VISION STATEMENT

To effectively mould quality and responsible Computer Professionals, with a mind of service and spirituality for nurturing the technological competence.

MISSION STATEMENT

Imparting Quality Education to Students to make them professionals in their domain replete with IT and Computational Skills par Excellence

Program Educational Objectives

PEO 1 *Pursue career in computer applications domain by developing abilities that are in synchrony with changing needs of Industry or academia*

PEO 2 *Demonstrate professionalism when working with teams and align with ethical principles.*

PEO 3 *Engage in lifelong learning to upgrade the professional skills*

Program Outcomes:

The graduates of MCA will:

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

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

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

d. ***Conduct Investigations of Complex Computing Problems:*** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

e. ***Modern Tool Usage:*** Create, select, adapt and apply appropriate techniques, resources, and modern computing tools to complex computing activities, with an understanding of the limitations.

f. ***Professional Ethics:*** Understand and commit to professional ethics and cyber regulations, responsibilities, and norms of professional computing practice.

g. ***Life-long Learning:*** Recognize the need, and have the ability, to engage in independent learning for continual development as a computing professional.

h. ***Project management and finance****:* Demonstrate knowledge and understanding of t h e c o mp u t i n g and management principles and apply these to one's own work, as a member and leader in a team, to 16 manage projects and in multidisciplinary environments.

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

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

k. ***Individual and Team Work:*** Function effectively as an individual and as a member or leader in diverse teams and in multidisciplinary environments.

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

Distribution of subjects
Total Credits: 150

Course Type	Credits	Percentage of Total Credits
Mathematics	11	8.33
Professional Core (including practical)	76	57.60
Professional Electives	15	11.36
Humanity subjects	04	3.00
Seminars(2)	02	1.51
Projects (Major & Mini)	24	18.20

Board of Studies Committee, MCA: 2019-20

Sl. No	Category	Nomination of the Committee	Name of the Person
1.	Director Dept. of M C A	Chairperson	Dr.M.A.Jayaram
2.	Faculty members at different levels veering different specializations	Members	1.Dr. B.G.Premasudha, professor 2. Dr..Ashagowda Karegowda, Associate Professor 3. Mr. H.S.Vijayakumar, Assistant Professor 4. Mr. T.M.Kirankumar, Assistant Professor 5. Mr. C.Bhanuprakash, Assistant Professor 6.Dr.P.T.Bharathi. Asst. Prof 7.Dr.G.K.Prashanth, Asst. Prof.
3.	Subject experts from outside the college nominated by Academic Council	Member	Dr.C.N.Ravikumar Principal (formerly Prof. & Head, Dept. of CS&E, SJCE, Mysore) Mysore College of Engg. & Management, Mysore Dr. Sudhakar Chapram Associate Professor Dept. of CS&E, NIT Warangal, Telangana.
4.	Expert from outside college, nominated by VTU.	Member	Dr. Sujatha B.N. Professor & Head Dept. of MCA, BMSCE Bengaluru.
5.	Representative from Industry/Corporate sector.	Member	Dr.Gopinatha S. Magge Principal Infrastructure Architect Advanced Technical Services Siemens PLM Software Digital Factory Division, Product Lifecycle Management, Global Services Siemans Pvt. Ltd., Bengaluru.
6.	Postgraduate Meritorious alumnus nominated by Principal/Chairman	Member	Mr. Seshaphani, Manager Capgimini Pvt. Ltd., Bengaluru.
7.	Co-opted members Also our distinguished alumni.	Member	Mr. Naga Prasad Engineering Leader Altimetrik India Pvt., Ltd Bengaluru.

SCHEME OF TEACHING OF
I-II SEMESTER MCA PROGRAMME

I-Semester MCA

Sl No.	Course code	Title	Teaching Hours/week			Examination				Credits
			Theory	Practical	Tutorial	Duration in hours	SEE marks	CIE marks	Total marks	
1	1RMCA01	Foundations of Mathematics	3	-----	2	3	50	50	100	4
2	1RMCA02	Unix fundamentals & shell programming	3	---	2	3	50	50	100	4
3	1RMCA06	Object Oriented Programming with C++	3	------	2	3	50	50	100	4
4	1RMCA04	Professional Communication and Ethics	3	------ ---	-	3	50	50	100	3
5	1RMCA05	Fundamentals of computer Organization	3	-----	----	3	50	50	100	3
6	1RMCAL1	Unix Lab	--- --- -	3	----- -	3	50	50	100	1.5
7	1RMCAL3	OOPS With C++ Lab	--- ---	3	----- -	3	50	50	100	1.5
8	1RMD1	Business English Lab	--- --	---	-----	2	50	----	100	1
									Total credits	22
									Total Class hours /week	21
									Total Contact hours/week	29

II-Semester MCA

Sl No.	Course code	Title	Teaching Hours/week			Examination				Credits
			Theory	Practical	Tutorial	Duration in hours	SEE marks	CIE marks	Total marks	
1	2RMCA01	Web Technologies	3	-----	2	3	50	50	100	4
2	2RMCA02	Data Structures using C++	3	---	2	3	50	50	100	4
3	2RMCA03	Operations research	3	------	2	3	50	50	100	4
4	2RMCA04	Software Engineering	3	---------	---	3	50	50	100	3
5	2RMCA05	Computer oriented numerical methods	3	-----	----	3	50	50	100	3
6	2RMCAL1	CONM & OR lab	-------	2	------	3	50	50	100	1
7	2RMCAL2	Data Structures with C++ lab	------	3	------	3	50	50	100	1.5
8	2RMCAL3	Web Technologies Lab	-----	3	-----	3	50	50	100	1.5
Total credits										22
Total Class hours /week										21
Total Contact hours/week										29

Regular Lab. Work and writing lab records	2**5 mar**ks
Evaluation of open ended experiments	**10 marks**
Lab. Test and Viva-voce at the end of the	**15 marks**
TOTAL	**50 marks**

III Semester

Sl. No	Subject Code	Title	Hours/week			Credits
			L	T	Lab	
1.	3RMCA01	Analysis and Design of Algorithms	3	2	-----	4
2.	3RMCA02	RDBMS	3	2	---	4
3.	3RMCA03	Operating Systems	3	---	---	3
4.	3RMCA04	Java Programming	3	2	----	4
5.	3RMCAEX	Elective -I	3	----	---	3
6.	3RMCAL1	Mini Project-I	---	---	4	2
7.	3RMCAL2	RDBMS Lab	---	---	2	1
8.	3RMCAL3	Java Lab	---	---	2	1
9.	MC06	Soft Skills(Mandatory)	----	2	----	----
					Total Credits	22
					Total class hours/week	21
					Total Contact Hours/week	32

Elective-I		
1.	3RMCAE3	Management Information Systems and E commerce
2.	3RMCAE4	Software Project Management
3.	3RMCAE7	Computer Graphics

IV Semester MCA

Sl. No	Subject Code	Title	Hours/week			Credits
			L	T	Lab	
1.	4RMCA01	Computer Networks	3	---	-----	3
2.	4RMCA02	Programming with Python	3	2	---	4
3.	4RMCA03	Data Mining & Warehousing	3	2	------	4
4.	4RMCA04	Advanced Java Programming	3	2	------	4
5.	4RMCAEX	Elective-II	3	----	-----	3
6.	4RMCAL1	Python Lab	-------	------	2	1
7.	4RMCAL2	Data Mining Lab	------	------	2	1
8.	4RMCAL3	Advanced Java lab	-----	-----	2	1
9.	MCR05	Aptitude related Analytical Skills	----	2	-----	1
					Total Credits	22
					Total class hours/week	21
					Total Contact Hours/week	29

Elective-II		
1.	4RMCAE1	Intelligent Data Analytics
2.	4RMCAE2	Artificial Intelligence
3.	4RMCAE7	Principles of User Interface Design

V SEMESTER

Sl No.	Course code	Title	Teaching Hours/week			Examination				Credits
			Theory	Practical	Tutorial	Duration in hours	SEE marks	CIE marks	Total marks	
1	5RMCA01	Object-Oriented Modeling and Design Patterns	3	-----	2	3	50	50	100	4
2	5RMCA02	Mobile Application Development	3	---	2	3	50	50	100	4
3	5RMCAE1X	Elective-III	3	------	--	3	50	50	100	3
4	5RMCAE2X	Elective -IV	3	------	---	3	50	50	100	3
5	5RMCAE3X	Elective-V	3	-----	----	3	50	50	100	3
6	5RMCAL1	OOMD Lab	-------	3	------	3	50	50	100	1.5
7	5RMCAL2	Mobile Application Development Lab	------	3	------	3	50	50	100	1.5
8	5RMCAL3	Mini Project-II	-----	4	-----	3	50	50	100	2
									Total credits	22
									Total Class hours /week	19
									Total Contact hours/week	29

Elective-III (Thrust areas)	
5RMCAE12	Cloud Computing
5RMCAE13	Machine Learning
5RMCAE15	Data Science

Elective-IV(Software Engineering Stream)	
5RMCAE21	Software Testing
5RMCAE22	Software agents
5RMCAE23	Software Quality Management

Elective-V(Other allied areas)	
5RMCAE31	Big Data Analysis
5RMCAE32	System Simulation and Modeling
5RMCAE34	Digital Image Processing

SCHEME OF TEACHING OF VI SEMESTER MCA PROGRAMME

Sl No.	Course code	Title	Examination				Credits
			Duration in hours	SEE marks	CIE marks	Total marks	
1	6RMCA01	Major Project	----	50	50	100	20
2	6RMCAS1	Technical Seminar			50	50	02
							22

Detailed Syllabi for I semester

Foundations of Mathematics

Contact Hours/Week	:	(3+2)	Credits	:	04
Total Lecture Hours	:	39	CIE Marks	:	50
Total Tutorial Hours	:	13	SEE Marks	:	50
Course Code	:	1RMCA01			

Course Objectives:

This Course will enable students to:	
1.	Apply the concept of set theory to probability, deriving the equalities or inequalities using Mathematical Induction.
2.	Derive logical implications and equivalences using laws of logic, use of quantifiers and different ways of proving a given statement.
3.	Identify the different types of function and calculate composition and inverse of functions.
4.	Compute zero-one matrix, composition of relations and draw Hasse diagram.
5.	Explain the concept of groups, subgroup, Abelian group and derive Lagrange's theorem in groups.
6.	Identify types of graph, outline properties of graphs, describe when the graphs are said to be same even though the shapes are different (isomorphism) and apply to some practical problems like seven bridge problem, traveling sales man problem.
7.	Describe how to color the vertices/ edges of a graph, apply graph coloring in map coloring, describe what is a tree and its properties and apply the concept of trees in constructing optimal prefix codes.
8.	Determine the shortest path between two vertices, write algorithms for finding minimal spanning trees and apply the concepts in transport network

UNIT – I	12 Hours
Set Theory: Introduction, Definition and concepts, Representation of sets, Finite sets, Infinite sets (Definition), Set Operations: Union, Intersection, Addition theorem, difference, symmetric difference, D'Morgan's law, subsets, power sets, partitions sets, Mathematical Inductions, Permutations and Combinations.	

UNIT – II	10 Hours
Propositional Logic: Prepositions, logical operations, tautologies, contradictions, logical implication, logical equivalence, normal forms, theory of inference and deduction. Predicate calculus: predicates and quantifiers.	

UNIT – III	10 Hours
Functions: Introduction: Definitions and concepts, one to one, onto functions, invertible functions, Mathematical functions: Floor and ceiling functions, Integer and Absolute value functions, Remainder functions, Exponential functions, logarithmic functions, Sequences and Series: definitions, deference between sequences and series, to find nth term and sum of n terms, Recursive functions: definition and examples.	

UNIT – IV	10 Hours
Boolean Algebra: Introduction, basic definitions, duality, basic theorems, Boolean algebra and lattice, Representation theorem, Sum-of-product form for sets, Sum-of-products form for Boolean algebra.	

UNIT – V	10 Hours
Vectors and Matrices: Vectors: definition only, metrics :- Definition and concept, matrix addition, multiplication, scalar multiplication, transpose of a matrix, square matrices, invertible matrices, inverse of a matrix, determinants, basic theorems of determinants, Boolean Matrix.	

TEXT BOOKS:

1.	C Liu, D. Mohapatra : Elements of Discrete Mathematics: A Computer Oriented Approach | 4th Edition 2017
2.	Seymour Lipschutz, Marc Lipson: SCHUM'S OUTlines, Discrete Mathematics, 2nd Edition, TATAMcGRAW-HILL.

REFERENCE BOOKS:

1.	Ralph P Grimaldi, B V Ramana: Discrete and Combinatorial Mathematics, 5th Edition, PEARSON.
2.	Kenneth H Rosen: Discrete mathematics and Its Applications, 5th Edition, TATA McGRAW-HILL.

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Compute the set operations like union, intersection, symmetric difference
CO2.	Derive logical implications and equivalences using laws of logic, describe use quantifiers and prove given statement in different ways
CO3.	Determine the inverse of the function, functional values
CO4.	Differentiate the functions as lattice or not able to prove the theorems on Boolean algebra.
CO5.	Apply basic mathematical operations on Matrices to compute inverse of a matrix

Unix fundamentals & shell programming

Contact Hours/Week	:	(3+2)	Credits	:	04
Total Lecture Hours	:	39	CIE Marks	:	50
Total Tutorial Hours	:	13	SEE Marks	:	50
Course Code	:	1RMCA02			

Course Objectives:

This Course will enable students to:	
1.	Describe the architecture of Unix Operating System (OS). – (Comprehension)
2.	Demonstrate and analyse the Unix commands usage. – (Comprehension, Analysis, Application)
3.	Use Unix utilities to create simple tools for information processing. – (Application)
4.	Illustrate the power of Unix shell by writing shell scripts. – (Application, Synthesis)
5.	Define various Unix standards. – (Knowledge)
6.	Explain and analyse the process concepts in Unix OS. – (Comprehension, Analysis)
7.	Illustrate the power of Report generation using sed and awk(Application and synthesis)

UNIT – I	**06 Hours**
Introduction : The operating system , The UNIX operating system , How it all clicked , The UNIX architecture , Features of UNIX ,Locating commands , Internal and external commands ,command structure , flexibility of command usage , browsing the manual pages on line *General Purpose Utilities* : cal, date, echo, printf, bc, script , passwd , who , uname ,tty , sty The file system: The file, What's in a file name? The parent – child relationship, pwd , the home directory , cd change the current directory , mkdir : making directories , rmdir : removing directories . absolute pathnames , relative pathnames Handling Ordinary Files: cat, cp, rm, mv, more, file, wc, od, cmp, comm., diff.	

UNIT – II	**06 Hours**
Basic file attributes : ls –l , -d option , File permissions , chmod The vi editor : vi basics , Input Mode : Entering and replacing Text, saving text and quitting – The ex mode , Navigation , editing Text , undoing ,repeating the last command , searching for a pattern , substitution. Simple filters: The sample Database, pr, head, tail, cut, paste, sort, uniq, tr commands. More file attributes: File systems and inodes hard links, symbolic links, umask, find.	

UNIT – III	**15 Hours**
The Process: Process basics, ps: Processes status, system processes Mechanism of Process Creation, Internal and External Commands. The SHELL: The shell's interactive cycle, pattern matching- The-wild-cards, Escaping- The backslash (\), Quoting, Redirection - The three standard files, /dev/null and /dev/tty: Two special files, Pipes, Tee: creating a tee, Command Substitution, Shell Variables. Essential Shell Programming : Shell Scripts, read, Command Line Arguments, Exit Status of a Command, The Logical Operators && and ||, if and case conditions, expr, while, for, set and shift , The here document	

UNIT – IV	**08 Hours**
Filters using Regular expressions: grep, Basic Regular Expression, egrep, fgrep. Sed : The stream editor , Line addressing , context addressing , writing selected lines to a File , Text Editing , substitution	

UNIT – V	07 Hours
Awk – An advanced filter: Simple awk Filtering, Splitting a Line in to Fields, printf, the comparison Operators, Number Processing, Variables, The –f option, The BEGIN and END sections, Built-in variables	

TEXT BOOKS:

1.	Sumitabha Das, UNIX Concepts and Applications, Third Edition, Tata McGraw Hill Chapters : 1 , 2.1 ,2.4, 2.5 , 3.1 to 3.5 , 4 , 5 , 6 ,7 , 8, 9, 10.1 to 10.5 , 13.1 , 13.2 , 13.3 , 13.5, 13.7, 14, 15, 16, 21.1 to 21.9

REFERENCE BOOKS:

1.	Kenneth Rosan et al, UNIX: The Complete Reference, Osborne/McGraw Hill, 2000
2.	M G Venkateshmurthy UNIX and Shell Programming, Pearson Education Asia, 2005

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Interpret the basic Unix commands , various components of Unix along with Unix file system
CO2.	Apply suitable commands and basic filters for file processing using vi editor
CO3.	Demonstrate the importance of process, shell and essential shell programming
CO4.	Design solutions for Text processing problems using Regular Expressions and tools like grep and Sed
CO5.	Design suitable AWK scripts to solve computing problem

Object Oriented Programming with C++

Contact Hours/Week	:	(3+2)	Credits	:	04
Total Lecture Hours	:	39	CIE Marks	:	50
Total Tutorial Hours	:	13	SEE Marks	:	50
Course Code	:	1RMCA06			

Course Objectives:

This Course will enable students to:	
1.	To illustrate the power of Modularization and user defined data type
2.	Provide in-depth coverage of object-oriented programming principles and techniques using C++.
3.	Introduce the following topics such as classes, overloading, data abstraction, information hiding, encapsulation, inheritance, polymorphism, templates.
4.	Briefly covers C++ implementation and object-oriented considerations for software design and reuse.

UNIT – I	08 Hours

Principles of object oriented programming: Object oriented programming paradigm, Basic concepts of OOP, Benefits of OO languages. A sample program, Structure of C++ program.

Different data types, Reference Variables, Scope resolution operator,
Functions In C++: Introduction, The main function (), function prototyping, Call by reference, return by reference, Inline function, Default arguments, function overloading.

UNIT – II	08 Hours

Classes And Objects: Difference between Class and Structure, specifying class, Defining member functions, A C++ Program with class, making an outside function Inline, nesting of member functions, private member functions, arrays within a class.
Static members: Static data members and member functions array of objects, objects as function arguments, Friendly functions, returning objects.

UNIT – III	08 Hours

Constructors And Destructors: Introduction, constructors, parameterized constructors, multiple constructors in a class, constructors with default arguments, dynamic initialization of objects, copy constructors, constructing two dimensional arrays, destructors.
Operator Overloading: Defining operator overloading, overloading unary and binary operators, overloading binary operators using friend, rules for overloading operators.

UNIT – IV	07 Hours

Inheritance: Introduction, defining derived classes, single inheritance, making private member inheritable. **Types of Inheritance:** Multilevel, multiple, hierarchical, hybrid inheritance, Virtual base classes. Constructors in derived classes member inheritable, multilevel, multiple, hierarchical, hybrid inheritance, virtual base classes.

UNIT – V	08 Hours

Virtual Functions And Polymorphism: 'this' pointer, virtual functions-calling a virtual function through base class reference. Pure virtual functions-examples, using virtual functions.
Templates: class templates, class templates with multiple parameters, function templates, function templates with multiple parameters, overloading of template functions, member function templates, Non-type template arguments.

TEXT BOOKS:

1.	E Balagurusamy, Object oriented programming with C++, 4th edition, Tata Mc_Graw Hill publishing.

REFERENCE BOOKS:

1.	Herbert Schildt , The Complete Reference C++, , 4th Edition, TMH
2.	Stanley B. Lippman, Josee Lajoie , Barbara E. Moo, C++ Primer, 4th Edition, Addison Wesley publishing.
3.	Sourav Sahay , Object-Oriented Programming with C++, , Oxford University Press.

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Design the programs involving decision structure, loop and functions.
CO2.	Design, implement and debug simple programs by using object-oriented concept
CO3.	Apply the concept of constructors, destructors and operator overloading.
CO4.	Implement the programs by using Inheritance for achieving code reusability.
CO5.	Apply the polymorphism features of OOPs and generic function Techniques to solve the problems.

Professional Communication and Ethics

Contact Hours/Week	:	L+T (3+0)	Credits	:	03
Total Lecture Hours	:	39	CIE Marks	:	50
Total Tutorial Hours	:		SEE Marks	:	50
Course Code	:	1RMCA04			

Course Objectives:

This Course will enable students to:	
1.	Cover topics including: resume and cover letter writing; professional business documents –memo, letter, and email.
2.	Discuss a variety of communication strategies and techniques as they pertain to professional communication.
3.	Demonstrate the most effective oral communication skills that fit their personal and professional communication style
4.	Discuss Professional communication strategies in corporate settings.
5.	Discuss ethics in business word.

UNIT – I	10 Hours
Basics of Technical Communication Introduction, Process of Communication, Language as a Tool, Levels of Communication, The flow of Communication, Communication Networks, Importance of Technical Communication Barriers to Communication: Definition of Noise, Classification of Barriers. Active listening: introduction, types of listening, traits of a good listener, active versus passive listening	

UNIT – II	10 Hours
Strategies in the job search process. * Job search, Building a network of contacts, identifying appropriate jobs, finding your employer, preparing the application documents, constructing the resume. Interviews: Introduction, Objectives, Types of Interviews, Job Interviews Group Communication: Introduction, Group Discussion, Organizational Group discussion, Group discussion as part of selection process	

UNIT – III	07 Hours
Effective Presentation Strategies Introduction, Defining purpose, Analyzing Audience and Locale, Organizing Contents, preparing outline, Visual Aids, Understanding Nuances of Delivery, Kinesics, Proxemics, Paralinguistic, Chronemics.	

UNIT – IV	06 Hours
An Overview of Ethics What are Ethics? Ethics in the Business World, Fostering good business ethics, improving corporate ethics, improving corporate ethics, Ethics in Information Technology (IT)	

UNIT – V	07 Hours
Privacy Privacy Protection and the Law, The right of privacy, Recent history of privacy protection, Key Privacy and Anonymity Issues, Data encryption, Identity theft, Consumer profiling	

TEXT BOOKS:

1.	Meenakshi Raman and Sangeeta Sharma: Technical Communication - Principles and Practices, Oxford University Press, 2011, Chapters: 1, 2, 4, 5, 6, 7
2.	Raymond V Lesikar, Flatley, Rentz, Pande : Business Communication :making connections in a digital world, 11th Edition, Tata McGraw-Hill Edition. Chapter: 9.
3.	George Reynolds: Ethics in Information Technology, Thomson Course Technology, 2nd Edition, 2007 Chapters: 1, 4.

REFERENCE BOOKS:

1.	M.Ashraf Rizivi: Effective Technical Communication, 1st Edition, Tata McGraw Hill, 2005.
2.	Mike W Martin and Ronald Schinzinger: Ethics in Engineering, 3rd Edition, Tata McGraw Hill, 2003.

Self-Learning Component
The SLC is evaluated for a 4 Marks(Assignment component)

- **Group Communication - Conferences, symposia and seminars**
- **Letters -Business letters, , covering letter, Resume writing**
- **Reports- Importance of report, objective of reports, characteristic of a report, categories of report**

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Exhibit communication skills required for a profession
CO2.	Identify the job search strategies
CO3.	Demonstrate the effective presentation strategies
CO4.	Realize ethics in the business word
CO5.	Adapt ethical and privacy practices in day to day life and profession.

Fundamentals of Computer Organization

Contact Hours/Week	:	L +T(3+0)	Credits	:	03
Total Lecture Hours	:	39	CIE Marks	:	50
Total Tutorial Hours	:		SEE Marks	:	50
Course Code	:	1RMCA05			

Course Objectives:

This Course will enable students to:	
1.	Understand common forms of number representation in digital electronic circuits and to understand the logical operation of basic logic gates.
2.	Understand the types of computers and basic structure of the computer systems.
3.	Understand the concept of memory locations, storage, memory operations, various ways of addressing operands, and the implementation of stacks & queues.
4.	Understand the connection of I/O devices to the bus, handling of interrupts, transfer of data and the types of ports
5.	Understand the structure of various types of memory systems along with their operations.
6.	Understand how the basic arithmetic operations can be carried out along with the required circuits.

UNIT – I	**08 Hours**
Logic Gates : The AND Gate, The OR gate, The inverter and Buffer, The NAND gate, the NOR Gate, The exclusive OR gate, The Exclusive NOR Gate, The NAND Gate as a universal Gate, Gates with More than two inputs, Using Inverters to convert gates. **Combining Logic Gates:** Constructing Circuits from Boolean Expressions, Drawing a circuit from a Maxterm Boolean Expression, Truth Tables and Boolean Expressions, Sample Problem, Simplifying Boolean Expressions, Karnaugh Maps, Karnaugh Maps with three variables Karnaugh Maps with four variables, more Karnaugh Maps, using Demorgan's Theorems.	

UNIT – II	**07 Hours**
Arithmetic Circuits: Binary Addition, Half Adders, Full Adders, Three Bit Adders, Binary Subtraction, Parallel Subtractors, IC Adders, Binary Multiplication, Binary Multipliers. **Basic structure of computers :**Computer types, Functional units, Basic operational concepts, Bus structures	

UNIT – III	**08 Hours**
Machine Instructions & Programs Memory locations and addresses(Byte addressability, Big-endian & Little-endian assignments, Word alignment, Accessing numbers, characters & character strings), Memory operations, Instructions and instruction sequencing(Register transfer notation, Assembly language notation, Basic instruction types, Instruction execution & Straight-line sequencing, Branching, Condition codes), Addressing modes(Implementation of variables and constants, Indirection and pointers, Indexing & arrays, relative addressing, Additional modes), Basic Input/Output operations, Stacks and queues	

UNIT – IV	**08 Hours**
Input/Output Organization Accessing I/O devices, Interrupts (Interrupt hardware, Enabling and disabling interrupts, Handling multiple devices, Controlling device requests, Exceptions), Direct memory access (Bus arbitration), Buses (Synchronous bus, Asynchronous bus), Interface circuits (Serial port)	

UNIT – V	08 Hours
The Memory System Some basic Concepts, Semiconductor RAM memories (Internal organization of memory chips, Static memories, Asynchronous DRAMs, Synchronous DRAMs, Memory system considerations, Rambus memory), Read Only Memories (ROM, PROM, EPROM, EEPROM, Flash memory), Speed, size & Cost, Cache memories (Mapping functions)	

TEXT BOOKS:

1.	Tokheim: Digital Electronics Principles and Applications, TATA McGraw Hill, 6th Edition, 2004. (Chapters: 3-1 to 3-10, 4-1 to 4-9, 4-16, 10-1 to 10-9)
2.	Carl Hamacher, Z Varnesic and S Zaky: Computer Organization, 5th Edition, McGraw Hill, 2002. (Chapters: 1.1 to 1.4, 2.2, 2.3, 2.4, 2.5, 2.7, 2.8, 4.1, 4.2(4.2.1 to 4.2.5), 4.4, 4.5(4.5.1,4.5.2), 4.6(4.6.2), 5.1, 5.2(5.2.1 to 5.2.4, 5.2.6, 5.2.7), 5.3, 5.4, 5.5(5.5.1)

REFERENCE BOOKS:

1.	M Morris Mano: Digital Logic and Computer Design, 10th Edition, Pearson, 2008.
2.	Morris Mano and Charles R Kime: Logic and Computer Design Fundamentals, 2nd Edition, Pearson Education, 2001.
3.	Mostafa Abd-El-Barr & Hesham El-Rewini, Fundamentals of Computer Organization and Architecture, (2005) Wiley publishing.

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Design the combinational logic circuits for the Boolean expressions and interpret the De Morgan's laws.
CO2.	Analyze and design arithmetic circuits and interpret the basic structure of computer systems.
CO3.	Distinguish & develop machine instructions and programs using various addressing modes.
CO4.	Interpret the organization of input and output devices.
CO5.	Analyze the working of various memory systems and design the internal organization of a memory chip.

Unix Lab

Contact Hours/Week	:	3	Credits	:	1.5
Total Lecture Hours	:	-	CIE Marks	:	50
Total Tutorial Hours	:	-	SEE Marks	:	50
Course Code	:	1RMCAL1			

Course Objectives:

This Course will enable students to:	
1.	Understand the basic UNIX commands and its environment
2.	Obtain complete knowledge on Shell programming
3.	Understand the concepts of awk and sed

Sl.no	Programs
1.	Following exercises may require referring to man pages of commands for syntax and options and their usage I. Display current date in mm-dd-yy and dd-mm-yy formats. II. Use ls command to list only directories along with other attributes including inode number III. Write commands for the following queries: list of user names who have currently logged in list of logins for a given user name IV. Perform the following operations. 1. Create myproj directory in the login directory (home directory). 2. Create src/include directory within myproj using a single Command. V. List the directory entries (in long format) as per the following criteria in ascending order of file size
2.	a)Write a shell script that accepts two file names as arguments, checks if the permissions for these files are identical and if the permissions are identical, output common permissions and otherwise output each file name followed by its permissions b)Write a non-recursive shell script which accepts any number of arguments and prints them in the reverse order (For example, if the script is named rags, then executing rags A B C should produce C B A on the standard output).
3	a) Write a shell script that accepts a path name and creates all the components in that path name as directories. For example, if the script is named mpc, then the command mpc a/b/c/d should create directories a, a/b, a/b/c, a/b/c/d. b) Write a shell script that takes a valid directory name as an argument and recursively descend all the subdirectories, finds the maximum length of any file in that hierarchy and writes this maximum value to the standard output.
4	a)Write a shell script which accepts valid log- in names as arguments and prints their corresponding home directories, if no arguments are specified, print a suitable error message b) Write a shell script that accept one or more filenames as argument and convert all of them to uppercase, provided they exist in current directory.
5.	a) Write a shell script that displays all the links to a file specified as the first argument to the script. The second argument, which is optional, can be used to specify in which the search is to begin. If this second argument is not present, the search is to begin in current working directory. In either case, the starting directory as well as all its subdirectories at all levels must be searched. The script need not include any error checking. b) Write a shell script that accepts as filename as argument and display its creation time if file

	exist and if it does not send output error message.
6.	a) Write a shell script to display the calendar for current month with current date replaced by * or ** depending on whether the date has one digit or two digits. b) Write a shell script to find smallest of three numbers that are read from keyboard
7.	a) Write a shell script that compute gross salary of an employee, accordingly to rule given below : If basic salary is < Rs15000 then HRA=10% of basic & DA=90% of basic. If basic salary is >=Rs15000 then HRA=Rs500 & DA=98% of basic b) Write a shell script that delete all lines containing a specific word in one or more file supplied as argument to it.
8.	a) Write a shell script that accept a list of filenames as its argument, counts and report occurrence of each word that is present in the first argument file on other argument files. b) Write a shell script that accept the file name, starting and ending line number as an argument and display all the lines between the given line number
9.	a) Write a shell script to find the factorial of a given number b) Write a shell script that determine the period for which a specified user is working on system
10.	a) Create a program called words that will continue to prompt the user to input a single word until the user enters quit. Save each word that is entered. After the user types quit echo back all of the words that have been entered. (Optional: display all of the words entered in alphabetical order.) b)Create an awk program which prints the alternate lines like 2^{nd}, 4^{th}, 6^{th} etc…from the file /etc/group.

Course Outcomes:

After the completion of this course, students will be able to:	
CO1	Design command structure and execute Basic commands
CO2	Demonstrate shell script using simple filters and grep
CO3	Develop and execute shell scripts using advance filters sed and report report generation language - AWK

OOPS With C++ Lab

Contact Hours/Week	:	3	Credits	:	1.5
Total Lecture Hours	:	-	CIE Marks	:	50
Total Tutorial Hours	:	-	SEE Marks	:	50
Course Code	:	1RMCAL3			

Course Objectives:

This Course will enable students to:	
1.	Understand object oriented programming concepts.
2.	Understand OOP concepts such as polymorphism, inheritance
3.	Obtain knowledge on standard C++ I/O classes and exception handling

Part A

Sl.no	Programs
1.	Write a C++ program to swap the values of two variables and demonstrates a function using call by value.
2.	Write a C++ program to swap the values of two variables and demonstrates a function using Call by reference using reference type (&).
3	Write a C++ program to swap the values of two variables and demonstrates a function using Call by reference using pointer (*).
4	Write a program to find the largest and smallest of two numbers. (Use inline function MAX MIN to find largest & smallest of 2 numbers)
5.	Write a C++ program to find the area of rectangle using function calls as rect(), rect(10.0), rect(10.0,12.0) to demonstrate the default arguments passing mechanism.
6.	Write a program to calculate the volume of different geometric shapes like cube, cylinder and sphere and hence implement the concept of Function Overloading.
7.	Write a C++ program to create a template function for swapping of two numbers and demonstrate this for swapping two integers, doubles and char variables.

Part B

Sl.no	Programs
1.	Given that an EMPLOYEE class contains following members: Data members: Emp_number, Emp_name, Basic, DA, IT, Net_sal and Member functions: to read the data, to calculate the data, and to print the data members. Write a C++ program to read the data of N employees and compute Net Sal of each employee (DA = 52% of basic and income tax (IT) = 30% of the gross salary).
2.	Write the following C++ programs to create a class called COMPLEX and implement the following overloading functions ADD that return a complex number: (i) Design a program to handle following friend functions ADD (a, s2) – where 'a' is an integer to be added to real part only and s2 is a complex number ADD (s1, s2) – where s1 and s2 are complex numbers. (ii) Design a program to handle following member functions s2.ADD (a) – where 'a' is an integer to be added to real part only and s2 is a complex number s1.ADD (s2) – where s1 and s2 are complex numbers.
3	Friend functions and friend classes: (i) Write a program to define class name HUSBAND and WIFE that holds the income respectively. Calculate and display the total income of a family using Friend function. (ii) Write a program to accept the student detail such as name and 3 different marks by

	get_data() method and display the name and average of marks using display() method. Define a friend class for calculating the average of marks using the method mark_avg().
4	Create a class called MATRIX using two-dimensional array of integers. Implement the following operations by overloading the operator == which checks the compatibility of two matrices to be added and subtracted. Perform the addition and subtraction by overloading the + and – operators respectively. If (m1== m2) then perform m3= m1+m2 and m4 = m1-m2 else display error.
5.	Write a program to create an HUMAN class with features as number of Head, Legs, Hands.(NOTE: Number of Head, Legs and Hands are of integer types) (i) Create an object HUMAN1 using default constructor. (Default features to have 1 Head, 2 Legs and 2 Hands) (ii) Create an object HUMAN2 with customized inputs using Parameterized Constructor
6.	Create an object HUMAN3 using existing object HUMAN1 (Copy Constructor). All Humans die after their lifetime. (Destructor) Demonstrate Simple Inheritance concept by creating a base class FATHER with data members SurName and BankBalance and create a derived class SON, which inherits SurName and BankBalance feature from base class but provides its own feature FirstName and DOB. Create and initialize F1 and S1 objects with appropriate constructors and display the Father & Son details. (Hint : While creating S1 object, call Father base class parameterized constructor through derived class by sending values)

Note 1: In the practical Examination each student has to pick one question from PART-A and PART-B each.
Note 2: Change of program is not permitted in the Practical Examination.

Course Outcomes:

After the completion of this course, students will be able to:	
CO1	Recognize, examine and resolve the real world problems with object oriented paradigm.
CO2	Apply all the Oops concepts
CO3	Identifying best solution for a problem-domain

Business English Lab

Contact Hours/Week	:	2	Credits	:	01
Total Lecture Hours	:	26	CIE Marks	:	50
Total Tutorial Hours	:		SEE Marks	:	50
Course Code	:	1RMD1			

Course Objectives:

This Course will enable students to:	
1.	Acquire practical and oral communication skills,
2.	Use written Communication skillfully in different business situations
3.	Practice principles of effective business writing and document design in all written documents.
4.	Design and deliver a persuasive presentation that convinces the audience of the topic's relevance and overcomes resistance, using appropriate visual support and adhering to a specified time limit.

UNIT – I	12 Hours

Introduction:

Diagnostic test to evaluate, attitude, goals, level of exposure and skills in English Language-set Goals to achieve time bound results in making improvement and develop skillls in English Language

Develop an attitude to learn to empower oneself and learning how to learn. An aptitude for listening and reproducing.

Understanding the nuances of English Language-capturing the English sound with ears and expressing thoughts to create meanings by employing articulation techniques- Basic English Vocabulary-Comprehensive vocabulary lesson, Phrases and Sentences to cover topics across personal, social and business scenarios

UNIT – II	12 Hours

Exposure to the grammatical framework

Framing sentences- what they are and how they made- simple, compound and complex sentences-learning the basics of sentence structures-subject, predicate and object- usages of verbs, prepositions, adjectives, adverbs, articles, conjunctions and nouns.

Punctuations- how to use them effectively to generate, proper meanings in sentences and paragraphs

Tenses - Tenses transformation - W-H Questions – Question Tags – Active and Passive Voice – Degrees of Comparison.

Oral Communication:

Generating ideas- learning how to think- thinking in English- setting the points in coherent way and capturing them in sentences and paragraphs to express facts, opinions and thoughts in logical progression

What is oral communication – Principles of successful oral communication – Barriers to communication – What is conversation control – Reflection and empathy: two sides of effective oral communication – Effective listening – Nonverbal communication.

UNIT – III	12 Hours
Individual and group presentation- presenting facts, analysis, commentaries, news on key and interesting business and social situations-noting points, preparing drafts and delivering before audiences, using the standard tools and techniques Gaining skills in written English- Purpose of writing- structuring and designing the presentation- composing drafts and documents- clarity in writing- principles of effective writing-approaching the writing process systematically: The 3X3 writing process for business communication: Pre writing – Writing – Revising – Specific writing features – Coherence – Electronic writing process.	

UNIT – IV	12 Hours
Case Method of Learning: Understanding the case method of learning – Different types of cases – Overcoming the difficulties of the case method – Reading a case properly (previewing, skimming, reading, scanning) – Case analysis approaches (Systems, Behavioral, Decision, Strategy) – Analyzing the case – Dos and don'ts for case preparation.	

UNIT – V	12 Hours
Employment communication: Introduction – Writing CVs – Group discussions – Interview skills – Business Etiquettes; Group Communication: Meetings – Planning meetings – Objectives – Participants – Timing – Venue of meetings – Leading meetings	

TEXT BOOKS:

1.	Chaturvedi P. D, Mukesh Chaturvedi, Business Communication: Concepts, Cases and Applications, 3rd Edition, Pearson Education, 2013.
2.	Mary Ellen Guffey, Business Communication: Process and Product, 8th Edition, Cengage Learning, 2015.

REFERENCE BOOKS:

1.	Lesikar, Flatley, Rentz & Pande, Basic Business Communication, 13th Edition, Mcgraw Hill Education, 2015
2.	Penrose, Rasberry, Myers, Advanced Business Communication, 5th Edition, Cengage Learning, 2007.

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Develop a learning habit and motivation
CO2.	Conceptualize the patterns and usage of English Language
CO3.	Learn and use vocabulary, phrases to comfortably describe features, express thoughts and opinions, assimilate facts from varied materials and reproduce them by using standard forms of expressions in written and oral forms
CO4.	Acquire skills to think creatively, organize ideas and reproduce them in English Language across wide range of concepts.
CO5.	Learn to make sentences, frame paragraphs and create standard documents.
CO6.	Gain insight and experience in delivery skills- word choice, tone, and format.
CO7.	Imbibe acceptable presentation skills (e.g., demonstrates confidence and poise through body language, eye contact, vocal tone).
CO8.	Learn Group Discussion, Interview Skills and develop ability to build CV/Resume

II Semester

Web Technologies

Contact Hours/Week	:	L +T(3+2)	Credits	:	04
Total Lecture Hours	:	39	CIE Marks	:	50
Total Tutorial Hours	:	26	SEE Marks	:	50
Course Code	:	2RMCA01			

Course Objectives:

This Course will enable students to:	
1.	Creating the small web page using html5
2.	Different elements of html4 and html5
3.	Use different tags of html to create web page
4.	Use of CSS and Javascript
5.	Use of JQuery

UNIT – I	06 Hours
HTML & HTML 5: Getting Started, The browser wars, Feature Detection, Browser Detection, HTML Basics, Links, Images, Tables, Lists, Blocks, IFrames, Forms, Form Elements, Input Types, Input Attributes, The HTML5 New Elements – Form Input Types, Form Elements, Form Attributes, Programming Examples	

UNIT – II	09 Hours
HTML5 Semantics, HTML5 Graphics – Canvas, SVG, HTML5 Media – Video, Audio, HTML5 Web Storage, HTML5 Media – Video, Audio, HTML5 Web Storage, Programming Examples CSS CSS Basic – Introduction, Syntax, Id & Class, CSS Types, CSS Styling – Styling Backgrounds, Styling Text, Styling Fonts, Styling Links, Styling Lists, Styling Tables. CSS Box Model – Border, Margin, Padding, Outline	

UNIT – III	08 Hours
CSS3: Introduction, Borders, Backgrounds, Text Effects, 2D Transforms, 3D Transforms, Transitions, Programming Examples **JavaScript:** JS Introduction, JS Statements, JS Comments, JS Variables, JS Operators, JS Data Types, JS Functions, JS Objects, JS Comparisons, JS Conditions, JS Switch, Programming Examples	

UNIT – IV	08 Hours
JS Looping Statements – For, While, JS Breaks, JS DOM – DOM Introduction, DOM Elements, DOM HTML, JS Objects – Object, Number, String, Date, Array, Boolean, Math Object, Programming Examples	

UNIT – V	08 Hours
Dynamic Document with JavaScript: Positioning Elements, Moving Elements, Element Visibility, Changing colors and fonts, Stacking Elements, Pattern Matching using Regular Expressions, Positioning JS Events & Event Handling, Click Event, Focus & Blur Events, Load and Unload Events, Mouse Events, Keyboard Events, Programming Examples	

TEXT BOOKS:

1.	HTML 5 Black Book (Covers CSS3, JavaScript, XML, XHTML, AJAX, PHP, jQuery) 2Ed.
2.	Programming the World Wide Web (4th Edition) 4th Edition by Robert W. Sebesta

REFERENCE BOOKS:

1.	https://www.w3schools.com/html/default.asp
2.	https://www.w3schools.com/css/default.asp
3.	https://www.w3schools.com/js/default.asp

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Develop a static web page using html and html5.
CO2.	Apply the basic styles to design the webpage.
CO3.	Apply the style and validation for the elements of the webpage using CSS3 and JavaScript.
CO4.	Develop JavaScript using built–in objects like String, Array, Date etc.
CO5.	Apply different events to validate the inputs for development of web application

Self-Learning Components

The SLC is evaluated for a CIE Marks of 7(Assignment and quiz)

i. For the self-study component, students should submit a report for having developed a static web site for the following (not a limitation):
   - Commercial site.
   - Educational Institution.
   - Business House.
   - Software Company

Or

Students have to give a seminar on a chosen topic pertaining to web technologies.

Data Structures using C++

Contact Hours/Week	:	(3+2)	Credits	:	04
Total Lecture Hours	:	39	CIE Marks	:	50
Total Tutorial Hours	:	26	SEE Marks	:	50
Course Code	:	2RMCA02			

Course Objectives:

This Course will enable students to:	
1.	Understand how different data structures store data on physical memory.
2.	Understand the basic operations of linear data structures.
3.	Understand the basic operations of non-linear data structures.
4.	Realize the working of various sorting and searching techniques.
5.	Compare different implementations of the same data structure

UNIT – I	**08 Hours**
Hours Classification of Data Structures: Primitive and Non- Primitive, Linear and Nonlinear; Data structure Operations Stack: Definition, Operations (push, pop and peep) and Applications: Infix to postfix conversion, evaluation of postfix expression Recursion - Recursive definition and processes, Properties of recursive definition or Algorithm, Recursive algorithms: Factorial, GCD, Fibonacci sequence, sum of individual digits of given number, product of two numbers, find largest element in array	

UNIT – II	**08 Hours**
Queue: Simple queues, Operations on simple queue(insertion and deletion). Queue Variants: Circular Queue, Priority Queues, Double Ended Queues. Implementation of simple queues and circular queues using arrays. Linked list: Limitations of array implementation, Memory Management: Static (Stack) and Dynamic (Heap) Memory Allocation. Basic operation on singly linked list: insertion and deletion at first position, insertion and deletion at last position, insertion and deletion after a given key, searching a key. Applications of Singly linked list: Implementation of stacks and queues using singly linked list.	

UNIT – III	**08 Hours**
Implementation of Double ended queues and Priority queues using singly linked list. Doubly linked list: Basic operations on doubly linked List: Insertion and deletion at first position, insertion and deletion at last position, insertion and deletion after a given key, searching a key. Applications of doubly linked list: Implementations of stacks and queues using doubly linked list. Circular Singly Linked List: create CSLL, delete first node in CSLL, search key in CSLL. Circular doubly linked list introduction.	

UNIT – IV	**08 Hours**
Trees: Definitions, Terminologies, Array and linked Representation of Binary Trees using arrays and linked list, Types- Complete/full, Almost Complete, Strictly, Skewed, Binary search tree, binary expression tree, threaded binary tree, Heap trees. Traversal methods - Inorder, postorder, and preorder. Implementation of Binary Search Trees - Creation, Tree Traversal, Searching, finding minimum element, maximum element in BST. Graphs: Memory representation of graphs, Graph traversal methods: DFS and BFS (no implementations of graph and graph traversals.)	

UNIT – V	07 Hours
Sorting & Searching: Bubble sort, Selection sort, Radix sort, Shell sort. Implementation of Bubble sort, Insertion Sort, Selection sort using templates Implementation of Linear Search and Binary Search using templates. Hashing: The Hash Table organizations, Hashing Functions, Collision-Resolution Techniques. Problems on hashing(linear probing, chaining).	

TEXT BOOKS:

1.	Data Structures Using C and C++ by YedidyahLangsam and Moshe J. Augenstein and Aaron M Tenanbanum, 2nd Edition, Pearson Education Asia, 2002.

REFERENCE BOOKS:

1.	Introduction to Data Structure and Algorithms with C++ by Glenn W. Rowe
2.	Data Structures and Algorithms in C++, Michael T. Goodrich, Roberto Tamassia, David M. Mount, ISBN 978-0-470-38327-8, February 2011. Paperback
3.	Data Structures Using C++ by Malik, D. S. [Cengage,2009] (Paperback) 2nd Edition

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Analyse the working of Stacks and Queues, its applications and implementation using static memory allocation.
CO2.	Distinguish types of linked list; its applications and implementation using dynamic memory allocation
CO3.	Distinguish types of nonlinear data structures: Trees and Graphs, its applications and its implementation.
CO4.	Analyse the working of various sorting and searching techniques and its implementation.

Operations Research

Contact Hours/Week	:	L +T(3+2)	Credits	:	04
Total Lecture Hours	:	39	CIE Marks	:	50
Total Tutorial Hours	:	26	SEE Marks	:	50
Course Code	:	2RMCA03			

Course Objectives:

This Course will enable students to:	
1.	Understand the basic Operation Research concepts and techniques for solving particular problem and identification of appropriate solution
2.	Identifying decision making techniques, both qualitative and in particular quantitative.
3.	Understand analytical methods to solve real world decision making problems
4.	Build more productive systems based on latest decision tools and techniques.

UNIT – I	**08 Hours**
Introduction and Overview of the Operation Research: What is OR? - The origin of OR - the nature of OR – Different phases of OR solutions - Gathering of data and formulation of mathematical model - Validation and implementation **Introduction to Linear Programming:** Two variable LP model - Construction of the LP Model. Graphical LP solution – OR – models of some practical situations.	

UNIT – II	**08 Hours**
The Simplex Method: Standard LP form and its basic solutions - Computational procedure of the simplex method - Artificial variable method, Big-M method - Special cases in the simplex method	

UNIT – III	**08 Hours**
Transportation Model: Definition of the Transportation Model - Different methods of finding the starting solution - Iterative computation of the solution – Traveling Salesman Problem. **Assignment Model:** The Assignment Model - Mathematical formulation - Solution by Hungarian Method - Transshipment model	

UNIT – IV	**08 Hours**
Introduction to Game Theory: Formulation of game, pure and mixed strategies – Dominance principle, modified Dominance principle - Two persons Zero sum games – Saddle point -, graphical solution of 2 X n and m X 2 games	

UNIT – V	**07 Hours**
Historical development of PERT/CPM Techniques – Network construction – Fulkerson's rule – Determination of Critical path and duration – Slacks/Floats – Estimation of project duration – Crashing – Least cost project scheduling	

TEXT BOOKS:

1.	Operations Research: An Introduction, Hamdy A. Taha, 8th Edition, Prentice Hall India, 2009. Chapters: 1, 2(2.1, 2.2, 2.3), 3, 4, 5 and 13(13.4).

REFERENCE BOOKS:

1.	S. D. Sharma, Operations Research, 15th Edition – Kedarnath – Ramnath, Meerut – Delhi (2008).
2.	Introduction to Operations Research, Frederick S. Hillier and Gerald J. Lieberman, 8th Edition, Tata McGraw Hill, 2007.

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Apply different OR models used in decision making, formulate Linear programming problem and solve using graphical method.
CO2.	Identify, formulate and solve linear programming problems using Simplex method and Big M method.
CO3.	Formulate and solve given management problems like transportation problem and assignment problem.
CO4.	Apply game theory concepts for decision making.
CO5.	Schedule events and activities of a project using CPM & PERT

Software Engineering

Contact Hours/Week	:	L +T (3+0)	Credits	:	03
Total Lecture Hours	:	39	CIE Marks	:	50
Total Tutorial Hours	:		SEE Marks	:	50
Course Code	:	2RMCA04			

Course Objectives:

This Course will enable students to:	
1.	Define software engineering and explain its importance
2.	Discuss the concepts of software products and software processes
3.	Explain the importance of process visibility
4.	Introduce the notion of professional responsibility

UNIT – I	10 Hours
Introduction: Professional software development; Software engineering ethics. **Software Processes**: Software process models; Process activities; Coping with change; The Rational Unified Process.	

UNIT – II	10 Hours
Requirements Software Requirements: Functional and Non-functional requirements; User requirements; System requirements; Interface specification; the software requirements document. **Requirements Engineering Processes**: Feasibility studies; Requirements elicitation and analysis; Requirements validation; Requirements management.	

UNIT – III	07 Hours
System modeling: Context models; Interaction models; Structural models; Behavioral models; Model driven engineering. **Agile Software Development**: Agile methods; Plan- driven and agile development; Extreme programming; Agile project management; Scaling agile methods	

UNIT – IV	06 Hours
Software testing: Development testing; Test driven development; Release testing; User testing. **Project Management**: Risk management; Managing people; Teamwork; Project scheduling;	

UNIT – V	06 Hours
Project planning: Software pricing; Plan driven development; Project scheduling; Agile planning**;** Estimation techniques; Algorithmic cost modeling; Cocomo II model; Project duration and staffing.	

TEXT BOOKS:

1.	Ian Sommerville: Software Engineering, 9th Edition, Person Education Ltd., 2017.(Chapters-: 1, 2, 3, 4, 5, 8, 20, 21)

REFERENCE BOOKS:

1.	Guide to the Software Engineering Body Of Knowledge (**SWEBOK Version 3.**) 2004Library of Congress Online Catalog

2.	Roger.S.Pressman: Software Engineering-A Practitioners approach, 7th Edition,McGraw-Hill, 2010.
3.	Shari Lawrence Pfleeger, Joanne M. Atlee : Software Engineering Theory and Practice, 4rd Edition, Pearson Education, 2009.
4.	Waman S Jawadekar: Software Engineering Principles and Practice, Tata McGraw Hill, 2004.

Assignments:

The students are asked to come up with the Feasibility report and the IEEE-830 format of software requirement specification for the project what they a have completed.

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Practice fundamental knowledge in software engineering
CO2.	Analyze the functional and non-functional requirements.
CO3.	Identify the design and methods for application development
CO4.	Identify the project management and testing activities
CO5.	Manage and motivate the people

Computer Oriented Numerical and Statistical Methods

Contact Hours/Week	:	L+T(3+0)	Credits	:	03
Total Lecture Hours	:	39	CIE Marks	:	50
Total Tutorial Hours	:		SEE Marks	:	50
Course Code	:	2RMCA05			

Course Objectives:

This Course will enable students to:	
1.	Understand the representation of floating point numbers in normalized form, to solve nonlinear equations by using various iterative methods and to write algorithms for the same.
2.	Understand the application of direct and iterative methods for simultaneous algebraic equations along with algorithms.
3.	Understand the type of differentiation problem with the corresponding formula and to write the algorithms for them.
4.	Solve numerical integration problems with algorithms and the foundations of Statistics.
5.	Understand Measures of Central Tendency and Dispersion with problems and algorithms.

UNIT – I	**09 Hours**
Iterative Methods Introduction, Floating point representation of numbers, introduction, beginning an iterative method, the method of successive bisection, the method of false position, Newton-Raphson iterative method. The Secant method, the method of successive approximations, Comparison of iterative methods.	

UNIT – II	**07 Hours**
Solution of Simultaneous Algebraic Equations Introduction, The Gauss elimination method, pivoting, Ill conditioned equations, Refinement of the solution obtained by Gaussian elimination, The Gauss-Seidel iterative method, An algorithm to implement the Gauss-Seidel method, Comparison of direct and iterative methods.	

UNIT – III	**08 Hours**
Numerical Differentiation Introduction, Formulae for numerical differentiation- First derivative using forward difference formula, backward difference formula, Lagrange's interpolation formula, Newton's divided difference formula	

UNIT – IV	**08 Hours**
Numerical Integration Numerical integration, Trapezoidal rule, Simpson's $1/3^{rd}$ rule, Algorithms for integration of tabulated function, Gaussian quadrature formulae, Comparison of integration formulae. **Introduction to Statistics** Introduction, Meaning & Scope of Statistics, Data, Types of Data and Chart, Frequency, Class Boundaries, Frequency Chart.	

UNIT – V	**08 Hours**
Measures of Central Tendency and Dispersion Introduction, Mean for Frequency Less Data, Arithmetic Mean for Discrete & Continuous Frequency Distributions, Weighted Mean, Median for Raw Data, Mode for Raw Data, Dispersion, Skewness & Kurtosis.	

TEXT BOOKS:

1.	V Rajaraman, Computer Oriented Numerical Methods, Prentice Hall of India, 3rd Edition. (Chapters: 2.1, 2.2, 3.1 to 3.8, 4.1 to 4.8, 8.1, 8.2, 8.3, 8.4, 8.6, 8.8, 8.9, 9.1, 9.2, 9.4, 9.5, 9.6, 9.7, 9.8)
2.	Sant Sharan Mishra, Computer Oriented Numerical and Statistical Methods, PHI, 2013 (Chapters: 9.1 to 9.7, 10.1 to 10.9)

REFERENCE BOOKS:

1.	Steven C Chapra, Raymond P Canale, Numerical Methods for Engineers, Tata McGraw-Hill, Fourth Edition.
2.	M K Jain, S R K Iyengar, R K Jain, Numerical Methods for scientific & Engineering computation, NEW AGE, Sixth edition.
3.	Dr. B S Grewal, Numerical Methods in Engineering & Science, Khanna Publishers, 9th edition 2013.
4.	Richard A Johnson, Irwin Miller, John Freund, Probability and Statistics for Engineers, Pearson Education India; 8 edition (2015)
5.	George Argyrous, Statistics for Research: With a Guide to SPSS, SAGE South Asia, Third edition, 2012.

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Represent numbers in normalized floating-point format and apply a suitable iterative method to transcendental equations.
CO2.	Solve problems on simultaneous algebraic equations using the direct and iterative methods.
CO3.	Solve problems on numerical differentiation of data and functions.
CO4.	Solve problems on numerical integration of data, functions and the fundamentals of statistics.
CO5.	Represent the data and calculate the measures of central tendency and dispersion.

CONM & OR lab

Contact Hours/Week	:	2	Credits	:	1.0
Total Lecture Hours	:	-	CIE Marks	:	50
Total Tutorial Hours	:	-	SEE Marks	:	50
Course Code	:	2RMCAL1			

Course Objectives:

This Course will enable students to:	
1.	Understand the representation of floating point numbers in normalized form, to solve nonlinear equations by using various iterative methods and to write algorithms for the same.
2.	Understand the application of direct and iterative methods for simultaneous algebraic equations along with algorithms.
3.	Build more productive systems based on latest decision tools and techniques

Sl.no	**Part A Programs**
Exercises on Computer Oriented Numerical Methods:	
1.	Programs on iterative methods.
2	Programs on Simultaneous Algebraic Equations.
3	Programs on numerical differentiation.
4.	Programs on numerical integration.
5.	Programs on Measures of central tendency and dispersion

Sl.no	**Part B Programs**
Exercises on Operations Research:	
1.	Programs on graphical methods.
2	Programs on Simplex methods.
3	Programs on Big-M methods.
4.	Programs on travelling sales problem (TSP).
5.	Programs on Game theory.

Note: Students are required to execute one question from Part A and one from Part B.

Course Outcomes:

After the completion of this course, students will be able to:	
CO1	Develop the computer programs with C-Programming Language to solve the problems of Numerical methods.
CO2	Develop the Operations Research models by using TORA software tool.

Data Structures with C++ Lab

Contact Hours/Week	:	3	Credits	:	1.5
Total Lecture Hours	:	-	CIE Marks	:	50
Total Tutorial Hours	:	-	SEE Marks	:	50
Course Code	:	2RMCAL2			

Course Objectives:

This Course will enable students to:	
1.	Understand linear and non linear data structure.
2.	Understand applications of stacks, queues, linked list and trees
3.	Understand various sorting and techniques.

Sl.no	Programs
1.	Programs on Stack and stack applications (recursion, conversion of infix to postfix and evaluation of postfix expression using arrays)
2.	Program on Queue and circular queue implementation using arrays
3	Programs on basic operations of singly linked list (create, delete first, delete last, search), doubly linked list (create, delete first, delete last, search) and circular singly linked list (create, delete first, search)
4	Programs on applications of linked list: implement stack, queue, priority queue, double ended queue using singly linked list
5.	Programs for basic operation on binary search tree (create, three tree traversals, finding min and max element) using doubly linked list.
6.	Programs on searching(linear and binary search) and sorting (bubble and selection sort) using templates

Course Outcomes:

After the completion of this course, students will be able to:	
CO1	Design and develop Stacks and queues data structures, its applications and using static memory allocation.
CO2	Design and develop Linear data structures like stack, queues and Linked list and non-linear data structures: Trees using dynamic memory allocation.
CO3	Implementation of various sorting and searching techniques.

Web Technologies Lab

Contact Hours/Week	:	3	Credits	:	1.5
Total Lecture Hours	:	-	CIE Marks	:	50
Total Tutorial Hours	:	-	SEE Marks	:	50
Course Code	:	2RMCAL3			

Course Objectives:

This Course will enable students to:	
1.	Analyze a web page and identify its elements and attributes.
2.	Create web pages using XHTML and Cascading Style Sheets.
3.	Apply the style for the web page using CSS3.

Sl.no	Programs
1.	Programs on HTML5 elements
2.	Programs on HTML5 Web Storage
3	Programs on CSS
4	Programs on CSS3
5.	Programs on JavaScript Statement Identifiers
6.	Programs on JavaScript Objects
7.	Programs on JavaScript Events

Course Outcomes:

After the completion of this course, students will be able to:	
CO1	Analyse a web page and identify its elements and attributes.
CO2	Create web pages using XHTML and Cascading Style Sheets.
CO3	Build dynamic web pages using JavaScript (Client side programming).

III Semester

Analysis and Design of Algorithms

Contact Hours/Week	:	(3+2)	Credits	:	04
Total Lecture Hours	:	39	CIE Marks	:	50
Total Tutorial Hours	:	13	SEE Marks	:	50
Course Code	:	3RMCA01			

Course Objectives:

This Course will enable students to:	
1.	Study paradigms and approaches used to analyze and design algorithms and to appreciate the impact of algorithm design in practice.
2.	Understand how the worst-case time complexity of an algorithm is defined, how asymptotic notation is used to provide a rough classification of algorithms.
3.	Assess how a number of algorithms for fundamental problems in computer science and engineering work and compare with one another.
4.	Understand how there are still some problems for which it is unknown whether there exist efficient algorithms, and how to design efficient algorithms.
5.	To create analytical skills, to enable the students to design algorithms for various real-time problems.
6.	To provide basic idea of machine learning algorithms

UNIT – I	**10 Hours**
Notion of Algorithm, Fundamentals of Algorithmic Problem Solving, Important Problem Types, Basics of data Structures. Analysis Framework, Asymptotic Notations and Basic efficiency classes. Mathematical analysis of Recursive and Non-recursive algorithms, Examples. Algorithm visualization. Brute force techniques: Selection Sort and Bubble Sort, Sequential Search and String Matching, Exhaustive Search.	

UNIT – II	**08 Hours**
Divide-and-Conquer: Divide and conquer recurrence relation, Master's theorem, Merge sort, Quicksort, Multiplication of large integers. Decrease-and-Conquer: Insertion Sort, Depth First and Breadth First Search, Topological sorting.	

UNIT – III	**08 Hours**
Transform-and-Conquer: Presorting, Generating Balanced Search Trees. Space and Time Tradeoffs: Sorting by Counting, Input Enhancement in String Matching, Hashing techniques	

UNIT – IV	**08 Hours**
Dynamic Programming: Computing a binomial coefficient, Warshall's and Floyd's Algorithms Greedy Technique: Prim's Algorithm, Kruskal's Algorithm, Dijkstra's Algorithm	

UNIT – V	**07 Hours**
Algorithmic power P, NP and NP complete problems. Backtracking: N-queen's problem, Introduction to machine learning algorithms, supervised and unsupervised learning,	

TEXT BOOKS:

1.	Anany Levitin: Introduction to the Design and Analysis of Algorithms, 3rd Edition, Pearson Education
2.	**Ethem Alphydin**, Introduction to Machine Learning, III edition, MIT Press.

REFERENCE BOOKS:

1.	Horowitz E., Sahani S., Rajasekharan S.: Fundamentals of Computer Algorithms, 2nd Edition, Universities Press, 2007.
2.	Coremen T.H., Leiserson C.E., and Rivest R.L.: Introduction to Algorithms, 2nd Edition, PHI, 2001.

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Apply an appropriate method for analyzing algorithm complexity, including asymptotic notation, and determine worst case complexity.
CO2.	Apply brute force, divide and conquer and decrease and conquer strategies to design the algorithms for solving sorting and graph problems.
CO3.	Apply transform and conquer and space trade off strategies to design algorithms for solving problems related to trees, string matching and hashing.
CO4.	Problems of practical interest.
CO5.	Identify the limitation of the algorithm in solving complex problems and able to demonstrate the knowledge of machine learning algorithms.

RDBMS

Contact Hours/Week	:	(3+2)	Credits	:	04
Total Lecture Hours	:	39	CIE Marks	:	50
Total Tutorial Hours	:	13	SEE Marks	:	50
Course Code	:	3RMCA02			

Course Objectives:

This Course will enable students to:	
1.	Learn the fundamentals of DBMS with an emphasis on how to organize, maintain and retrieve data from the database.
2.	Learn the basic principles of database design in the form of Entity – Relationship diagrams to handle the needs of any user oriented projects
3.	Learn the programming fundamentals involved in using Structured Query Language and formation its programming lines in the form of queries.
4.	Gain the knowledge on how to design the database by using Normalization process and its normal forms.
5.	Know the different aspects of transactions with the given database.

UNIT – I	**08 Hours**
Introduction, An example, Characteristics of Database approach, Actors on the scene, Workers behind the scene, Advantages of using DBMS approach, A brief history of database applications, when not to use a DBMS. Data models, schemas and instances, Three-schema architecture and data independence, Database languages and interfaces, The database system environment, Centralized and client-server architectures, Classification of Database Management systems.	

UNIT – II	**08 Hours**
Data Modeling using the Entity-Relationship (ER) Model Using High-Level Conceptual Data Models for Database Design, An Example Database Application, Entity Types, Entity Sets, Attributes and Keys, Relationship types, Relationship Sets, Roles and Structural Constraints, Weak Entity Types, Refining the ER Design, ER Diagrams, Naming Conventions and Design Issues, Relationship types of degree higher than two. Relational Model and Relational Algebra Relational Model Concepts, Relational Model Constraints and Relational Database Schemas, Update Operations, Transactions and dealing with constraint violations, Unary Relational Operations: SELECT and PROJECT, Relational Algebra Operations from Set Theory, Binary Relational Operations, JOIN and DIVISION, Additional Relational Operations, Examples of Queries in Relational Algebra, Relational Database Design Using ER- to-Relational Mapping.	

UNIT – III	**10 Hours**
SQL SQL : Data Definition and Data Types, Specifying constraints in SQL, Schema change statements in SQL, Basic queries in SQL, More complex SQL Queries, Insert, Delete and Update statements in SQL, Specifying general constraints as Assertions and Triggers, Views (Virtual Tables) in SQL, Additional features of SQL.	

UNIT – IV	**06 Hours**
Database Design Informal Design Guidelines for Relation Schemas, Functional Dependencies, Normal Forms Based on Primary Keys, General Definitions of Second and Third Normal Forms, Boyce-Codd Normal Form.	

Properties of Relational Decompositions, Algorithms for Relational Database Schema Design, Multi valued Dependencies and Fourth Normal Form, Join Dependencies and Fifth Normal Form, Inclusion Dependencies, Other Dependencies and Normal Forms.

UNIT – V	**07 Hours**
Transaction Management The ACID Properties, Transactions and Schedules, Concurrent Execution of Transactions, Lock- Based Concurrency Control, Performance of locking, Transaction support in SQL. Introduction to crash recovery, 2PL serializability and recoverability. Introduction to lock management, Lock conversions, Dealing with Deadlocks, specialized locking techniques, Concurrency control without locking to ARIES, The log, Other recovery – related data structures, The write –ahead Log protocol, Check pointing, recovering from a system crash, Media recovery, Other algorithms and Interaction with concurrency control.	

TEXT BOOKS:

1.	Elmasri and Navathe: Fundamentals of Database Systems, 5th Edition, Addison-Wesley, 2007(Chapters 1, 2, 3 except 3.8, 5, 6.1 to 6.5, 7.1, 8, 10,11)
2.	Raghu Ramakrishnan and Johannes Gehrke: Database Management Systems, 3rd Edition, McGraw-Hill, 2003.(Chapters 16, 17.1, 17.2, 18)

REFERENCE BOOKS:

1.	Silberschatz, Korth and Sudharshan: Data base System Concepts, 5th Edition, Mc-GrawHill, 2006.
2.	C.J. Date, A. Kannan, S. Swamynatham: A Introduction to Database Systems, 8th Edition, Pearson education, 2006.

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Identify the fundamentals of database technologies and its different architectures.
CO2.	Analyse the relational model concepts and Design relational algebraic expressions for SQL queries
CO3.	Illustrate the various concepts of SQL and Design SQL Queries to perform CRUD (Create, Retrieve, Update and Delete) operations on database.
CO4.	Apply the database design with Normalization concepts, Functional Dependencies, Normal Forms and Relational decomposition.
CO5.	Analyze the various criteria's of transaction management techniques, concurrency control algorithms and Compare various locking mechanisms, Crash and recovery methods.

Following are the case studies given to the students. They have to design the ER model by carefully the go thru these cases by making analysis.

1. A university registrar's office maintains data about the following entities:
   - Courses, including number, title, credits, syllabus, and prerequisites;
   - Course offerings, including course number, year, semester, section number, instructor(s), timings, and classroom;
   - Students, including student-id, name, and program;
   - Instructors, including identification number, name, department, and title.

Further, the enrolment of students in courses and grades awarded to students in each course they are enrolled for must be appropriately modelled.
Construct a E-R diagram for registrar's office. Document all assumptions that you make about the mapping constraints

2. Design an E-R diagram for keeping track of the exploits of your favourite sports team. You should store the matches played, the scores in each match, the players in each match, and individual player statistics for each match. Summary statistics should be modelled as derived attributes.

3. Banking scenario:

- A Bank is identified by name, Id number, location of the main office and an address
- A bank has many branches
- Each branch is identified by branchId, branch name, and an address
- A branch has many customers who hold the account
- A customer is identified by customerId, customer_name and an address
- Account is held by Customer and a customer may have single account and many accounts (SB account, Joint Account, Business Account)
- Each Account is identified by account_no, account_type, balance_amount
- A bank offers Loans to customers through branches
- Loans are identified by loanId, loanType (House, Car, Business, Personal) , loan_amount

Draw an ER diagram to represent this application

4. For each department the database contains a department number (unique), a budget value, and the department manager's employee number (unique). For each department the database also contains information about all employees working in the department, all projects assigned to the department, and all offices occupied by the department. The employee information consists of employee number (unique), the number of the project on which he or she is working, and his or her office number and phone number; the project information consists of project number (unique) and a budget value; and the office information consists of an office number (unique) and the area of the office in square feet. Also, for each employee the database contains the title of each job the employee has held, together with date and salary for distinct salary received in that job; and for each office it contains the numbers (unique) of all phones in that office.

Convert this hierarchical structure to an appropriate collection of normalized relations. Make any assumptions you deem reasonable about the dependencies involved.

Operating Systems

Contact Hours/Week	:	3+0(L+T)	Credits	:	03
Total Lecture Hours	:	39	CIE Marks	:	50
Total Tutorial Hours	:		SEE Marks	:	50
Course Code	:	3RMCA03			

Course Objectives:

This Course will enable students to:	
1.	Learn the fundamentals of Operating Systems.
2.	Learn the mechanisms of OS to handle processes and threads and their communication
3.	Learn the mechanisms involved in memory management in contemporary OS.
4.	Gain knowledge on distributed operating system concepts that includes architecture, mutual exclusion algorithms, deadlock detection algorithms and agreement protocols
5.	Know the components and management aspects of concurrency management
6.	Learn programmatically to implement simple OS mechanisms

UNIT – I	6 Hours

Introduction

What is an Operating System, Mainframe systems, Desktop systems, Multiprocessor system, Distributed system, clustered systems, Real Time systems, Handheld systems, Feature Migration, Computing environment, System Components, OS Services, System calls, System Programs, System Structure, Virtual Machines.

UNIT – II	7 Hours

Process Management:

Process concept, Process Scheduling, operation on Processes, Cooperating Processes, Inter process Communication, Threads-Overview, Multithreading models, Threading issues, CPU Scheduling-Basic concepts, Scheduling Criteria, Scheduling algorithms, Multiple processor scheduling, Real Time Scheduling.

UNIT – III	7 Hours

Process Synchronization and Deadlocks

The critical Section problem, Synchronization hardware, Semaphores, Classical problems of synchronization, Critical regions, Monitors, Deadlocks-System model, Deadlock, Characterization, Methods for handling deadlocks, Dead Lock Prevention, Deadlock avoidance, Deadlock Detection, Recovery from Deadlock

UNIT – IV	10 Hours

Storage Management:

Memory Management: Background, Swapping, contiguous allocation, Paging, Segmentation, Segmentation with paging.

Virtual Memory – Background, Demand Paging, Copy - on-write, Page replacement algorithms, Thrashing.

UNIT – V	8 Hours

Mass Storage Structure – Disk Structure, Disk Scheduling, Disk Scheduling algorithms, Disk Management.

File System Interface – File concept, Access Methods, Directory and Disk Structure, File System

mounting, File Sharing, Protection, File System Implementation – File system structure, File system Implementation, Directory Implementation, Allocation Methods, Free space management.

TEXT BOOK:

1.	Operating System Concept by Avi Silberschartz and Peter Galvin and Greg Gagne, 9th Edition, John Wiley and Sons, 2012. Chapter-1: 1.1 to 1.11, Chapter-2: 2.1 to 2.7, Chapter-3: 3.1 to 3.6, Chapter-4: 4.1 to 4.3, Chapter-5: 5.1 to 5.8, Chapter-6: 6.1 to 6.3, Chapter-7: 7.1 to 7.7, Chapter-8: 8.1 to 8.6, Chapter-9: 9.1 to 9.6 (except 9.5), Chapter-10: 10.1 to 10.5, Chapter-11: 11.1 to 11.6, Chapter-12: 12.1 to 12.5.

REFERENCE BOOKS:

1.	Operating System concepts and design by Milan Milankovie II Edition McGrawHill 2001.
2.	Operating Systems by Harvey M Deital, 3rd Edition, Addison Wesley, 2010
3.	Operating Systems – A Concept based Approach, D.M. Dhamdhere, 2nd Edition, Tata McGraw Hill, 2006.

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Identify the services, functions and structure of different operating system
CO2.	Analyze the various inter process communication mechanisms and different techniques for solving CPU scheduling problems.
CO3.	Analyze the problems that occurs in the design of OS based knowledge gained through process synchronization techniques and design a deadlock free system by applying the principles of concurrency and deadlock related algorithms.
CO4.	Identify and Analyze the performance of different memory management techniques and page replacement algorithms.
CO5.	Compare various disc scheduling algorithms and file management techniques.

Java Programming

Contact Hours/Week	:	5+2(L+T)	Credits	:	04
Total Lecture Hours	:	39	CIE Marks	:	50
Total Tutorial Hours	:	26	SEE Marks	:	50
Course Code	:	3RMCA04			

Course Objectives:

This Course will enable students to:	
1.	Familiarize the basic concepts of Java Programming constructs and OOPs concepts.
2.	Grasps the significance of Generalization and runtime polymorphism applications, usage of Packages, Interfaces, Exceptions and Multithreading
3.	Learn to apply Wrappers, Auto boxing and I/O Operations

UNIT – I	6 Hours

Fundamentals of Java Language

Fundamentals of Java Language – Introduction to Java, History of Java, Oak, Java Features, Difference between Java, C and C++, The Java Vtual Machine, Java Program Structure, Java Tokens, Data Types in Java, Arrays, Operators, Control Stiratements – Selection Statement, Iteration Statement, Switch Statement

Creating Classes and Methods

Defining Classes, Creating instance and Classes Variables, Defining instance Variables, Creating Objects, this Keyword, Passing Arguments to methods, Accessing Class Members, Constructors, Default (implicit), Parameterized Constructor, Constructor Overloading, Java.lang.Object, toString(), overriding equals() method, static keyword, Nesting of Methods

UNIT – II	8 Hours

Inheritance:

Extending a Class, Defining a subclass, Subclass Constructor, Multilevel Inheritance, Hierarchical Inheritance, super keyword, Visibility Control, public access, private access, protected Access, Creating Java Applications, Overriding Methods, final keyword, finalize() method, Dynamic Method Dispatch, Java Applications and Command Line arguments, Abstract methods and Abstract Classes, Inner Classes, Access Control for the methods and variables

Interfaces

Defining an Interface, implementing Interfaces, Nested Interfaces, Applying Interfaces, Variables in Interfaces, Interfaces can be Extended, Implementing Multiple Inheritance concept through Interfaces, Differences between the Abstract Class and Interfaces, Programming examples.

UNIT – III	8 Hours

Packages and Wrapper Classes

Java API Packages, The import Statement, Using System Packages, Naming Conventions, CLASSPATH Settings, Creating and using Packages, Importing from other Packages, Access protection in Packages, Example for package.

Introduction to Wrapper Classes: The Numeric types(Byte class, Short class, Integer class, The Long class, The Float class, The Double class), The Character class, The Boolean class

Enumerations, Autoboxing and Annotations (Metadata)

Enumerations: Enumeration Fundamentals, the values() and valueOf() Methods, Enumerations Inherit Enum, Example.

Autoboxing: Autoboxing and Methods, Autoboxing / Unboxing Boolean and Character values

Annotations (MetaData): Annotation Basics, Built – in annotations, custom annotations, specifying a Retention Policy,

UNIT – IV	9 Hours
Introduction to Strings, String Constructors, The Length() method, String Operations[String Literals, String Concatenation, String Concatenation with other data types, String Conversion and toString()], String Extraction[charAt(), getChars(), getBytes(), toCharArray()], String Comparison[equals() and equalsIgnoreCase(), regionMatches(), startsWith() and endsWith(), equals() versus ==, compareTo()], Searching Strings[indexOf(), lastIndexOf()], Modifying a String[substring(), concat(), replace(), trim()], Changing the case of characters within a String[toUpperCase(), toLowerCase()], String Buffer – Constructors, length(), capacity(), ensureCapacity(), setLength(), charAt(), setCharAt(), getChars(), append(), insert(), reverse(), delete(), deleteCharAt(), replace(), substring() methods, String Builder **Exception Handling** Exceptions, Managing Exceptions, Syntax of Exception handling, Java Keywords for handling Exception(try, catch, throw, throws and finally), Multiple Catch Statements, Using finally Block (Statement), Understanding the usage of throw and throws clause, Declaring methods that might throw exceptions, Creating and throwing user Exceptions. Programming examples.	

UNIT – V	8 Hours
Java I/O Streams File: Directories, Using list() and listFiles() methods, Creating Dirctrories, The Stream Classes: Byte Stream and Character Stream, Byte Streams: Input Stream Classes, Output Stream Classes Character Streams: Reader Classes, Writer Classes, Creating Byte Streams using FileInputStream / FileOutputStream Classes, Creating Character Streams using FileReader / FileWriter Classes, Accepting the input from the keyboard using InputSreamReader, BufferedReader and System.in **Multi Threading** Multi – Threading Fundamentals, The Main Thread, Creating a Thread(Implementing Runnable Interface, Extending Thread Class), Creating a Multiple Threads, Using isAlive() and join() methods, Thread Creation Diagram, The Life Cycle of a Thread, Thread Priorities, Thread Synchronization, Producer – Consumer Problem and wait() and notify() Methods, Suspending and Resuming Threads	

TEXT BOOK:

1.	Herbert Schildt: The Complete Reference JAVA, 8th Edition, Tata McGraw Hill, 2013.

REFERENCE BOOKS:

1.	Y. Daniel Liang: Introduction to JAVA Programming, 6th Edition, Pearson Education, 2007
2.	Herbert Schildt: Java Fundamentals – A Comprehensive Introduction, Tata McGraw Hill Edition 2013

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Describe the concept of OOP as well as the purpose and usage principles of inheritance, polymorphism, encapsulation and method overloading.
CO2.	Identify classes, objects, members of a class and the relationships among them needed for a specific problem.
CO3.	Exemplify the usage of Packages, Enumerations, Wrappers, Auto boxing and Annotations in building efficient applications.

CO4.	Develop programs using String, StringBuffer and Exception Handling.
CO5.	Demonstrate the ability to use I/O operations, Threads and synchronization in java.

Self-study component: Students are required to do any of the following (for 7 marks):

1. For the self – study component, students should submit a report for having developed application for the following (Not a limitation). They have to design the front – end by AWT or Swing components and for backend they have to use file concept or JDBC
   Mail Server
   Print Server
   Web Server
   Proxy Server
2. Students have to give a seminar on a chosen topic pertaining to Programming with Java.

Management Information Systems and E Commerce

Contact Hours/Week	:	3+0(L+T)	Credits	:	03
Total Lecture Hours	:	39	CIE Marks	:	50
Total Tutorial Hours	:		SEE Marks	:	50
Course Code	:	3RMCAE3			

Course Objectives:

This Course will enable students to:	
1.	Understands the distinction between data, information and its classification, various methods for collection of information
2.	Learn to describe how MIS support its business process and operations.
3.	Learn DSS, and types of client-server architectures
4.	Describe how the problems of managing data resources in a traditional file environment are solved by a ERP system.
5.	Learn all phases of E-Commerce and modes of payments of E-Commerce.

UNIT – I	8 Hours

Management Information System in a digital firm:

MIS: Concept, MIS Definition, Role of MIS, Impact of MIS, MIS & the user, Management as a control system, MIS: A support to management, management effectiveness & MIS, Organization as a system, MIS: organization effectiveness, MIS for a digital firm.

Decision making and DSS

Decision making concepts, decision making process, decision analysis by analytical modeling, behavioral concepts in decision making, organizational decision making, MIS and decision making. DSS: concept & philosophy, DSS models: behavioral, management science & operations research models, GDSS

UNIT – II	8 Hours

Information and Knowledge

Information concepts, Information: A quality product, classification of information, methods of data and information collection, value of the information, general model of a human as an information processor,, summary of information concepts and their implications, knowledge and knowledge management systems

Management of Global Enterprise

Enterprise management system, ERP System, supply chain management, Customer relationship management, EMS and MIS

UNIT – III	8 Hours

Client-Server architecture and E-Business technology

Client-server architecture, client-server implementation strategies, Service oriented architecture, Introduction to E-business, Models of E-business, Internet & WWW, Intranet/Extranet, Security in e-business, Electronic payment systems, Impact of web on strategic management, web enabled business management.

UNIT – IV	8 Hours

E-Commerce techniques and Issues

Introduction to Active Server Pages (ASP), Building an E-Commerce Web Site, E-Commerce Payment Systems, E-Commerce Marketing Techniques, Building product catalogue, Search product catalogue, Web Spider and search agent, Ethical, Social and Political Issues in E-Commerce

UNIT – V	7 Hours

Internet Communication

Transaction Systems, Shopping Carts, XML, E-Commerce Applications: Business-to-Consumer (B2C), Consumer-to-Consumer (C2C), Business-to-Business (B2B), Digital Government, Marketplaces, and Communities, Security and Encryption, Web Security.

TEXT BOOK:

1.	Waman S Jhawadekar: Management Information Systems, 5th Edition, McGraw Hill.
2.	Turban, Rainer, and Potter, Introduction to E-Commerce, second edition, 2003
3.	. H. M. Deitel, P. J. Deitel and T. R. Nieto, E-Business and E-Commerce: How to Programe, Prentice hall, 2001

REFERENCE BOOKS:

1.	Henry Chan, Raymond Lee, Tharam Dillon, Elizabeth Chang: E-Commerce Fundamentals and Applications, Wile India Edition
2.	Ralph M Stair and George W Reynolds: Principles of Information Systems, 7th Edition, Thomson, 2010
3.	Steven Alter: Information Systems - The Foundation of E-Business, 4th Edition, Pearson Education, 2001
4.	Rahul De, Managing Information Systems in Business, Government and Society, , Wiley India.

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Recognize and demonstrate the value, quality and the role of various types of information and its usage in the business systems
CO2.	Interpret, translate and demonstrate the roles, impact and effectiveness of Management information systems in organizations,
CO3.	Relate, translate, analyse and implement the process of decision making with the support of decision support technology in the business organizations at all levels.
CO4.	Interpret, contrast, translate and demonstrate the initiatives of Enterprise Resource Planning in improving the performance of the business systems
CO5.	Recognize, interpret and apply the contemporary e-commerce concepts and terminology

Software Project Management

Contact Hours/Week	:	3+0(L+T)	Credits	:	03
Total Lecture Hours	:	39	CIE Marks	:	50
Total Tutorial Hours	:		SEE Marks	:	50
Course Code	:	3RMCAE4			

Course Objectives:

This Course will enable students to:	
1.	Deliver successful software projects that support organization's strategic goals
2.	Match organizational needs to the most effective software development model
3.	Plan and manage projects at each stage of the software development life cycle (SDLC)
4.	Create project plans that address real-world management challenges
5.	Develop the skills for tracking and controlling software deliverables

UNIT – I	7 Hours

Introduction to Project Management

Importance of software project management: What is a project? Problems with Software Projects What are Project Management? Stages of Project. The Feasibility Study Planning. Project Execution. The Stakeholder of Project. All parties of project. The Role of Project Manager. Project Management Framework.

Project Planning

Integration Management. what is Integration Management. Project Plan Development. Plan Execution. Scope Management. what is Scope Management?. Methods for Selecting Projects. Project Charter. Scope Statement. Work Breakdown Structure. Stepwise Project Planning Overview. Main Steps in Project Planning

UNIT – II	9 Hours

Project Scheduling

Time Management. Importance of Project Schedules. Schedules and Activities. Sequencing and Scheduling Activity. Project Network Diagrams. Network Planning Models. Duration Estimating and Schedule Development. Critical Path Analysis. Program Evaluation and Review Technique (PERT).

Project Cost Management

Importance and Principles of Project Cost Management. Resource Planning. Cost Estimating. Types of Cost Estimates. Expert Judgment. Estimating by Analogy. COCOMO Model. Cost Budgeting. Cost Control.

UNIT – III	8 Hours

Project Quality Management

Quality of Information Technology Projects. Stages of Software Quality Management Quality Planning. Quality Assurance. Quality Control. Quality Standards. Tools and Techniques For Quality Control

Project Human Resources Management

What is Project Human Resources Management?. Keys to Managing People.

Organizational Planning. Issues in Project Staff Acquisition and Team Development.

UNIT – IV	7 Hours

Project Communication Management

Communications Planning. Information Distribution. Performance Reporting. Administrative Closure. Suggestions for Improving Project Communications.

UNIT – V	7 Hours
Project Procurement Management Importance of Project Procurement Management. Procurement Planning. Solicitation. Source Selection. Contract Administration. Contract Close-out.	

TEXT BOOK:

1.	Kathy Schwalbe, "Information Technology Project Management",International Student Edition, THOMSON Course Technology, 2003
2.	Bob Hughes and Mike Cotterell, "Software Project Management", Third Edition, Tata McGraw-Hill.

REFERENCE BOOKS:

1.	Basics of Software Project Management, NIIT, Prentice-Hall India, 2004.
2.	Pankaj Jalote, Software Project Management in Practice, Pearson Ed., 2002
3.	S.A. Kelkar, Software Project Management, a Concise Study, Revised Edition, Prentice-Hall India, 2003

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Adopt the practices and methods for successful software project management
CO2.	Identify techniques for requirements, policies and decision making for effective resource management
CO3.	Apply the evaluation techniques for estimating cost, benefits, schedule and risk
CO4.	Devise a framework for software project management plan for activities, risk, monitoring and control
CO5.	Devise a framework to manage people

Computer Graphics

Contact Hours/Week	:	3+0(L+T)	Credits	:	03
Total Lecture Hours	:	39	CIE Marks	:	50
Total Tutorial Hours	:		SEE Marks	:	50
Course Code	:	3RMCAE7			

Course Objectives:

This Course will enable students to:	
1.	To acquire knowledge to adopt green computing practices to minimize negative impacts on the environment.
2.	To examine technology tools that can reduce paper waste and carbon footprint by user.
3.	To understand how to minimize equipment disposal requirements.
4.	To gain skill in energy saving practices in their use of hardware.

UNIT – I	8 Hours

Overview Of Graphics Systems

Video Display devices. Raster-scan Systems, Random - Scan Systems. Graphics monitors, Input Devices.

Output Primitives

Points and Lines, Line-Drawing Algorithms. Line Function, Circle-Generating Algorithms. Filled-Area Primitives. Fill-Area Functions. Cell Array, Character Generation.

UNIT – II	8 Hours

Attributes of Output Primitives

Line Attributes, Color and Grayscale levels, Area-Fill Attributes. Character Attributes

Two-Dimensional Geometric Transformations

Basic Transformations. Matrix Representations and Homogeneous Coordinates. Composite Transformations, Other Transformations, Transformations Between Coordinate Systems

UNIT – III	8 Hours

Two-Dimensional Viewing

Window-to-Viewport Coordinate Transformation. Two-Dimensional Viewing Functions, Clipping Operations, Point Clipping, Line Clipping, Polygon Clipping. Text Clipping, Exterior Clipping

Interactive Picture - Construction Techniques

UNIT – IV	8 Hours

6. Three-Dimensional Geometric And Modeling Transformation

Three Dimensional Display Methods, Translation, Rotation. Scaling, Other Transformations, Composite Transformations

UNIT – V	8 Hours

Three-Dimensional Viewing

Viewing Pipeline, Viewing Coordinates, Projections – Parallel & Perspective Projections

Visible-Surface Detection Methods

Classification of Visible-Surface Detection Algorithms, Back-Face Detection, Depth-Buffer Method, Scan-line method.

TEXT BOOK:

1.	Computer Graphics: C version - Donald Heam and M.Pauline Baker, Pearson Education – Second Edition – 2003

REFERENCE BOOKS:

1.	Computer Graphics - A Programming Approach - Steven Harrington, Tata McGraw Hill - Second Edition,
2.	Computer Graphics - Principles and Practice - Folcy, VanDam, Fcincr & Hughcs. Pearson Education 2nd Edition.

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Illustrate the various output primitives and graphics systems.
CO2.	Apply the attributes of output primitives and 2D geometric transformations.
CO3.	Illustrate the various viewing and clipping techniques with widow to viewport transformation for two dimensional objects.
CO4.	Represent the objects and apply transformation using 3D graphics concepts.
CO5.	Illustrate projections and clipping using concepts of visible surface detection methods.

Mini Project-I

Contact Hours/Week	:	3	Credits	:	2.0
Total Lecture Hours	:	-	CIE Marks	:	50
Total Tutorial Hours	:	-	SEE Marks	:	50
Course Code	:	3RMCAL1			

Course Objectives:

This Course will enable students to:	
1.	Develop problem statement
2.	Understand requirement analysis and identify design methodologies
3.	Use computing tools and appropriate programming language

Sl.no	Programs
	The intent of mini project work is to bring out creativity and innovation in students. Included in the course at third semester level with an expectation that it will help students to make transitions from pre-set practical to open ended computational work. Students are expected to do the project work using any of the languages they learnt thus far and may use VB, Visual C# or any other tool for creating front end. The maximum number of students in a batch is TWO. The projects may include (but not limited to) • Graphical Simulation of algorithms. • Simulation of physics problems • Student information system • Railway/airline/Bus reservation system • Online quiz /entrance exam • Static web site development, shopping carts application. • Computer network simulation • Simulation of Biological systems • Simulation of Operating System

Course Outcomes:

After the completion of this course, students will be able to:	
CO1	Review the literature to analyze and Identify the problem statement
CO2	Design the solution for the problem and Apply appropriate techniques and tools to develop the solution
CO3	Test the working of the product or software
CO4	Develop effective communication skills and presentation skills to Demonstrate the work in a team/individual
CO5	Apply ethical principles and inculcate lifelong learning.

RDBMS Lab

Contact Hours/Week	:	2	Credits	:	1.0
Total Lecture Hours	:	-	CIE Marks	:	50
Total Tutorial Hours	:	-	SEE Marks	:	50
Course Code	:	3RMCAL2			

Course Objectives:

This Course will enable students to:	
1.	Explore all SQL language commands related to DDL, DML, DCL, TCL.
2.	Create and implement the databases for the domains like healthcare, energy, agriculture, telecom, tourists and others.
3.	Create portfolios that demonstrates their proficiency in RDBM concepts and skills.

Sl.no	Programs
1.	Consider the Insurance database given below. The primary keys are underlined and the data types are specified. DRIVER (driver-id#:String, driver_name : string, address: string) CAR (Regno: String, car_model:string, yearofpurchase:int) ACCIDENT (Report-Number: int, dateofaccident:date, location: string) CAR_OWNER (driver-id#:string, Regno: String) PARTICIPATED (driver-id#:string, report-number: int, damage_amount:int) 1. Create the above tables by properly specifying the primary keys and the foreign keys. 2. Enter at least five tuples (i.e. records) for each relation (i.e. table). Execute SQL queries for the following requirements: Query-1 : List out driver name, address, model of the car, year of purchase for a particular owner : Query-2 : List out driver name, model of the car, year of purchase, place of accident and date of accident occurred at a particular place. Query-3 : List the number of accidents done by a particular driver : Query-4 : List out the accident details (i.e. driver name, model of the car, date of accident, place of accident, damage amount) where the damage amount exceeds Rs. 25,000 Query-5 : Find out the place of accident where the number of accidents occurred are more than one. Query-6 : Find the total number of people who owned car that were involved in an accident in the year 2009 Query-7 : Find the number of accidents in which car belongs to specific model were involved.
2.	Consider the following relations for an order processing database application in a company. CUSTOMER (cust#: int, cname: String, City: String) ITEM (item#: int, ItemName: string, unitprice: int) WAREHOUSE (warehouse#: int, warehouse_name: string, city: String)

	CORDER (order#: int, odate: date, cust#: int, ord-Amt:int) ORDER-ITEM (order#: int, Item#: Int, Qty: int) SHIPMENT (order# int, warehouse#: int, ship-date: date) Create the above tables by properly specifying the primary keys and the foreign keys. Enter at least 5 to 10 tuples for each relation. Execute SQL queries for the following requirements: Query-1: List out the details of orders, i.e. orderno, warehouse name, shipdate for a particular city : Query-2: List out customer name, number of orders they made, their total ordered amount and their average ordered amount for all the customers: Query-3: Find out the warehouse and number of orders they obtained from different customers. Query-4: Find out the warehouse, which has got number of orders more than 2. Query-5: Find out the customer who has ordered the number of items more than 3
3	Consider the following database of student enrollment in courses and books adopted for each course. STUDENT (regno#: string, name: string, major: string, bdate: date) COURSE (course#: int, cname: string, dept: String) TEXT (book_ISBN#: int, book_title: string, publisher: string, author: string) ENROLL (regno#: string, course#: int, sem: int, marks: int) BOOK_ADOPTION (course#: int, sem: int, book_ISBN: int) Create the above tables by properly specifying the primary keys and the foreign keys Enter at least 7 to 10 records to each table. 0Execute SQL queries for the following requirements: Query-1 List out the student details, and their course details. The records should be ordered in a semester wise manner. Query-2 List out the student details under a particular department whose name is ordered in a semester wise Query-3 List out all the book details under a particular course Query-4 Find out the Courses in which number of students studying will be more than 2. Query-5 Find out the Publisher who has published more than 2 books. Query-6 Find out the authors who have written book for I semester, computer science course. Query-7 List out the student details whose total number of months starting from their date of birth is more than 225 Query-8: Find out the course name to which maximum number of students have joined
4	For the same previous database tables (i.e. Practical – 3) STUDENT (regno#: string, name: string, major: string, bdate: date) COURSE (course#: int, cname: string, dept: String) TEXT (book_ISBN#: int, book_title: string, publisher: string, author: string) ENROLL (regno#: string, course#: int, sem: int, marks: int) BOOK_ADOPTION (course#: int, sem: int, book_ISBN: int) Execute SQL queries for the following requirements : Query 1 :List out the students those who have born in the month of April Query 2 : List out all the students whose age lies between 17 and 19

	Query 3: List out all the student details whose marks greater than 750 Query :4 Increase the marks by 10 % to students belongs to Computer Science Department and I semester Students. Query :5 Find out the department in which no books are available : Query :6 Find out the department to which no Students have joined : Query :7 Remove all the details related to the Electronics department
5.	Consider the following Employee database of working in a department and getting salary grade. Employee (empno: number, empname: string, job: string, manager: number (self reference key), hiredate: date, salary: number, commision: number, deptno number(foreign key)) Department (deptno: number, deptname: string, location: string) SalaryGrade (grade: number, LowSalary: number, highSalary: number) i. Create the above tables by properly specifying the primary keys and the foreign keys ii. Enter atleast ten tuples for each relation. Execute SQL queries for the following requirements : Query 1: List details of employees who have joined before 30 Sep 81. Query 2: List employee names those who have joined between the months June to december of the year 1981. Query 3: List the name and designation of the employee who does not report to anybody. (i.e. doesn't have any managers) Query 4: List the names of employees whose names should have 'A' as the third character. Query 5: List the employees whose name should not start with letter 'A' and should not end with the letter 'A' but it should be there in the name other than First and Last character. Query 6:. List the names of employees who have finished their 35 years of experience in the company. Query 7: List the employee name, salary, PF, HRA, DA and gross; order the results in the ascending order of gross. (PF is 10%, HRA is 50%, DA is 30% of the salary and gross is sum of salary, PF, HRA & DA) Query 8: List the department name, number of employees, total_salary, average salary, maximum salary and minimum salary in each of the department. Query 9: List the total salary, maximum salary, minimum salary and average salary of the employees according to job wise
6.	For the same previous database tables (i.e. Practical – 5) Employee (empno: number, empname: string, job: string, manager: number (self reference key), hiredate: date, salary: number, commision: number, deptno number(foreign key)) Department (deptno: number, deptname: string, location: string) SalaryGrade (grade: number, LowSalary: number, highSalary: number) Execute SQL queries for the following requirements : Query 1: List details of the employees who have joined in the month of February. Query 2: List the employees having experience greater than 300 months. Query 3: List empno, empname, job, salary, deptname, and grade of the salary of all the employees except clerks. Query 4: List empno, empname, job, salary, deptname and grade of the salary with annual remuneration greater than 36000 Query 5: Find the job that was filled in first half of 1981 and the same job that was filled during second half of 1981

	Query 6: Find all the employees who have joined the company before their managers Query 7: List departments for which no employees exists. Query 8: In which year did most people join the company. Display the year and no of employees Query 9: List the department in which highest number of employees are working.

Course Outcomes:

After the completion of this course, students will be able to:	
CO1	Apply the knowledge of database management system development process and conduct the experiments using SQL queries to find the solution for a given database problem.
CO2	Analyze and Design solutions for database system components to meet the specified needs of online transaction processing and information systems like Banking systems, Ticket Reservation systems etc.
CO3	Develop coding skills for server side programming objects like procedures, functions, packages, triggers and assertions.

Java Lab

Contact Hours/Week	:	2	Credits	:	1.0
Total Lecture Hours	:	-	CIE Marks	:	50
Total Tutorial Hours	:	-	SEE Marks	:	50
Course Code	:	3RMCAL3			

Course Objectives:

This Course will enable students to:	
1.	Write Java application programs using OOP principles and proper program structuring
2.	Demonstrate the concepts of polymorphism and inheritance
3.	Write Java programs to implement error handling techniques using exception handling , IO Streams and Multithreading

Sl.no	Programs
1.	Programs on classes and objects
2.	Programs on Inheritance
3	Programs on abstract class and inner class
4	Programs on interfaces
5.	Programs on packages
6.	Programs on string and string buffer classes
7.	Programs on Collections
8.	Programs on exception handling
9.	Programs on IO streams
10.	Programs on Multithreading

Course Outcomes:

After the completion of this course, students will be able to:	
CO1	Demonstrate the fundamental data types and constructs of Java Programming by writing executable/interpretable programs.
CO2	Illustrate the object oriented principles with the help of java programs.
CO3	Write java programs to demonstrate the concepts of interfaces, inner classes and I/O streams.
CO4	Develop reusable and efficient applications using inheritance and multi-threading concepts of java.

Soft Skills (Mandatory Course)

Contact Hours/Week	: 0+0(L+T)	Credits	: 00
Total Lecture Hours	: 24	CIE Marks	: 100
Total Tutorial Hours	:	SEE Marks	:
Course Code	: MC06		

Course Objectives:

This Course will enable students to:	
1.	Effectively Communicate individually and in team.
2.	Demonstrate Presentation skills
3.	Prepare Action plan for achieving goals
4.	Prepare Resume.

UNIT – I	6 Hours

Introduction:

- Role of soft-skills, its necessity in the present day business world and work place

Basic communicative English

- Communication process
- Barriers to communication
- Non – verbal Communication
- Body language, gestures
- Voice modulation
- Eye contact and proximity

(Assessment of individual student's ability to communicate)

UNIT – II	3 Hours

Presentation skills – using visual aids .

- Tips for preparing power point presentation
- Essential features of PPT

(Students are to be given individual topics for making a PPT and they are to be assessed after making a presentation for 3-4 minutes)

UNIT – III	6 Hours

Team building & team work .

- Stages in team formation
- Difference between team and group
- Elements of team work
- Team roles
- Conflict resolution

(The students in the class are divided into several small teams & tasks are assigned to team and the students' performance in the team are assessed)

Intrapersonal skills

- Emotional Management
- Building self confidence and self esteem
- Prioritization case study

UNIT – IV	6 Hours

SWOT Analysis & Time Management

SWOT:

- What is SWOT analysis?
- Areas of strengths and weaknesses of individual with respect to personal and professional competency.
- Analyzing SWOT of an individual
- Need and usefulness of SWOT analysis

Goal Setting:

- Long term and short term goal setting
- Prioritization of goals
- Action plan for achieving goals
- Professional goal setting

Time management:

- Tools & techniques for time management
- Time management matrix

Interpersonal skills

- Situational leadership
- Collaborative & competing approach, understanding the peers, case study and proactive group activities, conflict resolution and optimization
- Dale Carnage's principles of interpersonal skills.

(Assessment of students regarding their participation in group activities)

UNIT – V	3 Hours

Resume Writing

- Important and essential elements of a good resume
- Dos and don'ts in resume writing
- Business etiquette, email and telephone etiquettes

(Assessment of individual resume)

TEXT BOOK:

1.	Stephen Covey, Seven Habits of Highly Effective People, the bath press UK 2012
2.	Subhratho Bagchi, The professional, Penguin Random House, India, 2009

IV semester

Computer Networks

Contact Hours/Week	:	3+0(L+T)	Credits	:	03
Total Lecture Hours	:	39	CIE Marks	:	50
Total Tutorial Hours	:		SEE Marks	:	50
Course Code	:	4RMCA01			

Course Objectives:

This Course will enable students to:	
1.	Impart knowledge on of computer networks by going through basic terminologies and concepts.
2.	Study the conceptual and implementation aspects of network applications, including application layer protocols, clients, servers, processes and interfaces.
3.	Understand the principles as to how two entities can communicate reliably over a medium through series of complicated scenarios.
4.	Understand how forwarding and routing functions of the network layer.
5.	Uunderstand two fundamentally different types of link layer channels, broadcast channels and HFCs, To get clear idea about wireless Internet devices, mobility related services.
6.	Understand network security related algorithms and protocols.

UNIT – I	7 Hours
Computer Networks and the Internet: What is the Internet? The network edge, the network core, delay loss in packet switched networks, protocol layers and service models, networks under attack.	

UNIT – II	8 Hours
Application layer, principles of network applications, the web and HTTP, file transfer: FTP, electronic mail in the internet. **Transport layer services**, Introduction to transport layer services, multiplexing and demultiplexing, connectionless transport, UDP, principles of reliable data transfer, connection oriented transport.	

UNIT – III	8 Hours
The Network layer, Introduction, virtual circuit and data gram networks, what is inside router?, The IP, forwarding and addressing in the internet, routing algorithms.	

UNIT – IV	8 Hours
The link layer: Introduction and services, error detection and correction techniques, multiple access links and protocols. Ethernet frame structure	

UNIT – V	7 Hours
Physical Layer: Guided transmission media, magnetic media, twisted pairs, coaxial cable, power lines, and fiber optics. Wireless and mobile networks. Security in computer networks, What is network security?, principles of cryptography	

TEXT BOOK:

1.	James F.Kurose, Keith W. Ross: Computer Networking-A Top-Down Approach, 5th Edition, Pearson Education. { 1.1-1.6, 2.1-2.4, 3.1-3.5, 4.1-4.5, 5.1-5.5 }
2.	Andrew S Tenenbaum and David J Wetherall, Computer Networks, 5th Edition, Pearson Publishing. {Unit 5: 2.2, 2.3, 2.4, 2.7}

REFERENCE BOOKS:

1.	Behrouz A. Forouzan: Data Communications and Networking, 4th Edition, Tata McGraw-Hill, 2006.
2.	Nader F. Mir: Computer and Communication Networks, Pearson education, 2007.
3.	Larry L. Peterson and Bruce S. David, Computer Networks- A Systems approach, 4th edition, Elsevier, 2007.

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	***Elaborate on computer networks in a wholesome manner in terms of its edge, core, the infrastructure, and the functionalities of protocols in different layers.***
CO2.	***Illustrate the functionalities of the predominant protocols of application layer and transportation layer and how they enable connection oriented and connectionless data communication.***
CO3.	***Differentiate IPv4 and IPv6 in terms of their format, enhancements, and functionalities. Also able to decipher about the functioning of routers, as well as routing protocols***
CO4.	***Apply error checking algorithms on a given data bit string and also to illustrate various functionalities of link layer protocols.***
CO5.	Present on physical layer primitives, wireless communication and also to explain the secure communication related concepts.

Self-study component/ practical component: Students are required to do any of the following (for 4 marks):

- Submit a report of having used any network simulator and for having run any protocol that they have learnt in theory.
- Present a seminar on recent topic in computer networks (not included in syllabus) and submit a report.

Programming with Python

Contact Hours/Week	:	3+2(L+T)	Credits	:	04
Total Lecture Hours	:	39	CIE Marks	:	50
Total Tutorial Hours	:	26	SEE Marks	:	50
Course Code	:	4RMCA02			

Course Objectives:

This Course will enable students to:	
1.	Differentiate the syntax of Python from other programming languages
2.	Get familiar in writing simple programs using Python language
3.	Understand various data structures existing in Python library including string, list, dictionary etc.,
4.	Teach different libraries for scientific and graphics applications.
5.	Motivate the students to build real-world applications using oops, files and exception handling available in Python Language

UNIT – I	7 Hours

Introduction

Python Introduction, History of Python, Python features, Python Installation, Python Environment Variables, Running Python, Simple Programs, Python Identifiers, Reserved words, Lines and Indentation, Multi line statements, Quotation in Python, Comments in Python, Command line arguments, Assigning values to the variables, Multiple assignment, Standard data types, Type Conversion, Operators in Python, Operators Precedence, Decision Making, Looping, Loop Control statement.

User defined functions, function basics, function definition(def) statements, function call, local variables, Scope basics- scope rules, name resolution, the built-in scope, global statement, scope and nested functions, non-local statement. Arguments, arguments passing basics, special argument matching modes.

Case Study: Check the number is odd or even by using if conditions, Display the largest number among three numbers using different if conditions, displaying and checking prime and to display the Twin prime number, Armstrong, Ramanujan numbers, Fibonacci series and Multiplication table by using looping constructs, Converting decimal to octal, hexadecimals and vice versa by using functions.

UNIT – II	8 Hours

Python Built-in Data Structures

String: Assigning values in strings, String manipulations, String special perators, String formatting operators, Triple Quotes, Raw String, Unicode String, Build-in-String methods. **Lists**- Introduction, Accessing values in list, List manipulations, List
Operations, Indexing, slicing & matrices. **Tuples**- introduction, Accessing values, Tuple functions, **Dictionary**- Introduction, Accessing values, Dictionary functions.

Case Study: With the help of string array or list, display a simple calendar for a given year without using the calendar Unit.

UNIT – III	8 Hours

Object Oriented Programming

User defined functions, function basics, def statements, definition, call, polymorphism in python, local variables, **Scope basics**- scope rules, name resolution, the built-in scope, global statement, scope and nested functions, non-local statement. **Arguments**, argument-passing basics, special argument matching modes.

Class coding: basics, class, method, inheritance, classes vs Unit, operator overloading, multiple inheritance.

Case Study: Design functions to calculate mean and median for a given list of data, Demonstrate classes and inheritance (account transactions details of the customer using Account class and student mark details of UG, PG & Research Student and Mark classes)

UNIT – IV	8 Hours

Files & Exception Handling

Files, persistence, reading and writing, format operator, file names and paths, catching exceptions, file object attribute, manipulations of the files, copying content from one file to another file.

Exception, Handling Exception, try/except/else statement, try/finally statement, unified try/except/finally, raise and assert statement. **Exception objects**, Exceptions: back to the future, built-in Exception classes. **Designing with Exceptions**: Nesting exception handlers, Exception Idioms.

UNIT – V	8 Hours

Graphics databases and SQL in python

Graphics: The object of objects, simple graphics programming, uses of graphical objects, graphing future value, choosing coordinates, interactive graphics, getting mouse clicks, handling textual input, GraphWin, Graphics and Entry objects, displaying images, generating colors, animation. What is a database? Creating a database table, demonstrating the four basic SQL commands like SELECT, INSERT, DELETE and UPDATE.

Case study: Python calculator, A calculator as an object, constructs the interface, processing buttons. drawing human face, Celsius-to-Fahrenheit convertor, Draw a car object in the window, Obtain a triangle shape in the window after clicking three times on the window area, Draw a color ball and move it n number of times over the screen according to continues mouse clicking, Design and implement a graphical user interface to perform any arithmetic operation.

TEXT BOOK:

1.	Mark Lutz, Learning Python, O'Reilly, 4th Edition, 2009.
2.	John M. Zelle, Jim Leisy, PYTHON Programming: An Introduction to Computer Science, Tom Sumner - Franklin, Beedle, 2004 (Chapter 5, 11.5)

REFERENCE BOOKS:

1.	Pavel Solin, Martin Novak, "Introduction to Python Programming", FEMhub Inc, 2012
2.	Jakob Fredslund, "Introduction to Python Programming", 2007
3.	John C. Lusth, "An Introduction to Python", The University of Alabama, 2011
4.	DaveKuhlman, "Introduction to Python", 2008
5.	Andrew N. Harrington, "Hands-On Python: A Tutorial Introduction for Beginners", Creative commons Attribution-Noncommercial-Share Alike 3.0
6.	Emmanuelle Gouillart, Gaël Varoquaux, "Python scientific lecture notes", EuroScipy turorial team, 2010

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Apply fundamental constructs like conditional statements, loops, and functions to design and test small python programs to large scale programs
CO2.	Implement Python data types like strings, lists, tuples, dictionaries and sets to solve basic programs to real world problems.
CO3.	Apply Object-Oriented Programming concepts like encapsulation, inheritance, and polymorphism and to develop OOP programs.
CO4.	Implement file handling techniques to create files, copy the file and to update the file and to manage exception handling.
CO5.	Develop Graphical User Interphase applications with supporting Database connection

Self-Study Component (Assignment):

Students need to develop a project using Python GUI and the connectivity using the back-end.

(or)

Students need to develop a game project using Python

Data Mining & Warehousing

Contact Hours/Week	: 3+2(L+T)	Credits	: 04
Total Lecture Hours	: 39	CIE Marks	: 50
Total Tutorial Hours	: 26	SEE Marks	: 50
Course Code	: 4RMCA03		

Course Objectives:

This Course will enable students to:	
1.	Understand the need of Data warehousing, data warehouse architecture, OAP operations and Data mining.
2.	Realize the significance of data pre-processing for data warehouse and data mining
3.	Learn the data mining algorithms mainly classification, association rule mining and clustering to discover the hidden patterns

UNIT – I	7 Hours
What is Data warehouse?, Differences between operational database systems, A multidimensional data model: data cubes, starts, snowflakes and fact constellations, Concept hierarchies, OLAP operations in multidimensional data model, Data warehouse architecture, a three tier data warehouse architecture, types of OLAP servers. What is data mining?	

UNIT – II	8 Hours
Why pre-process data, Descriptive data summarization – measuring the central tendency, dispersion of data, Data cleaning - missing values, noisy data, data cleaning as process, Data integration and Transformation , data reduction – data cube aggregation, attribute subset selection, discretization and concept hierarchy generation for numerical data.(problems on correlation ,binning, descriptive data summarization , normalization)	

UNIT – III	8 Hours
What is classification? What is prediction? Classification by decision tree induction algorithm, attribute selection method: information gain, gain ratio, Gini index. (Problems on information gain and Gini index). Lazy learners: k-nearest-neighbor classifier (problems on kNN). Prediction (problems on linear regression).Prediction Accuracy and error measures- classifier accuracy measure, predictor error measures, evaluating the accuracy of a classifier or predictor – holdout method and random sub sampling, cross validation, bootstrap. (problems on classification accuracy measures: confusion matrix)	

UNIT – IV	7 Hours
Basic concepts and a road map,Market basket analysis, frequent itemsets, maximal and closed itemsets, and association rules , Association Rule mining: Association Rule Mining Basic concepts, frequent item sets, The Apriory Algorithm, generating association rules from frequent item sets. Association Rule Mining without candidate generation -using FP-growth algorithm, mining frequent itemsets using vertical data format, mining multilevel association rules	

UNIT – V	8 Hours
Clustering: What is Cluster analysis? Types of data in cluster analysis-interval-scaled variables, binary variables, categorical, ordinal, ratio-scaled variables (Problems on computing distance between two objects). A categorization of major clustering methods, Partitioning Methods: The K-means method. (Problems on k-means clustering). Hierarchical methods: Agglomerative and Divisive hierarchical clustering, plotting Dendrogram, Measures for distance between clusters: Minimum distance, maximum distance, average distance (Clusters Problems on agglomerative clustering using these measures).	

TEXT BOOK:

1.	Jiawei Han & Micheline Kamber, Data Mining Concepts and Techniques , Morgan Kaufmann Publishers – Second Edition Unit I-Chapter 1& 3 (1.2, 3.1.1, 3.1.2, 3.2.1, 3.2.2, 3.2.3, 3.2.5,3.2.6, 3.3.2, 3.3.5,) Unit II – Chapter 2(2.1, 2.2.1,2.2.2, 2.3, 2.4, 2.5: 2.5.1,2.5.2, 2.6.1) Unit III – Chapter 6 (6.1, 6.3, 6.3.1, 6.3.2,6.3,3, 6.9:6.9.1, 6.11: 6.11.1, 6.11.2,6.12, 6.13) Unit IV Chapter 5(5.1.1,5.1.2, 5.1.3, 5.2.1, 5.2.2, 5.2.4,5.2.5, 5.3.1) Unit V - Chapter 7(7.1, 7.2.1-7.2.3, 7.3,7.4: 7.4.1, 7.5.1)

REFERENCE BOOKS:

1.	Pang-Ning Tan, Michael Steinbach, Vipin Kumar , Introduction to Data mining, Pearson Education

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Interpret the importance of data warehouse and OLAP operations.
CO2.	Apply data prepossessing steps for real world data applications.
CO3.	Design and deploy appropriate classification techniques.
CO4.	Apply Association rule mining for finding strong rules
CO5.	Analyse various clustering techniques and their applications.

Advanced Java Programming

Contact Hours/Week	:	3+2(L+T)	Credits	:	04
Total Lecture Hours	:	39	CIE Marks	:	50
Total Tutorial Hours	:	26	SEE Marks	:	50
Course Code	:	4RMCA04			

Course Objectives:

This Course will enable students to:	
1.	Understand and use Java Collection Framework elements
2.	Connect to a database using JDBC and associating with JDBC-ODBC bridge
3.	Use JDBC API statements to access database and metadata.
4.	Design and develop Enterprise applications using the Servlets technology
5.	Understand the details of Web services technologies: WSDL,SOAP

UNIT – I	9 Hours
The Collections Framework: Collections overview, Recent changes to Collections, The Collection Interfaces, The Collection Interface, The List interface, The Set interface, The Queue interface, The Collection classes: The ArrayList class, The LinkedList class, The HashSet class, The TreeSet class, Accessing a Collection via an Iterator, Using an iterator, The For-Each alternative to iterators, Storing user-defined classes in Collections	

UNIT – II	8 Hours
JDBC Objects: The concept of JDBC, JDBC Driver Types, JDBC Packages, A brief overview of JDBC process, Database Connection, The Connection, Associating the JDBC/ODBC bridge with the database. **Statement Objects:** Statement object, PreparedStatement object, CallableStatement, ResultSet, Reading the ResultSet, Scrollable ResultSet, Updatable ResultSet, Transaction processing, Metadata, ResultSet Metadata, Data types, Exceptions	

UNIT – III	8 Hours
Java Servlets Java servlets and Common Gateway Interface(CGI) programming, Benefits of using a Java servlet, A simple Java servlet, Anatomy of a Servlet, Deployment Descriptor, Reading data from a client, Reading Http request headers, Sending data to a client and writing the Http Response header, Working with cookies, Tracking sessions.	

UNIT – IV	7 Hours
Java Server Pages: JSP, Installation, JSP Tags: Comment tag, Declaration Statement tags, Directive tags, Expression tags, Scriptlet tags, Variables and Objects, Methods, Control statements, loops, Tomcat, Request string, Parsing other information, User Sessions, Cookies, Session Objects	

UNIT – V	7 Hours
Web Services Description Language(WSDL): Inside WSDL, The WSDL document, Type Element, Message Element, PortType Element, Binding Element, Port Element, Service Element. **Simple Object Access Protocol(SOAP):** SOAP one-way transmission primitive, SOAP request-response transmission primitive, SOAP binding element, SOAP operation element, SOAP body element, SOAP fault element, SOAP header element, SOAP address element. WSDL and HTTP binding, WSDL and MIME binding.	

TEXT BOOK:

1.	The Complete Reference J2EE , Jim Keogh, McGraw Hill **Education**, ISBN: 978-0-07- 052912-0 (Unit-2 to Unit-5 followed and page numbers given below) (Unit-2: Page Nos: 123-160, Unit-3: Page Nos: 347- 369, Unit-4: Page Nos: 379-395 , Unit-5: Page Nos: 684-700)
2.	The Complete Reference Java 7th edition, Herbert Schildt, McGrawHill publications ISBN: 978-0-07-163177-8 (Unit-1 : PageNos: 440-446, 448-456, 458-463)

REFERENCE BOOKS:

1.	Java™ 2 Enterprise Edition 1.4 Bible, Wiley Publishing, **ISBN: 0-7645-3966-3**
2.	Begining Java Server Pages :Wiley Publications, Vivek chopra, **ISBN: 076457485X**

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Design and demonstrate the usage of Java collections in developing modular and efficient programs.
CO2.	Construct Database connection and access Database by performing queries on a Builder tool
CO3.	Design and develop dynamic interactive web applications with client side HTML and server side Servlet API package elements and its services
CO4.	Design and develop interactive and dynamic Java Server Pages web applications and enhance the customized features of JSP Tags.
CO5.	Design and develop web services using WSDL and SOAP services across clients and servers

Intelligent Data Analytics

Contact Hours/Week	:	3+0(L+T)	Credits	:	03
Total Lecture Hours	:	39	CIE Marks	:	50
Total Tutorial Hours	:		SEE Marks	:	50
Course Code	:	4RMCAE1			

Course Objectives:

This Course will enable students to:	
1.	Get a overall view of data analysis by going through basic concepts
2.	Study data quality assessment and visualization techniques for data involving two attributes and for higher dimensional data.
3.	Introduce R-language.
4.	Understand principles of modeling by going through various data modeling techniques.
5.	Get clear idea about data preparation.
6.	Tstudy statistical concepts related to data analysis.

UNIT – I	8 Hours
Introduction to Data analysis, data and knowledge, intelligent data analysis, data analysis process, methods, tasks, tools, practical data analysis, data understanding and pattern finding, explanation finding, predicting the future. Project understanding, determine the project objective, assess the situation, and determine analysis goals.	

UNIT – II	8 Hours
Data Understanding-attribute understanding, data quality, data visualization, methods for one and two attributes, methods for higher dimensional data, correlation analysis, dealing with missing values.	

UNIT – III	9 Hours
Principles of Modeling, model classes, fitting criteria and score functions, error functions, measures of interestingness, algorithms for model fitting, closed form solutions, gradient method, combinatorial optimization, random search, errors- experimental, simple, model, algorithmic errors.	

UNIT – IV	8 Hours
Data Preparation, selection of data, feature selection, dimensionality reduction, record selection, cleaning of data, improving data quality, providing operability, impartiality and maximizing efficiency. Complex data types, data integration, vertical and horizontal data integration.	

UNIT – V	6 Hours
Statistics, terms and notations, tabular representations, graphical representations, measures for one dimensional data, for multidimensional data, principal component analysis.	

TEXT BOOK:

1.	Michael R. Berthod, Christian Borgelt, Frank Hoppner, Guide to Intelligent Data Analysis, Springer Series, II edition, 2010.

REFERENCE BOOKS:

1.	Charles M.Zudd, Garry H.Mcchelland, Carry S.Ryan, Data Analysis a Model comparison Approach, Routledge Publication, NY, 2009.
2.	Allan Agresty, An Introduction to categorical data analysis, 2nd edition, Wiley Publications.

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Elaborate on data analysis paradigms in general and CRISP-DM process model in particular.
CO2.	Apply appropriate data visualization technique and perform correlation analysis on the real world data.
CO3.	Develop appropriate model for the data using the suitable algorithm and validate the so
CO4.	Develop model using appropriate validation technique.
CO5.	Decide on appropriate method/ technique for data preparation and provide operability by assuring impartiality and integrity to the given data.
CO6.	Perform descriptive statistical analysis of the given data set in one and multiple dimensions.

Self-study component:

1. Students have to submit a report containing a detailed illustration of analysis of some real data. The analysis should be carried out using any of the open source analysis packages like R, KNIME, WEKA … The detailed report shall be evaluated for 7 marks.

Artificial Intelligence

Contact Hours/Week	:	3+0(L+T)	Credits	:	03
Total Lecture Hours	:	39	CIE Marks	:	50
Total Tutorial Hours	:		SEE Marks	:	50
Course Code	:	4RMCAE2			

Course Objectives:

This Course will enable students to:	
1.	Know the representations and algorithms used to build artificial intelligence systems
2.	To understand the role of the knowledge representation and reasoning.
3.	To achieve the knowledge in building the simple knowledge-based systems.
4.	Overviews to build the ability to apply knowledge representation, reasoning, and machine learning techniques to real-world problems.

UNIT – I	7 Hours
Introduction **What is AI?** **Intelligent Agents:** Agents and environment ; Rationality ; The nature of environment ; the structure of agents . **Problem solving:** Problem-solving agents ; Example problems ; Searching for solution; Uninformed search strategies.	

UNIT – II	8 Hours
Informed Search: Informed search strategies ; Heuristic functions ; **Adversial search:** Games ; Optimal decisions in games; Alpha-Beta pruning **Constraint satisfaction problems:** Defining Constraint satisfaction problems.	

UNIT – III	8 Hours
Logical Agents: Knowledge-based agents; The wumpus world ; Logic; propositional logic: A very simple logic; **First-Order Logic:** Representation revisited; Syntax and semantics of first-order logic; Using first-order logic.	

UNIT – IV	8 Hours
Inference In First-Order Logic: Propositional versus first-order inference ; Unification and lifting; Forward chaining; Backward chaining. **Classical Planning:** Definition of Classical planning; **Quantifying Uncertainty:** Acting under certainty	

UNIT – V	8 Hours
Probabilistic Reasoning: Representing knowledge in an uncertain domain; **Learning:** Forms of learning ; Supervised learning; **AI: The Present and Future:** Agent components; Agent architectures; Are we going in the right direction? What if AI does succeed?	

TEXT BOOK:

1.	Stuart Russel and Peter Norvig : Artificial Intelligence A Modern Approach, 3rd Edition, Pearson India Education limited, 2015, (1.1,2.1to2.4,3.1 to 3.6,5.1 to 5.3, 6.1, 7.1 to 7.4, 8.1 to 8.3, 9.1 to 9.4, 10.1, 13.1, 14.1, 18.1,18.2 and 27th full chapter)

REFERENCE BOOKS:

1.	Elaine Rich, Kevin Knight: Artificial Intelligence, 2nd Edition, Tata McGraw Hill, 1991
2.	Nils J. Nilsson: Principles of Artificial Intelligence, Elsevier, 1980.

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Apply basic principles of AI with problem solving agents.
CO2.	Analyze and apply searching techniques to solve real world problems
CO3.	Represent the real world problems by using first order and propositional logic.
CO4.	Demonstrate the working knowledge of reasoning in the presence of planning, incomplete and /or uncertain information.

Assignment Component: Integrate artificial intelligence into any of the apps.

Self-study component:

1. Integrate artificial intelligence into any of the apps.

Principles of User Interface Design

Contact Hours/Week	:	3+0(L+T)	Credits	:	03
Total Lecture Hours	:	39	CIE Marks	:	50
Total Tutorial Hours	:		SEE Marks	:	50
Course Code	:	4RMCAE7			

Course Objectives:

This Course will enable students to:

1. Explain the concepts of how to use Interactive systems like electronic mails, object action interface model, human-computer interaction
2. Designing the user-interface is intended primarily for designers, managers and evaluators of interactive systems
3. Apply the .concepts of task related menu organization like single menus, combinations of multiple menus, form filling, dialogue boxes, audio menus and window designing.
4. Achieve the goal of universal access.
5. Overviews procedures and guidelines to maintain high degree of interaction required by the system to the user

UNIT – I	7 Hours
Introduction **Usability of Interactive Systems:** Introduction, Usability Requirements, Usability measures, Usability Motivations, Universal Usability, Goals for our profession **Guideline, principles, and Theories**: Introduction, Guidelines, principles, Theories	

UNIT – II	8 Hours
Development Processes Evaluating Interfaces **Managing Design Processes:** Introduction, Organizational Design to support Usability, The Four pillars of design, Development Methodologies, Ethnographic Observation, Participatory Design, Scenario Development, Social Impact statement for Early Design Review, Legal Issues. **Evaluating Interface Designs** Introduction, Expert Reviews, Usability Testing and Laboratories, Survey Instruments, Acceptance Tests, Evaluation during Active Use.	

UNIT – III	8 Hours
Interaction Styles **Direct Manipulation and Virtual Environments:** Introduction, Examples of Direct Manipulation, 3D Interfaces, Teleoperation, Virtual and Augmented Reality. **Menu Selection, Form Fillin, and Dialog Boxes:** Introduction, Task-Related Menu Organization, Single Menus, Combinations of Multiple Menus, Content Organization, Fast Movement Through Menus, Data Entry with Menus: Form Fillin, Dialog Boxes, and Alternatives, Audio Menus and Menus for small Displays.	

UNIT – IV	8 Hours
Command and Natural Languages, Design issues Introduction, Command-Organization functionality Strategies and structure, Naming and Abbreviations, Natural Language in Computing. **Interaction Devices:** Introduction, Keyboards and Keypads, Pointing Devices, Speech and Auditory interfaces, Displays-Small and Large. **Balancing Function and Fashion:** Introduction, Error Messages, Nonanthropomorphic Design, Display Design, Window Design, Color.	

UNIT – V	8 Hours

User Manuals, Online Help, and Tutorials
Introduction, online versus paper documentation, Reading from Paper Verses Displays, Shaping the Content of the Manuals, Online Manuals and Help, Online Tutorials, Demonstrations, and Guides, Online Communities for User Assistance, The Development Process.
Information Search and Visualization:
Introduction, Search in Textual Documents and Database Querying, Multimedia Document Searches, Advanced Filtering and Search Interfaces, Information Visualization

TEXT BOOK:

1.	Ben Shneiderman plaisant Cohen, Jocobs : Designing the User Interface, 5th Edition, Pearson Education, 2010.

REFERENCE BOOKS:

1.	Alan J Dix et. al.: Human-Computer Interaction, II Edition, Prentice-Hall India, 1998.
2.	Eberts: User Interface Design, Prentice-Hall, 1994
3.	Wilber O Galitz: The Essential Guide to User Interface Design - An Introduction to GUI Design, Principles and Techniques, Wiley-Dreamtech India Pvt. Ltd, 1998.

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Elucidate the fundamental concepts of Interactive systems.
CO2.	Design the 3D interfaces using software tools .
CO3.	Implement Development processes based on some strategies .
CO4.	Create and use window designs and menus .
CO5.	Gain knowledge of Use human –computer interfaces like World Wide Web.

Programming with PYTHON Laboratory

Contact Hours/Week	:	2	Credits	:	1.0
Total Lecture Hours	:	-	CIE Marks	:	50
Total Tutorial Hours	:	-	SEE Marks	:	50
Course Code	:	4RMCAL1			

Course Objectives:

This Course will enable students to:	
1.	Understand the fundamentals of programming elements.
2.	Know python data structures like strings, lists, dictionaries, tuples and sets.
3.	Learn to design and develop, classes and objects using object oriented programming
4.	Study how to use and work with files and to use exception handling.
5.	Learn to develop GUIs and to link databases.

Sl.no	Programs
1.	Write python program to a. check the given number is prime or not b. find sum of natural number using recursion
2.	With the help of string array or list, display a simple calendar in python program without using the calendar Unit
3	Write and test a function a. removeDuplicates(somelist) that removes duplicate values from a list b. innerProd(x,y) that computes the inner product of two (same length) lists.
4	Design a user interface in python to function a simple calculator
5.	Show the functionalities of the Boolean operators and Boolean functions in performing a choice of colors via mouse clicks in Rectangles
6.	Demonstrate the different types of plots with necessary features using Matplotlib: Bar chart, Polar plot, pie charts, histograms, contour plot
7.	Demonstrate class and inheritance in python
8.	Demonstrate various image processes using python imaging library c. Convert RGB to grey d. Blur effects e. display only edges
9.	Create a text file using python file I/O. Read the content of the file and change them from lower to upper case characters. Write the updated content in another file and display it.
10.	Write a program to demonstrate the user-defined exception handling mechanism in Python.

Course Outcomes:

After the completion of this course, students will be able to:	
CO1	Implement functions and object oriented concepts with compound data using lists, Tuples and Dictionaries
CO2	Implement file handling and exception handling.
CO3	Implement database and GUI applications

Data Mining Lab

Contact Hours/Week	:	2	Credits	:	1.0
Total Lecture Hours	:	-	CIE Marks	:	50
Total Tutorial Hours	:	-	SEE Marks	:	50
Course Code	:	4RMCAL2			

Course Objectives:

This Course will enable students to:	
1.	Understand R studio to implement preprocessing
2.	Implement data mining techniques using R tool

Sl.no	Programs
1.	Data Exploration and visualization with R
2.	Regression with R
3	Classification with R
4	Data Clustering with R
5.	Association Rule Mining with R

Course Outcomes:

After the completion of this course, students will be able to:	
CO1	Analyse and apply different data preprocessing techniques and data visualization
CO2	Differentiate and apply classifications and regression techniques
CO3	Apply association rule mining algorithms to generate strong association rules and clustering techniques to form data cluster / segments

Advanced Java Lab

Contact Hours/Week	:	4	Credits	:	1.0
Total Lecture Hours	:	-	CIE Marks	:	50
Total Tutorial Hours	:	-	SEE Marks	:	50
Course Code	:	4RMCAL3			

Course Objectives:

This Course will enable students to:	
1.	Master the usage of Java Collection Framework elements and operations.
2.	Implement connectivity of a Database server using JDBC with Java applications.
3.	Master the usage of JDBC API statements to access Database , Metadata and perform queries.
4.	Able to deploy Web Server , Design and develop Enterprise applications using the Servlet and JSP technology
5.	Implement the usage of various web services technologies: WSDL,SOAP

Sl.no	Programs
1.	Java Programs related to Collection Framework elements HashSet, TreeSet
2.	2) Java Programs related to JDBC Statement objects.
3.	3) Design a GUI application using HTML code which accepts inputs for a Student Registration form and sends them to a Servlet which inserts them in to a Database table and display the form details on the web browser.
4.	4a) Write a Java Servlet Program to Auto Web Page refresh in regular intervals of time. (Consider a webpage which is displaying Date and Time.) b) Write a Java Servlet Program to create cookies for first and last names, set expiry date after 24 Hrs for both the cookies and display the cookies data on the web browser.
5.	Write a JAVA Servlet Program to implement HttpSession methods (Using HTTP Session Interface).
6.	JSP Program to demonstrate the usage of JSP Tags and objects.
7.	JSP Program to implement user sessions
8.	Program to implement the service of Web Services Description Language (WSDL)
9.	Program to implement the service of Simple Object Access Protocol (SOAP)

Course Outcomes:

After the completion of this course, students will be able to:	
CO1	Design and develop Java applications to demonstrate the usage of Java Collection framework elements and Java Database transactions over SL queries using Java AWT, SWING, Utilities and JDBC API
CO2	Design and develop dynamic interactive Java applications and client server based web applications to demonstrate database transactions, Java cookies and sessions using JAVA API, embedded SQL code and client side HTML code and using a builder tool Netbeans 7.0
CO3	Design, configure , deploy and demonstrate the implementation of web services using WSDL and SOAP

MCR05: Aptitude Related Analytical Skills

Credits: 1 CIE marks: 00
Course Duration: 36 Hours (6 Days) SEE marks: 00

Sl. No.	Unit covered	Duration (in hrs.)
01.	General Aptitude & Puzzle	10
02.	Verbal Reasoning	03
03.	Logical Reasoning	06
04.	Psychometric Tests	02
05.	Company Specific Special Tests	02
06.	Pre-training Assessments	03
07.	Sample Test	04
08.	Solving Model Test Papers – (Major Corporate Questions)	05
09.	Tips to clear Aptitudes within time limits	01
	Total Course Duration	**36**

Units Snap Shot:

1. General Aptitude:

a) Number based problems
b) Algebra based
c) Time and work related
d) Questions based on train, boat, streams….
e) Interest calculations
f) Clocks and time calculations
g) Area and volume based (measurements)
h) Age calculation based
i) Profit, Loss and other formulae based
j) Percentage, mean, median, mode S.D.

2. Puzzles

3. Verbal Reasoning:

a) Grammar and words usage
b) Sentence completion and comprehension
c) Synonyms and antonyms

4. Logical Reasoning:

a) Odd man series
b) Data interpretation type
c) Problems based on relationship
d) Picture based logical questions
e) Complete the series kind
f) Other logical type

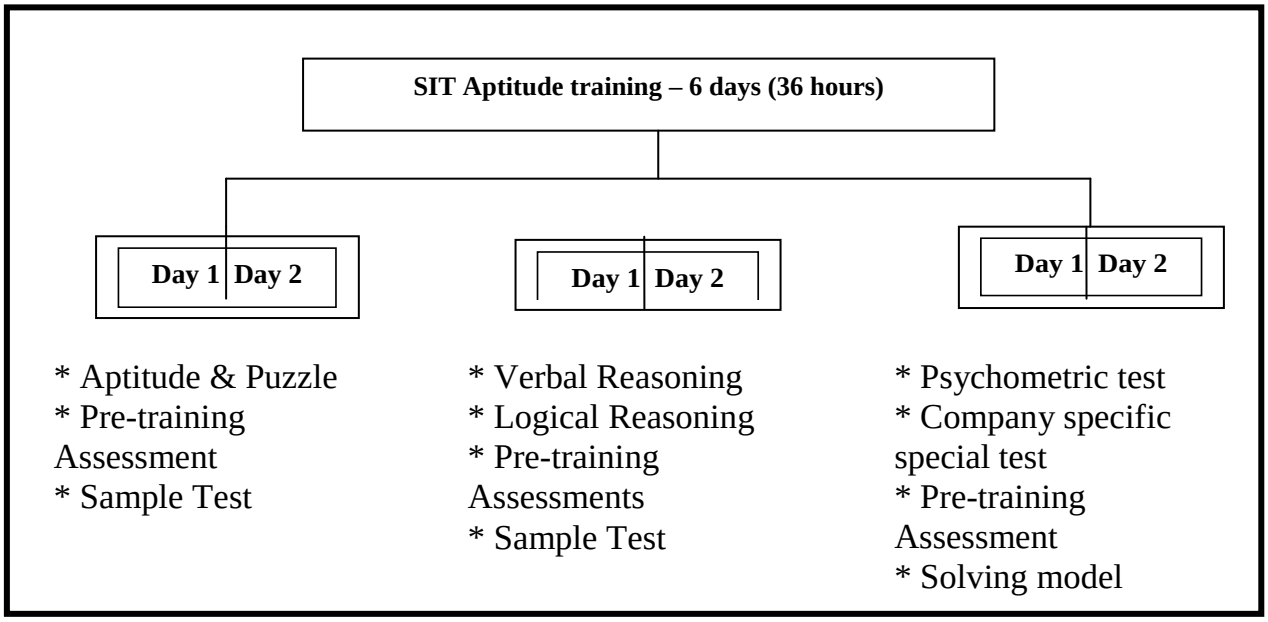
5. Psychometric Test:

a) Intelligence testing
b) Loyalty testing
c) Behavioral testing
d) Stress level test

6. Company Specific Special Test:

{Infosys – Puzzles, TCS – Critical reasoning, Google – Situational tests….
Major fresher recruiting company's special type of question will be discussed}

7. Course Design:



<u>Reference Materials</u>:

1) George Summers - Puzzles
2) R.S. Agarwal - Quantitative Aptitude
3) Baron's GRE 12th Edition
4) Shakuntala Devi – Puzzles
5) Quantitative Aptitude by Alok Chakraborty
6) Quantitative Aptitude by Abhijit Guha

V Semester

Object-Oriented Modeling and Design Patterns

Contact Hours/Week	:	3+2(L+T)	Credits	:	4.0
Total Lecture Hours	:	39	CIE Marks	:	50
Total Tutorial Hours	:	26	SEE Marks	:	50
Course Code	:	5RMCA01			

Course Objectives:

This Course will enable students to:	
1.	Design of simple UML models and structure programs using object-oriented methodologies.
2.	Implementation of object-oriented methods which includes principles abstraction, inheritance and polymorphism in the basic UML diagrams.
3.	Design and development of different models to navigate to the solution of programming problems in UML convention.
4.	Use the standard ethics and effective communication for developing UML models.
5.	Use and implement the gained experience to develop a software using various design patterns on a builder tool.

UNIT – I	8 Hours
OOMD Introduction: What is Object Orientation? What is OO development? Object Oriented themes; Evidence for usefulness of OO Development; Modeling as Design Technique: Modeling; abstraction; The three models. Class Modeling: Object and class concepts, Link and associations concepts, Generalization and inheritance; A sample class model, Practical tips, Advanced Class Modeling: Advanced object and class concepts; Association ends; N-ary associations; Aggregation, Abstract classes; Multiple inheritance; Metadata; Reification Constraints; .	

UNIT – II	7 Hours
State Modeling: Events, States, Transitions and Conditions, State diagrams; State diagram behavior, Practical tips Advanced State Modeling: Nested state diagrams , Nested states, Signal generalization; Concurrency ,A sample state model; Relation of class and state models; Practical tips, Interaction Modeling: Use case models; Sequence models, Activity models, Advanced Interaction Modeling, Use case relationships; Procedural sequence models Special constructs for activity models, summary, Process Overview: Development stages; Development life cycle.	

UNIT – III	8 Hours
Domain Analysis: Overview of analysis, Domain class model; Domain state model Domain interaction model; Iterating the analysis Application Analysis: Application interaction model; Application class model Application state model; Adding operations System Design : Overview of system design; Estimating performance; Making a reuse plan; Breaking a system in to sub-systems Identifying concurrency; Allocation of sub-systems; Management of data storage; Handling global resources, Choosing a software control strategy; Handling boundary conditions Setting the trade-off priorities;	

UNIT – IV	8 Hours
Class Design: Overview of class design; Bridging the gap; Realizing use cases, Designing algorithms; Recurring downwards, Refactoring; Design optimization, Reification of behavior; Adjustment of inheritance, Organizing a class design; ATM example Implementation Modeling: Overview of implementation; Fine-tuning classes, Fine-tuning generalizations; Realizing associations; Testing, Legacy Systems: Reverse engineering; Building the class models; Building the interaction model	
UNIT – V	8 Hours
Design Patterns : What is a pattern and what makes a pattern? Pattern categories; Relationships between patterns , Pattern description; Communication Patterns , Forwarder-Receiver , Design Patterns: Client-Dispatcher-Server, Publisher-Subscriber ,Management Patterns: Command processor, View Handler Communication : Forwader – Receiver, Client-Dispatcher-Server	

TEXT BOOK:

1.	Michael Blaha, James Rumbaugh: Object-Oriented Modeling and Design with UML, 2nd Edition, Pearson Education, 2005.(Chapters 1 to 17, 23)
2.	Frank Buschmann, Regine Meunier, Michael Stal, Pattern-Oriented Software Architecture, A System of Patterns, Volume 1, John Wiley and Sons, 2006. (Chapters 1, 3.5, 3.6, 4)

REFERENCE BOOKS:

1.	Grady Booch Object-Oriented Analysis and Design with Applications, 3rd Edition, Pearson, 2007.
2.	Mark Priestley Practical Object-Oriented Design with UML, 2nd Edition, Tata McGraw-Hill, 2003.
3.	Booch, and Jacobson, The Unified Modeling Language User Guide, 2nd Edition, Pearson, 2005

WEB LINKS:

1.	https://www.javatpoint.com/uml
2.	https://www.visual-paradigm.com
3.	https://www.geeksforgeeks.org/unified-modeling-language-uml-state-diagrams/
4.	https://www.tutorialspoint.com/uml/uml_interaction_diagram.htm
5.	https://sourcemaking.com/design_patterns

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Design simple UML models and structure programs using object-oriented methodologies.
CO2.	Implement object-oriented methods which includes principles abstraction, inheritance and polymorphism in the basic UML diagrams.
CO3.	Design and develop different models to navigate to the solution of programming problems in UML convention.
CO4.	Apply standard ethics and effective communication for developing UML models.
CO5.	Design and develop real time interactive systems in UML convention using various design patterns on a builder tool.

Mobile Application Development

Contact Hours/Week	:	3+2(L+T)	Credits	:	4.0
Total Lecture Hours	:	39	CIE Marks	:	50
Total Tutorial Hours	:	26	SEE Marks	:	50
Course Code	:	5RMCA02			

Course Objectives:

This Course will enable students to:	
1.	Learn to setup Android application development environment
2.	Illustrate user interfaces for interacting with apps and triggering actions
3.	Interpret tasks used in handling multiple activities.
4.	Identify options to save persistent application data
5.	Appraise the role of security and performance in Android applications

UNIT – I	8 Hours

Introduction

Preliminary Considerations – Cost of Development – Importance of Mobile Strategies in the Business World – Effective use of Screen Real Estate

Understanding Mobile Applications

Understanding Mobile Applications Users – Understanding Mobile Information Design – Understanding Mobile Platforms – Using the Tools of Mobile Interface Design.

UNIT – II	8 Hours

Getting Started with Android Programming

What is Android – Obtaining the required tools– Anatomy of an Android Application – Components of Android Applications – Activities – Fragments – Utilizing the Action Bar

UNIT – III	8 Hours

Android UI Design and Location Based Services

Views and View Groups – Basic Views – Fragments – Displaying Maps – Getting Location Data – Publishing for Publishing – Deploying APK Files

UNIT – IV	8 Hours

Android Messaging and Networking

SMS Messaging – Sending Email – Networking – Downloading Binary Data, Text files – Accessing Web Services – Performing Asynchronous Call – Creating your own services – Communicating between a service and an activity – Binding activities to services

UNIT – V	8 Hours

Feedback and Oscillator Circuits

iOS – Obtaining the tools and SDK – Components of XCODE – Architecture of iOS Building Derby App in iOS – Other useful iOS things – Windows Phone: Getting the tools you need – Windows Phone 7 Project – Building Derby App in Windows Phone 7 – Distribution – Other useful Windows Phone Thing

TEXT BOOK:

1.	Jeff McWherter and Scott Gowell, "Professional Mobile Application Development", 1st Edition, 2012, *ISBN*: 978-1-118-20390-3

2.	Wei-Meng Lee, "Beginning Android Application Development", Wiley 2011

REFERENCE BOOKS:

1.	Reto Meier "Professional Android 4 Application Development", Wrox Publications2012

WEB LINKS:

1.	https://www.youtube.com/watch?v=giVfVQlKBVM
2.	https://www.youtube.com/watch?v=aS__9RbCyHg
3.	https://www.tutorialspoint.com/android/index.htm
4.	https://developer.android.com/codelabs/build-your-first-android-app#0
5.	https://www.geeksforgeeks.org/android-tutorial/
6.	https://www.tutlane.com/tutorial/android
7.	https://www.w3schools.in/category/android-tutorial/

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Illustrate effective user interfaces that leverage evolving mobile device capabilities
CO2.	Develop applications using software development kits (SDKs), frameworks and toolkits.
CO3.	Establish various views methods involved to design app, API in getting map, its location and presenting the apps to the users.
CO4.	Design and develop an application with synchronizing the real time applications like SMS, Email, etc and services
CO5.	Build and deploy competent mobile application development solutions on various cross platforms.

Cloud Computing

Contact Hours/Week	:	3+0(L+T)		Credits	:	3.0
Total Lecture Hours	:	39		CIE Marks	:	50
Total Tutorial Hours	:	-		SEE Marks	:	50
Course Code	:	5RMCAE12				

Course Objectives:

This Course will enable students to:	
1.	Examine the use of Cloud Computing to solve given engineering problem
2.	Use of the public cloud computing platforms
3.	Illustrate through the technical talk and project demonstrations

UNIT – I	8 Hours
Defining Cloud Computing: Cloud Types, The NIST model, The Cloud Cube Model, Deployment models, Service models, Examining the Characteristics of Cloud Computing, Paradigm shift, Benefits of cloud computing, Disadvantages of cloud computing; Assessing the value proposition: Early adopters and new applications, the laws of cloudonomics, cloud computing obstacles, behavioral factors relating to cloud adoption, measuring cloud computing costs, specifying SLAs with examples.	

UNIT – II	8 Hours
Understanding Cloud Architecture: Exploring the Cloud Computing Stack, Composability, Infrastructure, Platforms, Virtual Appliances, Communication Protocols; Understanding Services and Applications by Type: Defining IaaS, Defining PaaS, Defining SaaS, Defining IDaaS.	

UNIT – III	8 Hours
Understanding Abstraction and Virtualization: Using Virtualization Technologies, Load balancing and Virtualization, Understanding Hypervisors; Capacity Planning: Defining Baseline and Metrics, Baseline measurements, System metrics, Load testing, Resource ceilings, Server and instance types, Network Capacity, Scaling	

UNIT – IV	8 Hours
Using Platforms: Defining Services, Salesforce.com versus Force.com: SaaS versus PaaS, Application development , Exploring Google Applications, Surveying the Google Application Portfolio, Understanding Amazon Web Services, Amazon Web Service Components and Services, Exploring Microsoft Cloud Services, Defining the Windows Azure Platform	

UNIT – V 7 Hours

Moving Applications to the Cloud : Applications in the Clouds , Functionality mapping, Application attributes, Cloud service attributes, System abstraction, Cloud bursting, Applications and Cloud APIs

TEXT BOOK:

1.	Barrie Sosinsky, "Cloud Computing Bible", Wiley Publishing Inc. 2011 (free e-book available).

REFERENCE BOOKS:

1.	David S. Linthicum, Cloud Computing and SOA Convergence in Your Enterprise: A Step-by-Step Guide (free e-book available)

2.	Kai Hwang, Geoffrey C. Fox, and Jack J. Dongarra, "Distributed and Cloud Computing – From Parallel Processing to the Internet of Things", Morgan Kaufman Publishers, 2012.
3.	GautamShroff, Enterprise Cloud Computing Technology Architecture Applications (free e-book available)
4.	Toby Velte, Anthony Velte, Robert Elsenpeter, Cloud Computing, A Practical Approach (free e-book available)

WEB LINKS:

1.	Barrie Sosinsky, "Cloud Computing Bible", Wiley Publishing Inc. 2011 https://arpitapatel.files.wordpress.com/2014/10/cloud-computing-bible1.pdf
2.	https://docs.microsoft.com/en-gb/learn/modules/intro-to-azure-fundamentals/what-is-cloud-computing
3.	Getting start with GCP, Azure, AWS https://cloud.google.com/compute/docs, https://docs.microsoft.com/en-gb/learn/azure, https://aws.amazon.com/getting-started/hands-on/
4.	Getting start with Docker & Kubernates https://docs.docker.com/get-started/ https://kubernetes.io/docs/tutorials/

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Analyse the trade-offs in cloud modules, services and cloud characteristics.
CO2.	Explore the cloud architecture, protocols and the services.
CO3.	Conceptualise the virtualization and its role in enabling cloud computing applications.
CO4.	Deploy applications over commercial cloud computing infrastructures such as Amazon Web Services, Windows Azure, and Google AppEngine.

Assignment (self-study) Component:

Demonstration of various cloud services to host static and dynamic web pages will be considered to the award of assignment marks.

Machine Learning

Contact Hours/Week	:	3+0(L+T)		Credits	:	3.0
Total Lecture Hours	:	39		CIE Marks	:	50
Total Tutorial Hours	:	--		SEE Marks	:	50
Course Code	:	5RMCAE13				

Course Objectives:

This Course will enable students to:	
1.	Learn the basic concepts of machine learning and Concept Learning.
2.	Learn the elements of decision trees & Model Evaluation.
3.	Understand the concept of artificial neural networks and instance based learning.
4.	Understand the Bayesian classifiers.
5.	Familiarize the concept of Evaluating Hypothesis, unsupervised learning.

UNIT – I	7 Hours
Introduction: Well posed learning problems, Designing a Learning system, Perspective and Issues in Machine Learning. **Concept Learning:** A Concept learning task, Concept learning as search, Find-S: Finding a maximally specific hypothesis, Version spaces and the Candidate Elimination algorithm, Inductive Bias.	

UNIT – II	8 Hours
Decision Tree Learning: Decision tree representation, Appropriate problems for decision tree learning, The Basic decision tree learning algorithm, Hypothesis space search in decision tree learning, Inductive bias in decision tree learning, Issues in decision tree learning. **Model Evaluation:** Metrics for evaluating Classifier Performance, Holdout method & Random Subsampling, Cross-Validation, Bootstrap.	

UNIT – III	8 Hours
Instance based Learning: Introduction, k-nearest neighbor learning, case-based reasoning. **Artificial Neural Networks:** Introduction, Neural Network representations, Appropriate problems for Neural network Learning, Perceptrons, Multilayer Networks and the backpropogation algorithm.	

UNIT – IV	8 Hours
Bayesian Learning: Introduction, Bayes theorem, Bayes theorem and concept learning, ML and LS error hypotheses, ML hypotheses for predicting probabilities, MDL principle, Naïve Bayes classier, Bayesian belief networks, The EM algorithm.	

UNIT – V	8 Hours
Evaluating Hypothesis: Motivation, Estimating hypothesis accuracy, Basics of sampling theory, A General approach for deriving confidence intervals, Difference in error of two hypotheses, Comparing learning algorithms. **Unsupervised Learning:** Cluster analysis: What is Cluster analysis?, Requirements for Cluster analysis, Partitioning Methods: k-Means, Hierarchical Methods: Agglomerative vs Divisive Hierarchical Clustering, Distance Measures in Algorithmic Methods	

TEXT BOOK:

1.	Tom M Mitchell , Machine Learning, McGraw Hill Education (India) Edition 2013. (Sections: 1.1 to 1.3, 2.2 to 2.5, 2.7, 3.2 to 3.7, 4.1 to 4.5, 6.1 to 6.6, 6.9,6.11,6.12, 5.1 to

	5.6, 8.1 to 8.5)
2.	Jiawei Han, MK, JP, Data mining, Concepts & Techniques, 3rd Edition, MK. (Sections: 8.5, 8.5.1,8.5.2,8.5.3,8.5.4,10.1, 10.1.1, 10.1.2, 10.2, 10.2.1, 10.3, 10.3.1, 10.3.2)

REFERENCE BOOKS:

1.	Ethem Alpaydin, Introduction to Machine Learning, 3rd Edition, EEE, MIT Press, 2018.
2.	Trevor Hastie, Robert Tibshiran, J. H. Friedman, The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition (Springer Series in Statistics), 19 April 2017.
3.	V.Susheela Devi, M.Narasimha Murthy, Pattern Recognition: An Introduction, Universities Press

WEB LINKS:

1.	https://www.kaggle.com/
2.	https://machinelearningmastery.com/
3.	https://www.tensorflow.org/
4.	https://www.youtube.com/?gl=IN&tab=w1
5.	https://ai.google/
6.	http://ndl.iitkgp.ac.in/

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Analyse the characteristics of algorithms related to concept learning and to use it in solving related problems.
CO2.	Perform classification/labelling of data objects using concepts of decision tree learning and investigate instant based learning.
CO3.	Implement the concept of neural networks for learning linear and non-linear functions.
CO4.	Apply Bayesian techniques and derive effectively learning rules for real world problems.
CO5.	Evaluate hypothesis and investigate unsupervised Learning to solve the given problems

Assignment Component:

Writing and executing Programs on Machine learning algorithms using Python . The assignment is valued for 4 marks.

Data Science

Contact Hours/Week	:	3+0(L+T)		Credits	:	3.0
Total Lecture Hours	:	39		CIE Marks	:	50
Total Tutorial Hours	:	-		SEE Marks	:	50
Course Code	:	5RMCAE15				

Course Objectives:

This Course will enable students to:	
1.	Understand fundamentals of Statistical modelling , probability distributions
2.	Understand data visualization and feature selection method
3.	Understand basic machine learning algorithms
4.	Understand Mining Social-Network Graphs
5.	Understand Ethical Issues, discussions on privacy, security, ethics

UNIT – I	7 Hours
Introduction: What is Data Science? Big Data and Data Science hype – and getting past the hype, why now? – Datafication, Current landscape of perspectives, Skill sets. Needed Statistical Inference: Populations and samples, Statistical modelling, probability distributions, fitting a model, Introduction to R	

UNIT – II	8 Hours
Exploratory Data Analysis and the Data Science Process: Basic tools (plots, graphs and summary statistics) of EDA, Philosophy of EDA, Data Visualization: Basic principles, ideas and tools for data visualization. The Data Science Process, Case Study: RealDirect (online real estate firm). Three Basic Machine Learning Algorithms: Linear Regression, k-Nearest Neighbours (kNN), k-means	

UNIT – III	7 Hours
Naïve Bayes classifier, Filtering Spam using Naïve Bayes, Why Linear Regression and k-NN are poor choices for Filtering Spam, Data Wrangling: APIs and other tools for scrapping the Web	

UNIT – IV	9 Hours
Feature Generation and Feature Selection (Extracting Meaning from Data): Motivating application: user (customer) retention. Feature Generation (brainstorming, role of domain expertise, and place for imagination), Feature Selection algorithms. Filters; Wrappers; Decision Trees; Random Forests. Recommendation Systems: Building a User-Facing Data Product, Algorithmic ingredients of a Recommendation Engine, Dimensionality Reduction, Singular Value Decomposition, Principal Component Analysis, Exercise: build your own recommendation system	

UNIT – V	8 Hours
Mining Social-Network Graphs: Social networks as graphs, Clustering of graphs, Direct discovery of communities in graphs, Partitioning of graphs, Neighbourhood properties in graphs, Data Science and Ethical Issues, Discussions on privacy, security, ethics, Next-generation data scientists	

TEXT BOOK:

1.	Doing Data Science, Cathy O'Neil and Rachel Schutt, Straight Talk from The Frontlin, O'Reilly publications , 2014
2.	Mining of Massive DatasetsV2.1, Jure Leskovek, AnandRajaraman and Jeffrey Ullman, Cambridge

	University Press, 2nd Edition, 2014

REFERENCE BOOKS:

1.	Data Mining: Concepts and Techniques, Jiawei Han, MichelineKamber and Jian Pei, Morgan Kauffman, Third Edition, 2012.

WEB LINKS:

1.	https://www.youtube.com/watch?v=iROHLA_TXQM
2.	https://www.javatpoint.com/r-tutorial
3.	https://www.javatpoint.com/spss-variables
4.	https://www.javatpoint.com/data-science
5.	https://towardsdatascience.com/feature-extraction-techniques-d619b56e31be

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Apply statistical method , probability and visualization methods for data
CO2.	Apply feature selection method , Data Wrangling and Web scrapping
CO3.	Apply basic machine learning algorithm: classification and clustering for various application

Software Testing

Contact Hours/Week	:	3+0(L+T)		Credits	:	3.0
Total Lecture Hours	:	39		CIE Marks	:	50
Total Tutorial Hours	:	-		SEE Marks	:	50
Course Code	:	5RMCAE21				

Course Objectives:

This Course will enable students to:	
1.	Understand the principles of testing.
2.	Identify the need for testing.
3.	Identify testing process.
4.	Describe various types of test and their objectives.
5.	Identify the types of tests to be conducted on applications.

UNIT – I	6 Hours
Introduction to testing as an engineering activity, The evolving profession of software engineering, the role of process in software quality, testing as a process. Testing fundamentals: Initiating a study of testing, basic definitions, why testing is necessary? What is testing? The psychology of testing? Software quality assurance group, software-testing principles, The Tester's Role in a Software Development Organization.	

UNIT – II	10 Hours
The test development life cycle (TDLC), when should testing stop?; Verification strategies; review, walkthrough, inspection Testing types and techniques; white box testing: basis path testing, flow graph notation, cyclomatic complexity, graph matrices, loop testing Black box testing: boundary value analysis, equivalence partitioning, graph based testing methods, error guessing.	

UNIT – III	10 Hours
Levels of testing; The Need for Levels of Testing, Unit Test, Unit Test Planning, Designing the Unit Tests. The Class as a Testable Unit, Running the Unit tests and Recording results, Integration tests, Designing Integration Tests, Integration Test Planning, System Test – The Different Types, Regression Testing, Alpha, Beta and Acceptance Tests, Web testing: Introduction to web testing, web testing process and techniques, cross browser testing, web browser error messages, Performance testing	

UNIT – IV	7 Hours
Test planning: what is test plan; why to plan test? Template for test plan; guidelines for creating the test plan; risk analysis Test design: importance; test design essentials; good test case; test case mistakes; test case template; test design stages.	

UNIT – V	6 Hours
Test execution: Objectives; execution considerations; execution activities Defect management: what is defect; defect life cycle; defect management process.	

TEXT BOOK:

1.	Ilene Burnstein, "Practical Software Testing", Springer international edition. (Unit-I)
2.	Foundations of Software Testing ISTQB certification (Level I) by Dorothy graham, Erik van veenendaal, Rex black.

REFERENCE BOOKS:

1.	Boris Beizer, "Software System Testing And Quality Assurance", vannostrand reinhold ,newyork
2.	Gordon schulmeyer , "Zero Defect Software" , McGraw –hill book co
3.	Watts Humphrey, "Managing the Software Process", Addison weselypub.co.inc.

WEB LINKS:

1.	https://www.coursera.org/specializations/software-testing-automation
2.	https://www.udemy.com/course/everything-for-software-tester/
3.	https://www.udacity.com/course/software-testing--cs258
4.	https://www.greatlearning.in/academy/learn-for-free/courses/software-testing-fundamentals1
5.	https://www.guru99.com/software-testing.html
6.	https://testinginstitute.com/Free-Software-Testing-Training.php
7.	https://onlinecourses.nptel.ac.in/noc19_cs71/preview

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Identify the need and the importance of software testing as an engineering activity
CO2.	Interpret the concept of testing using different types of testing and testing techniques.
CO3.	Identify the different levels of testing.
CO4.	Exemplify how to write the test plan and test cases.
CO5.	Interpret the defect management process

Software Agents

Contact Hours/Week	:	3+0(L+T)		Credits	:	3.0
Total Lecture Hours	:	39		CIE Marks	:	50
Total Tutorial Hours	:	-		SEE Marks	:	50
Course Code	:	5RMCAE22				

Course Objectives:

This Course will enable students to:	
1.	Understand agents and manipulation of agents
2.	Deal with agents for communication and coordination
3.	Understand agent oriented programming and language
4.	Understand agent architecture
5.	Explore mobile agent concepts

UNIT – I	8 Hours
Agent and User Experience: Interacting agents, agents from direct manipulation to deligation-interface agent metaphor with character designing agents- direct manipulation versus agent path to predictable	

UNIT – II	8 Hours
Agents for learning in intelligent assistance : Agents for information sharing and coordination-agents that reduce work information overhead-agents without programming language-life like computer character-S/W agents for cooperative learning-architecture of intelligent agents.	

UNIT – III	7 Hours
Agent Communication and collaboration: **Overview of agent oriented programming-agent communication language-agent based framework of interoperability.**	

UNIT – IV	9 Hours
Agent Architecture : Agents for information gathering-open agent architecture-communicative action for artificial agent	

UNIT – V	7 Hours
Mobile Agents : Mobile agent paradigm-mobile agent concepts-mobile agent technology-case study: Tele script, Agent Tel	

TEXT BOOK:

1.	Jeffrey M.Bradshaw, Software agents, MIT Press, 2000 (Units 1,2,3 & 4)
2.	William R.Cockayne, Michael Zyda, Mobile agents, Prentice Hall, 1998 (Unit 5).

REFERENCE BOOKS:

1.	Russel and Norvig, Artificial Intelligence: A Modern Approach, prentice Hall, 2nd Edition, 2002.

WEB LINKS:

1.	http://agents.umbc.edu/introduction/ao/
2.	http://jade.tilab.com/

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Elucidate concepts related to software agents
CO2.	Elucidate Concepts of agent learning and to design a software agent for a given problem
CO3.	Design mobile software agents that can tackle the problems of societal interest

Software Quality Management

Contact Hours/Week	:	3+0(L+T)		Credits	:	3.0
Total Lecture Hours	:	39		CIE Marks	:	50
Total Tutorial Hours	:	-		SEE Marks	:	50
Course Code	:	5RMCAE23				

Course Objectives:

This Course will enable students to:	
1.	Understand various software models dealing with software quality
2.	Explain the details of quality tasks
3.	Understand the quality control and reliability aspects of software management
4.	Understand quality management system and standards

UNIT – I	9 Hours
Introduction To Software Quality : Software Quality – Hierarchical models of Boehm and McCall – Quality measurement – Metrics measurement and analysis – Gilb's approach – GQM Model	

UNIT – II	8 Hours
Software Quality Assurance : Quality tasks – SQA plan – Teams – Characteristics – Implementation – Documentation – Reviews and Audits	

UNIT – III	8 Hours
Quality Control And Reliability: Tools for Quality – Ishikawa's basic tools – CASE tools – Defect prevention and removal – Reliability models – Rayleigh model – Reliability growth models for quality assessment	

UNIT – IV	8 Hours
Quality Management System : Elements of QMS – Rayleigh model framework – Reliability Growth models for QMS – Complexity metrics and models – Customer satisfaction analysis.	

UNIT – V	6 Hours
Quality Standards: Need for standards – ISO 9000 Series – ISO 9000-3 for software development – CMM and CMMI – Six Sigma concepts	

TEXT BOOK:

1.	Allan C. Gillies, "Software Quality: Theory and Management", Thomson Learning, 2003. (UI : Ch 1-4 ; UV : Ch 7-8)
2.	Stephen H. Kan, "Metrics and Models in Software Quality Engineering", Pearson Education (Singapore) Pte Ltd., 2002. (UI : Ch 3-4; UIII : Ch 5-8 ; UIV : Ch 9-11)

REFERENCE BOOKS:

1.	Norman E. Fenton and Shari Lawrence Pfleeger, "Software Metrics" Thomson, 2003
2.	Mordechai Ben – Menachem and Garry S.Marliss, "Software Quality", Thomson Asia Pte Ltd, 2003.
3.	Mary Beth Chrissis, Mike Konrad and Sandy Shrum, "CMMI", Pearson Education (Singapore) Pte Ltd, 2003.
4.	ISO 9000-3 "Notes for the application of the ISO 9001 Standard to software development.

WEB LINKS:

1.	https://www.tutorialspoint.com/software_testing_dictionary/quality_management.htm
2.	https://www.tutorialspoint.com/software_quality_management/index.htm
3.	http://moodle.autolab.uni-pannon.hu/Mecha_tananyag/szoftverfejlesztesi_folyamatok_angol/ch12.html
4.	https://www.geeksforgeeks.org/software-engineering-software-quality-assurance/

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Identify the quality assurance context and the hierarchical models of quality.
CO2.	Comprehend the SQA plan and its implementation.
CO3.	Demonstrate the different quality and reliability models for assessment.
CO4.	Comprehend the quality control and reliability growth models.
CO5.	Identify the need for quality standards and understand the different quality standards

Big Data Analysis

Contact Hours/Week	:	3+0(L+T)		Credits	:	3.0
Total Lecture Hours	:	39		CIE Marks	:	50
Total Tutorial Hours	:	-		SEE Marks	:	50
Course Code	:	5RMCAE31				

Course Objectives:

This Course will enable students to:	
1.	Understand basics of Big Data, its elements, its analytics , its usage in business context , computing in big data etc
2.	Understand modern tools of Hadoop Ecosystem to the solution of various problems in storage , processing , accessing , managing and analyzing the big data
3.	Understand the different layers of Big Data Stack architecture in effective analysis of big data and analyze the merits of using modern Data Warehouses against the limitations of traditional Databases
4.	Understand Map Reduce programs and requirement of Yarn Architecture.
5.	Understand the requirements of NO SQL and need of Hive tools.

UNIT – I	7 Hours

Getting an Overview of Big Data: What is Big Data? History of Data, Management – Evolution of Big Data, Structuring Big Data, Types of Data, Elements of Big Data, Volume, Velocity, Variety, Veracity, Big Data Analytics, Careers in Big data, Advantages of Big Data Analytics, Future of Big Data.

Exploring the Use of Big Data in Business Context: Use of Big Data in Social Networking, Business Intelligence, Marketing, Product Design and Development, Use of Big Data in Preventing Fraudulent Activities, Preventing Fraud Using Big Data Analytics, Use of Big Data in Retail Industry, Use of RFID Data in Retail.

UNIT – II	8 Hours

Introducing Technologies for Handling Big Data: Distributed and Parallel Computing for Big Data,How data models and computing models are different, Introducing Hadoop, HDFS and MapReduce, How does Hadoop Function? Cloud Computing and Big Data, Cloud Services for Big Data, In-Memory Computing Technology for Big Data

Understanding Hadoop Ecosystem: Hadoop Ecosystem**,** Hadoop Distributed File System, HDFS Architecture, Concepts of Blocks in HDFS Architecture, Name Nodes and Data Nodes, Features of HDFS, MapReduce, Hadoop Yarn, Introducing HBase, HBase Architecture, Regions, Storing Bigdata with HBase, Interacting with the Hadoop Ecosystem, HBase in Operation –Programming with HBase, Combining HBase and HDFS, REST and Thrift, Data Integrity in HDFS, Features of HBase ,hive, Pig and Pig Latin, Sqoop, Zookeeper, Flume, Oozie

UNIT – III	8 Hours

Understanding Big Data Technology Foundations:

Exploring the Big Data Stack, Data Sources Layer, Ingestion Layer, Storage Layer, Physical Infrastructure Layer, Platform Management Layer, Security Layer, Monitoring Layer, Analytics Engine, Visualization Layer, Big Data Applications, Virtualization and Big Data, Virtualization Approaches, Server virtualization, Application Virtualization, Network Virtualization, Processor and Memory Virtualization, Data and Storage Virtualization, Managing Virtualization with Hypervisor.

Storing Data in Databases and Data Warehouses: RDBMS and Big Data, CAP Theorem,

Issues with the Relational Model, Non-Relational Database, Issues with the Non-Relational Model, Polyglot Persistence, Integrating Big Data with Traditional Data Warehouses, Big Data Analysis and Data Warehouse, Changing Deployment Models in Big Data Era

UNIT – IV	8 Hours
Understanding MapReduce Fundamentals and HBase: The MapReduce Framework. Exploring the Features of MapReduce. Working of MapReduce. Exploring Map and Reduce Functions. Techniques to Optimize MapReduce Jobs. Hardware/Network Topology, Synchronization, File System. Uses of MapReduce. Role of HBase in Big Data Processing. Characteristics of HBase. **Understand Hadoop YARN Architecture**: Limitations of MapReduce, Advantages of YARN, YARN architecture : Resource manager, application manager , Integration of ResourceManager and Application Manager. Working of YARN. YARN schedulers: Capacity and Fail Schedulter. Backward compatibility with YARN	

UNIT – V	8 Hours
Exploring Hive: Introducing Hive, Getting Started with Hive,Hive services, Hive Variables, Hive Properties, Hive Queries, Data Types in Hive, Built-In Functions in Hive, Hive DDL, Creating Databases, Viewing a Database, Dropping a Database, Altering Databases, Creating Tables, Creating a Table Using the Existing Schema, Dropping Tables, Altering Tables, Using Hive DDL Statements, Data Manipulation in Hive, Loading Files into Tables, Inserting Data into Tables, Update in Hive, Delete in Hive, Using Hive DML Statements, Data Retrieval Queries, Using the SELECT Command, Using the WHERE Clause, Using the GROUP BY Clause, Using the HAVING Clause, Using the LIMIT Clause, Executing HiveQL Queries, Using JOINS in Hive, Inner Joins, Outer Joins, Cartesian Product Joins, Map-Side Joins, Joining Tables **NoSQL**: Introduction to NoSQL, why NoSQL, Characteristics of NoSQL. Types of NoSQL models: key value Data model, Column-oriented data model, document data model, graph databases. Schema less databbse, materialized views, Distributed models: CAP theorem. Sharding	

TEXT BOOK:

1.	BIG DATA Black Book ,D T Editorial Services, Dreamtech press 2016 Edition

REFERENCE BOOKS:

1.	Big Data, Anil Maheswari, Mc Graw Hill
2.	NoSQL For Mere Mortals, Dan Sullivan, Addison Wisley Pearson

WEB LINKS:

1.	https://nptel.ac.in/courses/106/104/106104189/
2.	https://www.youtube.com/watch?v=1vbXmCrkT3Y
3.	https://www.guru99.com/nosql-tutorial.html
4.	https://www.youtube.com/watch?v=2yQ9TGFpDuM
5.	https://www.w3schools.in/mongodb/introduction-to-nosql/
6.	https://www.tutorialspoint.com/big_data_tutorials.htm

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Comprehend big data applications of big data, elements of big data, and data types of big data
CO2.	Categorize the components of Hadoop Ecosystem and technologies for handling big Data.
CO3.	Differentiate layers of Big Data Stack architecture for effective analysis of big data
CO4.	Design and develop Map Reduce programs for Real world applications; identify the advantages of Yarn Architecture.
CO5.	Identify the need and application of Hive and No SQL

System Simulation and Modeling

Contact Hours/Week	:	3+0(L+T)		Credits	:	3.0
Total Lecture Hours	:	39		CIE Marks	:	50
Total Tutorial Hours	:	-		SEE Marks	:	50
Course Code	:	5RMCAE32				

Course Objectives:

This Course will enable students to:	
1.	Judge whether simulation can be a useful both an analysis tool for predicting the effects of changes to existing systems.
2.	Identify the major capabilities and limitations of discrete-event simulation for modeling types of systems
3.	Define a simulation model for a practical business scenario.
4.	Describe the generation of random number, input modeling, output analysis and comparison of alternative system
5.	Determine the appropriate simulation model to solve any real-world problems.

UNIT – I	7 Hours
When Simulation is the appropriate and not appropriate tool, 4 phases and 12-Steps in a simulation study. Advantages & Disadvantages of Simulation. Areas of application of simulation, Systems and System Environment, Components of a System, Discrete and Continuous Systems, Model of a System, Types of Models, Discrete – Event System Simulation, Steps in a Simulation Study, Queuing Models: Characteristics of Queuing systems, Queuing notations; Examples.	

UNIT – II	8 Hours
Simulation Examples, Simulation of Queuing Systems, Simulation of Inventory Systems, The Event – Scheduling / Time – Advance Algorithm, World Views , Manual simulation using Event Scheduling, List processing,	

UNIT – III	8 Hours
Properties of Random numbers, Generation of Pseudo-random Numbers, Techniques for generating random Numbers, Tests for random Numbers Frequency Test , Runs Test, Tests for Autocorrelation, Gap Test , Poker Test, Random Variate Generation: Inverse Transform Technique, Exponential Distribution, Uniform Distribution, Acceptance-Rejection technique, Poisson Distribution	

UNIT – IV	9 Hours
Data Collection, Identifying the distribution with data Histograms, Parameter Estimation Preliminary Statistics, Examples: sample Mean and Sample Variance, Goodness of Fit test : Chi-Square Test, Kolmogorov-Smirnov Goodness-of-Fit Test : Examples, Selecting Input Models without data, Multivariate and Time-Series Input Models	

UNIT – V	7 Hours
Introduction Model Building, Verification and validation, Verification of simulation Models, Calibration and validation of models, Face validity and validation of model assumptions, Input – output validation: using historical input data and turning test, Output Analysis for a single model : Types of Simulations with respect to Output Analysis, Stochastic Nature of Output data, measure of performance and Estimation, Examples for performance estimation, Output analysis for Terminating Simulation, Output analysis for Steady-state simulations	

TEXT BOOK:

1.	Jerry Banks , John S Carson , Barry L Nelson , David M Nicol " Discreate-Event System Simulation", 4th Edition, Prentice-Hall India

REFERENCE BOOKS:

1.	Lawrence M. Leemis , Stephen K. park : Dscreate – Event Simulation : A First Cource , Pearson Prentice-Hall, 2006
2.	Averill M Law, W David Kelton , "Simulation Modeling and Analysis", 4th Edition, McGraw - Hill. 2007

WEB LINKS:

1.	https://www.youtube.com/playlist?list=PLPTjP-gx7TM8n9SUdp0tt1m788LfOpo9u
2.	https://www.youtube.com/watch?v=Wp3jyLkfBQs
3.	https://nptel.ac.in/courses/112/107/112107220/

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Elucidate the basic concepts in modelling and simulation.
CO2.	Classify various simulation models and give practical examples for each category
CO3.	Generate random numbers and variants to apply them to develop simulation Models
CO4.	Analyse the effects of randomness on system behaviour and performance
CO5.	Gain knowledge of input modelling, verification, validation, calibration and Performance estimation of existing simulation models

Digital Image Processing

Contact Hours/Week	:	3+0(L+T)		Credits	:	3.0
Total Lecture Hours	:	39		CIE Marks	:	50
Total Tutorial Hours	:	-		SEE Marks	:	50
Course Code	:	5RMCAE34				

Course Objectives:

This Course will enable students to:	
1.	Understand fundamentals of digital image processing
2.	Taste of the applications of the theories of image processing algorithms
3.	Useful skill base that would allow them to carry out further study should they be interested and to work in the field

UNIT – I	8 Hours
Introduction, digital image processing, fundamental steps in digital image processing, components of digital image processing system. Elements of visual perception, image sensing and acquisition, image sampling and quantization, some basic relationships between pixels. Intensity transformation and spatial filtering, Background, some basic intensity transformation functions.	

UNIT – II	7 Hours
Histogram processing, fundamentals of spatial filtering, smoothing spatial filters sharpening of spatial filters.	

UNIT – III	8 Hours
Image restoration and reconstruction, noise models, restoration in the presence of noise only spatial filtering, estimating the degradation, inverse filtering.	

UNIT – IV	8 Hours
Color Image Processing, color fundamentals, color models, pseudo color image processing, basics of full color image processing color transformations, formulation and color complements, smoothing and sharpening. Image compression, fundamentals, coding redundancy, some basic compression methods.	

UNIT – V	8 Hours
Morphological Image processing, preliminaries, erosion and dilation, opening and closing, hit-or-miss transformation. Image Segmentation, fundamentals, point, line and edge detection	

TEXT BOOK:

1.	Rafael C. Gonzaliz and Richard E.Woods, Digital Image Processing, III edition, Pearson education, 2008. 1.1, 1.4, 1.5, 2.1,2.3,2.4, 2.5, 3.1-3.6, 4.1-4.3, 5.1-5.3, 5.6.1,5.7, 6.1-6.4, 6.5.1, 6.5.2, 6.6, 8.1-8.2.4, 9.1-9.4, 10.1-10.2.3

REFERENCE BOOKS:

1.	W.K.Pratt, Digital Image Processing, McGraw Hill, New Delhi.
2.	Malay K. Pakhira, Digital Image Processing and Pattern Recognition,Prentice Hall of India, 2011.

WEB LINKS:

1.	https://nptel.ac.in/courses/106/105/106105032/ (NPTEL IIT Kharagpur)
2.	https://www.geeksforgeeks.org/digital-image-processing-basics/
3.	https://sisu.ut.ee/imageprocessing/book/1

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	**Apply** the basic concepts to an acquired Image, Sampling and Quantization.
CO2.	**Apply** linear and nonlinear methods for spatial filtering techniques to a given image.
CO3.	**Apply** image enhancement techniques in spatial domains.
CO4.	**Identify** suitable compression technique for storage and transmission of images
CO5.	**Apply** and demonstrate the Morphological operations and segmentation methods for an image

Self-Study Component:

Students have to learn MatLab/SciLab and prepare a report of analysis of an image of their choice. This report shall be considered as an assignment component and shall be valued for 4 marks.

OOMD Lab

Contact Hours/Week	:	3	Credits	:	1.5
Total Lecture Hours	:	-	CIE Marks	:	50
Total Tutorial Hours	:	-	SEE Marks	:	50
Course Code	:	5RMCAL1			

Course Objectives:

This Course will enable students to:	
1.	Understand the basics of Tools
2.	Define problems of UML and implementation
3.	Understand basics of various design patterns

The student has to draw the necessary UML diagrams using any suitable UML Drawing Tool and implement in Java OR C++ OR C# a program to demonstrate the Design Pattern specified by the Examiner.

The Design Pattern is allotted based on lots from the following list:	
Sl.no	**Implementations**
1.	Expert
2.	Controller
3.	Publisher-Subscriber
4.	Command
5.	Forward-Receive
6.	Client-Dispatcher
7.	Proxy
8.	Façade
9.	Polymorphism

WEB LINKS:

1.	https://www.javatpoint.com/uml
2.	https://www.visual-paradigm.com
3.	https://www.tutorialspoint.com/uml/uml_interaction_diagram.htm
4.	https://sourcemaking.com/design_patterns

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Implement popular communication design patterns in UML convention depicting class model and interaction model and developing Java code over the resulted UML models using Rational Rose builder tool.
CO2.	Implement view handlers and controllers design patterns in UML convention depicting class model and interaction model and developing Java code over the resulted UML models using Rational Rose builder tool.

Mobile Application Development Lab

Contact Hours/Week	:	3	Credits	:	1.5
Total Lecture Hours	:	-	CIE Marks	:	50
Total Tutorial Hours	:	-	SEE Marks	:	50
Course Code	:	5RMCAL2			

Course Objectives:

This Course will enable students to:	
1.	Understand the installation of android studio tools and processes
2.	Identify the concepts and techniques used in creating applications.
3.	Learn how to use databases to store data from android applications
4.	Deploy self-developed applications on android devices.

Sl.no	Implementations
1.	Design an application that contains Phone Contacts in vertical linear manner. Selected contact appears at the top of the list with a large italicized font and a blue background.
2.	Create an application that uses Layout Managers and Event Listeners.
3.	Develop a standard calculator application to perform basic calculations like addition, subtraction, multiplication and division
4.	Devise an application that draws basic graphical primitives (rectangle, circle) on the screen.
5.	Build a mobile application that create, save, update and delete data in a database.
6.	Devise an application that implements Multi-threading.
7.	Develop a mobile application that displays different menu layout.
8.	Create an application that writes data to the SD card.
9	Implement an application for the following i. Creates an alert upon receiving a message.**or** ii. Login form, if success display the login form else display the in valid user
10	Devise a mobile application that creates alarm clock.

WEB LINKS:

1.	https://www.youtube.com/watch?v=giVfVQlKBVM
2.	https://www.youtube.com/watch?v=aS__9RbCyHg
3.	https://www.tutorialspoint.com/android/index.htm
4.	https://developer.android.com/codelabs/build-your-first-android-app#0
5.	https://www.geeksforgeeks.org/android-tutorial/
6.	https://www.tutlane.com/tutorial/android
7.	https://www.w3schools.in/category/android-tutorial/

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Develop applications using software development kits (SDKs), frameworks and toolkits
CO2.	Establish various views methods involved to design app, API in getting map, its location and presenting the apps to the users.
CO3.	Design and develop an application with synchronizing the real time applications like SMS, Email, etc and services

Mini Project-II

Contact Hours/Week	:	3	Credits	:	2.0
Total Lecture Hours	:	-	CIE Marks	:	50
Total Tutorial Hours	:	-	SEE Marks	:	50
Course Code	:	5RMCAL3			

Course Objectives:

This Course will enable students to:	
1.	Be able to develop problem statement
2.	Understand requirement analysis and identify design methodologies
3.	To use computing tools and appropriate programming language

Mini project which are oriented to topics not restricted to DBMS, Image processing, Wireless sensor network, data mining, data analytics is to be developed by each batch consisting of a maximum of two students.

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Identify and formulate problem statement by doing literature surveying variety of allied areas related to computer applications and information technology.
CO2.	Perform requirement analysis and identify design methodologies, apply ethical principles and lifelong learning.
CO3.	Apply advanced programming techniques, computational knowledge to develop solution to real world problems.
CO4.	Apply modern computing tools, programming language that is relevant and appropriate to the project.
CO5.	Work effectively in team and also individually, apply ethical principles and indulge in lifelong learning.
CO6.	Present technical report as a document with clear explanation and effective presentation.

VI Semester

Major Project

Contact Hours/Week	:	-	Credits	:	20.0
Total Lecture Hours	:	-	CIE Marks	:	50
Total Tutorial Hours	:	-	SEE Marks	:	50
Course Code	:	6RMCA01			

Course Objectives:

This Course will enable students to:	
1.	Be able to define an open problem for developing a computational system
2.	Understand various steps of system/ software development
3.	Know about the ethics of organization in general and project development in particular

Part-A	
1	The project may be based on; • Design of a system. • Theoretical/Analytical modelling. • Computer simulation. • Developing working software • Interdisciplinary computer application/modelling.
2	The project could be part of the research activity carried out in the department.
3	The literature survey should be one of the components of the project.
4	The project can be carried outside the institute in a recognized industry/research lab.
5	All the project works should be approved by DAAC.
6	Director and DAAC assign guides for the major project.
7	The project is to be carried out by individual student.

Part- B CIE for the major project	
Comprises of three seminars. Departmental Project Evluation Committee (DPEC) shall evaluate seminars along with the respective guides.	
1	In the *first seminar*, students have to present about the area of the project, literature survey and preliminary requirements of the project and what they propose to do.
2	In the *second seminar*, the students have to present about the project development issues like, specifications, flow chart, design steps, data flow diagrams, data structures, entity relation ship diagrams pertaining to the chosen project
3	In the *third seminar*, the students have to present a demo of the project.
4	CIE is done for a total of 100 marks, ***which shall be reduced to 50***.

The breakup of marks for CIE is given in Table 1.

Table 1. Break up of CIE marks for major project

Seminar-1	20 marks
Seminar-2	40 marks
Seminar-3	40 marks
Total	**100 marks**

*Conducted in presence of external examiner

SEE for the major project

SEE is conducted by one external examiner and the respective guide. The breakup of marks is given in Table 2.

SEE is done for a total of 100 marks, ***which shall be reduced to 50.***

Table 2. Break up of SEE marks for major project

Evaluation of the project report by external examiner and the guide (average of independent evaluations)	50 marks
Presentation, Demonstration and Quality of work, viva-voce	50 marks
Total	**100 marks**

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Identify the relevant topic that is in synchrony with recent trends in information technology by performing the literature survey through standard journal referrals or industry professionals.
CO2.	Identify the problem statement befitting the domain of interest.
CO3.	Interpret the problem, analyze the requirements, develop algorithms, and propose innovative ideas to solve the problem.
CO4.	Implement the project individually and in a team by applying the appropriate software engineering practices and principles.
CO5.	Create, identify, and apply appropriate techniques, resources, and modern computing tools.
CO6.	Commit to professional ethics, and to the norms of computing practice.
CO7.	Apply appropriate testing techniques and develop test cases.
CO8.	Effectively communicate individually and in a team and be able to comprehend and write effective project report including design documentation with high clarity.
CO9.	Practice ethical values and lifelong learning.
CO10.	Provide innovative solutions to societal and environmental issues.

Technical Seminar

Contact Hours/Week	:	-	Credits	:	2.0
Total Lecture Hours	:	-	CIE Marks	:	50
Total Tutorial Hours	:	-	SEE Marks	:	-
Course Code	:	6RMCAS1			

Course Objectives:

This Course will enable students to:	
1.	Identify the topic of the seminar appropriate to the domain
2.	Perform literature survey
3.	Understand the ppt and report preparation techniques

1.	Students have to present the seminar on a chosen topic.
2.	There should be no repetition/duplication of topics among the students.
3.	Students have to submit the reports of their seminar well before the presentation of seminar.
4.	Departmental seminar Evaluation Committee (DSEC) shall evaluate the seminar.
5.	No SEE for this subject.

Course Outcomes:

After the completion of this course, students will be able to:	
CO1.	Perform the literature survey through standard referrals and select the relevant topic that is in synchrony with recent trends in information technology.
CO2.	Gain an understanding of advanced methodologies in the chosen field, including theory, and interdisciplinary approaches.
CO3.	Prepare a visual presentation depicting the synthesis of the topic with required illustrations, interpretations and clear elaboration of the information gathered and to provide valid conclusions.
CO4.	Make an oral presentation effectively among the peers and computing community.
CO5.	Demonstrate through short written seminar report the ability to comprehend and summarize.