DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Scheme of Instruction

and

Syllabi of

B.E. VII & VIII- SEMESTERS

2018-2019

UNIVERSITY COLLEGE OF ENGINEERING

(AUTONOMOUS)

OSMANIA UNIVERSITY

HYDERABAD – 500 007, TELANGANA

SCHEME OF INSTRUCTION
BE (COMPUTER SCIENCE & ENGINEERING)
CSE: SEMESTER – VII

S.No	Course Code	Course Title	Scheme of Instruction			Contact Hrs/Wk	Scheme of Examination		Credits
			L	T	P		CIE	SEE	
Theory									
1.	PC 701 CS	Compiler Construction	3	1	0	4	30	70	3
2.	PC 702 CS	Distributed Systems	3	1	0	4	30	70	3
3.	PC 703CS	Information Security	3	1	0	4	30	70	3
4.	PC 704 CS	Embedded System Design	3	1	0	4	30	70	3
5.	PE-III	Professional Elective-III	3	1	0	4	30	70	3
6.	OE-II	Open Elective-II	3	0	0	3	30	70	3
Practicals									
8.	PC 751 CS	Compiler Construction Lab	0	0	2	2	25	50	1
9.	PC 752 CS	Distributed Systems Lab	0	0	2	2	25	50	1
	PC 753 CS	Embedded Systems Lab	0	0	2	2	25	50	1
10.	PW761CS	Project Work-II	0	0	2	2	50	--	4
11.	PW961CS	Summer Internship	0	0	0	0	50	--	2
Total			**18**	**05**	**08**	**31**	**355**	**570**	**27**

	Professional Elective-III
PE705CS	Mobile Computing
PE706CS	Image Processing
PE707CS	Software Quality and Testing
PE708CS	Web Services And Architecture
PE709CS	Computational Intelligence

	Open Elective-II
OE701BM	Human Factor Engineering
OE702BM	Basic Medical Engineering
OE701CE	Optimization Techniques
*OE701CS	Data Base Management Systems
*OE702CS	Information Security
OE701EC	Principles of electronic communication
**OE702EC	Fundamentals of IOT
OE701EE	Non-conventional Energy Sources
OE701ME	Startup Entrupreunership
OE702ME	Finite Element Methods

*CS Electives offered for BME/CE/EC/EE/ME branches only

**ECE Elective offered for BME/CE/EE/ME branches only

PC 701 CS With effect from the academic year 2018-2019

COMPILER CONSTRUCTION

Instruction	4 Periods per week
Duration of University Examination	3 Hours
University Examination	70 Marks
Sessionals	30 Marks
Credits	3

Course Objectives:

- To introduce the steps in language translation pipeline and runtime data structures used in translation
- To learn about Scanning (lexical analysis) process using regular expressions and use of LEX to generate scanner
- To introduce different Parsing strategies including top-down (e.g., recursive descent, Earley parsing, or LL) and bottom-up (e.g., backtracking or LR) techniques
- Describe semantic analyses using an attribute grammar
- To learn how to build symbol tables and generate intermediate code.
- To introduce techniques of program analysis and code optimization

Course Outcomes:

Student will be able to :

- Create lexical rules and grammars for a given language
- Generate scanners and parsers from declarative specifications.
- Describe an abstract syntax tree for a small language.
- Use program analysis techniques for code optimization
- Develop the compiler for a subset of a given language

UNIT – I

Introduction: Compilers, The translation process, Data structures and issues in compiler structure, Bootstrapping and Porting.

Scanning: The scanning process, Regular expressions, Finite Automata, Regular expressions to DFA's, use of LEX to generate scanner.

UNIT – II

Context Free Grammars & Parsing: The parsing process, Context free grammars, Parse tree & Abstract syntax trees, EBNF and syntax diagrams, and Properties of CFLs.

Top Down Parsing: Recursive descent parsing, LL (1) parsing, First and follow sets, Recursive descent parser, and Error recovery in top down parsers.

UNIT – III

Bottom-up Parsing: Overview, LR (0) items and LR (0) Parsing, SLR (1) Parsing, general LR(1) and LALR(1) parsing, YACC, and Error recovery in bottom-up parsers.

UNIT – IV

Semantic Analysis: Attributes and attribute grammars, Algorithms for attribute computation, Symbol table, Data types and Type checking.

Runtime Environments: Memory organization during program execution, Fully static runtime environments, Stack-based runtime environments, Dynamic memory, and Parameter parsing mechanisms.

UNIT – V

Code Generation: Intermediate code and data structures for code generation, Basic code generation techniques, Code generation of data structure references, Code generation of control statements and logical expressions, Code generation of procedure and function calls, Code generation in commercial compilers, Code optimization techniques, and Data flow equation.

Suggested Readings:

1. Kenneth C. Louden, "*Compiler Construction: Principles and Practice*", Thomson Learning Inc., 1997.
2. Ravi Sethi, Aho & Ullman JP, "*Compilers: Principles, Techniques and Tools*", Addison Wesley publishing co., 1986.
3. J.P. Tremblay and P.S. Sorenson, "*The Theory and Practice of Compiler Writing*", TMH-1985.

PC 702 CS With effect from the academic year 2018-2019

DISTRIBUTED SYSTEMS

Instruction	4 Periods per week
Duration of University Examination	3 Hours
University Examination	70 Marks
Sessionals	30 Marks
Credits	3

Course Objectives:

- To acquire an understanding of the issues in distributed systems
- To study architectures and working of distributed file systems
- To expose the students to distributed transaction management, security issues and replication

Course Outcomes:

Student will be able to :

- Describe the problems and challenges associated with distributed systems.
- Implement small scale distributed systems .
- Understand design tradeoffs in large-scale distributed systems

UNIT-I

Introduction: Goals and Types of Distributed Systems

Architectures: Architectural Styles, System Architectures, Architectures versus Middleware, and Self-Management in Distributed Systems.

Processes: Threads, Virtualization, Clients, Servers, and Code Migration.

Communication: Fundamentals, Remote Procedure Call, Message-Oriented Communication, Stream-Oriented Communication, and Multicast Communication.

UNIT-II

Naming: Names, Identifiers and Addresses, Flat Naming, Structured Naming, and Attribute-Based Naming.

Synchronization: Clock Synchronization, Logical Clocks, Mutual Exclusion, Global Positioning of Nodes, and Election Algorithms.

Consistency and Replication: Introduction, Data-Centric Consistency Models, Client-Centric Consistency Models, Replica Management, and Consistency Protocols.

UNIT-III

Fault Tolerance: Introduction to Fault Tolerance, Process Resilience, Reliable Client-Server Communication, Reliable Group Communication, Distributed Commit, and Recovery.

Distributed Object-Based Systems: Architecture, Processes, Communication, Naming, Synchronization, Consistency and Replication, Fault Tolerance, and Security.

UNIT-IV

Distributed File Systems: Architecture, Processes, Communication, Naming, Synchronization, Consistency and Replication, Fault Tolerance, and Security.

Distributed Web-Based Systems: Architecture, Processes, Communication, Naming, Synchronization, Consistency and Replication, Fault Tolerance, and Security.

UNIT-V

Distributed Coordination-Based Systems: Introduction to Coordination Models, Architecture, Processes, Communication, Naming, Synchronization, Consistency and Replication, Fault Tolerance, and Security.

Map-Reduce: Example, Scaling, programming model, Apache Hadoop, Amazon Elastic Map Reduce, Mapreduce.net, Pig and Hive.

Suggested Readings:

1. Andrew S. Tanenbaum and Maarten Van Steen, "*Distributed Systems*", PHI 2nd Edition, 2009.
2. R.Hill, L.Hirsch, P.Lake, S.Moshiri, "*Guide to Cloud Computing*, Principles and Practice", Springer, 2013.
3. R.Buyya, J.Borberg, A.Goscinski,"*Cloud Computing-Principles and Paradigms*",Wiley 2013.

PC 703 CS With effect from the academic year 2018-2019

INFORMATION SECURITY

Instruction	4 Periods per week
Duration of University Examination	3 Hours
University Examination	70 Marks
Sessionals	30 Marks
Credits	3

Course Objectives:

- To learn legal and technical issues in building secure information systems
- To provide an understanding of network security
- To expose the students to security standards and practices

Course Outcomes:

Student will be able to:

- Describe the steps in Security Systems development life cycle(SecSDLC)
- Understand the common threats and attack to information systems
- Understand the legal and ethical issues of information technology
- Identify security needs using risk management and choose the appropriate risk control strategy based on business needs
- Use the basic knowledge of security frameworks in preparing security blue print for the organization
- Usage of reactive solutions, network perimeter solution tools such as firewalls, host solutions such as antivirus software and Intrusion Detection techniques and knowledge of ethical hacking tools
- Use ethical hacking tools to study attack patterns and cryptography and secure communication protocols
- Understand the technical and non-technical aspects of security project implementation and accreditation

UNIT-I

Introduction: History, Critical Characteristics of Information, NSTISSC Security Model, Components of an Information System, Securing the Components, Balancing Security and Access, The SDLC, The Security SDLC.
Need for Security: Business Needs, Threats, Attacks, and Secure Software Development

UNIT-II

Legal, Ethical and Professional Issues: Law and ethics in Information Security, Relevant U.S. Laws, International Laws and Legal Bodies, Ethics and Information Security.

Risk Management: Overview, Risk Identification, Risk Assessment, Risk Control Strategies, Selecting a Risk Control Strategy, Quantitative versus Qualitative Risk Control Practices, Risk Management discussion Points, Recommended Risk Control Practices.

UNIT-III

Planning for Security: Security policy, Standards and Practices, Security Blue Print, Security Education, Continuity strategies.
Security Technology: Firewalls and VPNs: Physical Design, Firewalls, Protecting Remote connections.

UNIT-IV

Security Technology: Intrusion Detection, Access Control, and other Security Tools: Intrusion Detection and Prevention Systems-Scanning, and Analysis Tools- Access Control Devices.
Cryptography: Foundations of Cryptology, Cipher methods, Cryptographic Algorithms, Cryptographic Tools, Protocols for Secure Communications, Attacks on Cryptosystems

UNIT-V

Implementing Information Security: Information security project management, Technical topics of implementation, Non Technical Aspects of implementation, Security Certification and Accreditation.
Security and Personnel: Positioning and staffing security function, Employment Policies and Practices, and Internal control Strategies.
Information Security Maintenance: Security management models, Maintenance model, and Digital Forensics.

Suggested Readings:

1. Michael E Whitman and Herbert J Mattord, "*Principles of Information Security*", Cengage Learning, 2011.
2. Thomas R Peltier, Justin Peltier, John Blackley, "*Information Security Fundamentals*", Auerbach Publications, 2010.
3. Detmar W Straub, Seymour Goodman, Richard L Baskerville, "*Information Security, Policy, Processes, and Practices*", PHI, 2008.
4. Mark Merkow and Jim Breithaupt " *Information Security Principle and Practices*", Pearson Education, 2007

PC 704 CS With effect from the academic year 2018-2019

EMBEDDED SYSTEM DESIGN

Instruction	4 Periods per week
Duration of University Examination	3 Hours
University Examination	70 Marks
Sessionals	30 Marks
Credits	3

Course Objectives :

- To provide basics of embedded systems design and development flow.
- To study the processor architectures that supports embedded systems.
- To gain knowledge developing platforms for embedded systems.
- To provide basics of real time operating systems that supports embedded systems.
- To study the concepts on testing and development tools.

Course Outcomes :

Student will be able to :

- Understand the basics of embedded systems design and development flow.
- Apply knowledge to develop the embedded systems.
- Analyse the real time operating that supports embedded systems.

UNIT- I

Design of Embedded System: Sensors and Actuators, Embedded Processors, Memory Architectures, Input and Output.

UNIT –II

Embedded Systems development Environment: IDE, Cross compilation, Disassembler, Simulators, Emulators and Debugging, Target hardware debugging, Boundary Scan.

Embedded Computing Platform: Programming for Embedded systems using C, Device drivers, program modeling concepts, Process of Embedded system development: embedded software development on microcontroller platform, network-based embedded applications and embedded control applications.

UNIT-III

Embedded C Programming: Review of data types - Scalar types-Primitive types-Enumerated types-Subranges, Structure types-character strings -arrays- Functions. Interfacing C with Assembly. Embedded programming issues - Reentrancy, Portability, Optimizing, and testing embedded C programs.

UNIT- IV

Concept of Embedded Operating Systems, Differences between Traditional OS and RTOS. Real-time System Concepts, RTOS Kernel & Issues in Multitasking – Task Assignment, Task Priorities, Scheduling, Intertask Communication & Synchronization – Definition of Context Switching, Foreground ISRs and Background Tasks. Critical Section – Reentrant Functions, Interprocess Communication (IPC) – IPC through Semaphores, Mutex, Mailboxes, Message Queues or Pipes and Event Flags.

UNIT –V

VxWorks – POSIX Real Time Extensions, timeout features, Task Creation, Semaphores (Binary, Counting), Mutex, Mailbox, Message Queues, Memory Management – Virtual to Physical Address Mapping.

Suggested Readings:

1. Edward Ashford Lee and Sanjit Arun kumar Seshia, "Introduction to Embedded Systems- A cyber-Physiacal Systems Approach, Second Edition, MIT Press, 2017.
2. Jones, M Tim, *GNU/Linux Application Programming*, 2nd Edition, Course Technology PTR, 2008.
3. Raj Kamal, " Embedded systems Architecture, programming & Design", Tata McGraw Hill, 2010.
4. Real Time Systems, C.M.Krishna and G.Shin, McGraw-Hill Companies Inc., McGraw Hill International Edition, 1997.
5. Programming Embedded Systems with C and GNU Development Tools, Second Edition, 1977.

PE 708 CS *With effect from the Academic year 2018-2019*

WEB SERVICES AND ARCHITECTURE
(PROFESSIONAL ELECTIVE-III)

Instruction	4 Periods per week
Duration of University Examination	3 Hours
University Examination	70 Marks
Sessionals	30 Marks
Credits	3

Course Objectives

- To study the evolution of SOA and Web Services
- To understand the principles of service orientation , Service layers
- To learn about WS* Specifications , messaging with SOAP and Service composition
- To learn about service oriented analysis and service oriented design
- Gained knowledge on various open standards available for developing SOA compliant web services

Course Outcomes:

Student will be able to :

- Understand web service framework with respect to SOA
- Develop SOA compliant web services using open standards and various technologies
- Model and implement businesses processes using service oriented approach

UNIT- I :
SOA and Web Services Fundamentals: Introducing So, The Evolution of SOA, Web services and primitive SOA.

UNIT-II:
SOA and WS-*Extensions: Web Services and Contemporary SOA(I: Activity Management and Composition), Web Services and Contemporary SOA(II: Advanced Messaging, Metadata, and Security).

UNIT-III:
SOA and Service-Orientation: Principles of Service-Orientation, Service Layers.

UNIT-IV:
Building SOA (Planning And Analysis) : SOA Delivery Strategies, Services-Oriented Analysis (I: Introduction), Service-Oriented Analysis (II: Service Modeling).

UNIT-V:
Building SOA (Technology And Design): Service-Oriented Design (I: Introduction), Service-Oriented Design (II: SOA Composition Guidelines), Service-Oriented Design (III: Service-Design), Service-oriented Design (IV: Business Process Design), Fundamentals WS-*Extensions, SOA Platforms.

Suggested Readings

1. Thomas Eri, "*Service-Oriented Architecture(SOA): Concepts, Technology, and Design*", Prentice Hall PTR, 2005
2. James McGovern and Sameer Tyagi , "*Java Web Services Architecture*,", Morgan Kaufmann-May 2003.

OE701ME *With effect from the Academic year 2018-2019*

STARTUP ENTREPRENEURSHIP

(Open Elective-II)

Instruction	3 Periods per week
Duration of University Examination	3 Hours
University Examination	70 Marks
Sessionals	30 Marks
Credits	3

Course Objectives:

- To motivate students to take up entrepreneurship in future
- To learn nuances of starting an enterprise by creative thinking and shape ideas into reality.
- To understand action driven business plan and learn to prepare project budget.

Course Outcomes:

Students will be able to :

- Think creatively and and transform ideas into reality.
- Differentiate market transforming strategy.
- Create a complete business plan and workout the budget plan.

UNIT I: Creativity & Discovery

Definition of Creativity, self test creativity, discovery and delivery skills, The imagination threshold, Building creativity ladder, Collection of wild ideas, Bench marking the ideas, Innovative to borrow or adopt, choosing the best of many ideas, management of tradeoff between discovery and delivery, Sharpening observation skills, reinventing self, Inspire and aspire through success stories

UNIT II: From Idea to Startup

Introduction to think ahead backward, Validation of ideas using cost and strategy, visualizing the business through value profile, activity mapping, Risks as opportunities, building your own road map

UNIT III: Innovation career lessons

Growing & Sharing Knowledge, The Role of Failure In Achieving Success, Creating vision, Strategy, Action & Resistance: Differentiated Market Transforming Strategy; Dare to Take Action; Fighting Resistance; All About the startup Ecosystem; Building a Team; Keeping it Simple and Working Hard.

UNIT IV: Action driven business plan

Creating a completed non-business plan (a series of actions each of which moves your idea toward implementation), including a list of the activities to be undertaken, with degrees of importance (scale of 1 to 3, where 1 is 'most important'). A revision of the original product or service idea, in light of information gathered in the process, beginning to design the business or organization that will successfully implement your creative idea. Preparing an activity map.

UNIT V: Startup financing cycle

Preparing an initial cash flow statement, showing money flowing out (operations; capital) and flowing in. Estimate your capital needs realistically. Prepare a bootstrapping option (self financing). Prepare a risk map. Prepare a business plan comprising five sections: The Need; The Product; Unique Features; The Market; Future Developments. Include a Gantt chart (project plan – detailed activities and starting and ending dates); and a project budget.

Suggested Readings:

1. Vasant Desai, "Dynamics of Entrepreneurial Development and Management", Himalaya Publishing House, 1997.
2. Prasanna Chandra, "Project – Planning , Analysis, Selection, Implementation and Review", Tata McGraw-Hill Publishing Company Ltd., 1995.
3. B. Badhai, "Entrepreneurship for Engineers", Dhanpath Rai & Co., Delhi, 2001.
4. Stephen R. Covey and A. Roger Merrill, "First Things First", Simon and Schuster, 2002.
5. Robert D. Hisrich and Michael P.Peters, " Entrepreneurship", Tata McGRaw Hill Edition, 2002

PC 751 CS *With effect from the Academic year 2018-2019*

COMPILER CONSTRUCTION LAB

Instruction	2 Periods per week
Duration of University Examination	3 Hours
University Examination	50 Marks
Sessionals	25 Marks
Credits	1

Course Objectives:

- To learn usage of tools LEX, YAAC
- To develop a code generator
- To implement different code optimization schemes

Course Outcomes:

Student will be able to:

- To Generate scanner and parser from formal specification
- To design a compiler for a subset of any High level language

1. Construction of DFA from NFA
2. Scanner program using LEX
3. Construction of a Predictive parsing Table
4. SLR Parser table generation
5. Implement unification Algorithm
6. LR Parser table generation
7. Parser Generation using YACC
8. Write a program on code generation
9. Write a program on code optimization

PC 752 CS *With effect from the Academic year 2018-2019*

DISTRIBUTED SYSTEMS LAB

Instruction	2 Periods per week
Duration of University Examination	3 Hours
University Examination	50 Marks
Sessionals	25 Marks
Credits	1

Course Objectives:

- To implement client and server programs using sockets
- To learn about working of NFS
- To use Map Reduce model for distributed processing
- To develop mobile applications

Course Outcomes:

Student will be able to :

- Write programs that communicate data between two hosts
- Configure NFS
- Use distributed data processing frameworks and mobile application tool kits

1. Implementation FTP Client
2. Implementation of Name Server
3. Implementation of Chat Server
4. Understanding of working of NFS (Includes exercises on Configuration of NFS)
5. Implementation of Bulletin Board.
6. Implement a word count application which counts the number of occurrences of each word a large collection of documents Using Map Reduce model.
7. Develop an application (small game-like scrabble, Tic-tac-Toe) using Android SDK.

PC 753 CS *With effect from the Academic year 2018-2019*

EMBEDDED SYSTEMS LAB

Instruction	2 Periods per week
Duration of University Examination	3 Hours
University Examination	50 Marks
Sessionals	25 Marks
Credits	1

Course Objectives:

- To develop knowledge in programming techniques using embedded C language.
- To develop programming skill in understanding the interfacing techniques.
- To gain practical knowledge in scheduling and multitasking issues for Embedded System

1. Programs using Embedded C
2. Experiments to interface and to access all internal and external peripherals such as
   a. Stepper motor interface.
   b. LCD interface.
   c. LED interface.
   d. Keyboard interface.
   e. Serial and DAC system interface
3. Experiments on RTOS Applications using VxWorks
4. Practical implementation concepts of RTOS
   - Scheduling
   - Multiple Processes

PW 761 CS *With effect from the academic year 2018-2019*

PROJECT WORK –I

Instruction	2 Periods per week
Sessional	50 Marks
Credit	2

Course Objectives:

- To enhance practical and professional skills.
- To familiarize tools and techniques of systematic Literature survey and documentation
- To expose the students to industry practices and team work.
- To encourage students to work with innovative and entrepreneurial ideas

Course Outcomes:

Student will be able to :

- Demonstrate the ability to synthesize and apply the knowledge and skills acquired in the academic program to real-world problems
- Evaluate different solutions based on economic and technical feasibility
- Effectively plan a project and confidently perform all aspects of project management
- Demonstrate effective written and oral communication skills

The department can initiate the project allotment procedure at the end of VI semester and finalize it in the first two weeks of VII semester.

The department will appoint a project coordinator who will coordinate the following:

Collection of project topics/ descriptions from faculty members (Problems can also be invited from the industries)

Grouping of students (max 3 in a group)

Allotment of project guides

The aim of project work is to develop solutions to realistic problems applying the knowledge and skills obtained in different courses , new technologies and current industry practices. This requires students to understand current problems in their domain and methodologies to solve these problems. To get awareness on current problems and solution techniques, the first 4 weeks of VII semester will be spent on special lectures by faculty members, research scholars, post graduate students of the department and invited lectures by engineers from industries and R&D institutions. After completion of these seminars each group has to formalize the project proposal based on their own ideas or as suggested by the project guide.

Seminar schedule will be prepared by the coordinator for all the students from the 5th week to the last week of the semester which should be strictly adhered to.

Each group will be required to:

1. Submit a one page synopsis before the seminar for display on notice board.
2. Give a 30 minutes presentation followed by 10 minutes discussion.
3. Submit a technical write-up on the talk.

At least two teachers will be associated with the Project Seminar to evaluate students for the award of sessional marks which will be on the basis of performance in all the 3 items stated above.

The seminar presentation should include the following components of the project:

*Problem definition and specification

*Literature survey

*Broad knowledge of available techniques to solve a particular problem.

*Planning of the work, preparation of bar (activity) charts

*Presentation- oral and written.

PW 961 CS *With effect from the academic year 2018-2019*

SUMMER INTERNSHIP

University Examination	50 Marks
Credits	2

Course Objectives:

- To train and provide hands-on experience in analysis, design, and programming of information systems by means of case studies and projects.
- To expose the students to industry practices and team work.
- To provide training in soft skills and also train them in presenting seminars and technical report writing.

Course Outcomes:

Student will be able to :

- Get Practical experience of software design and development, and coding practices within Industrial/R&D Environments.
- Gain working practices within Industrial/R&D Environments.
- Prepare reports and other relevant documentation.

Summer Internship is introduced as part of the curricula of encouraging students to work on problems of interest to industries. A batch of three students will be attached to a person from the Computer Industry/Software Companies/R&D Organization for a period of 8 weeks. This will be during the summer vacation following the completion of the III year Course. One faculty coordinator will also be attached to the group of 3 students to monitor the progress and to interact with the industry co-ordinate (person from industry).

After the completion of the project, student will submit a brief technical report on the project executed and present the work through a seminar talk to be organized by the Department. Award of sessionals are to be based on the performance of the students, to be judged by a committee constituted by the department. One faculty member will co-ordinate the overall activity of Industry Attachment Program.

UNIT- III

Linear Regression Using R: Introduction, Model Fitting, Linear Regression, Assumptions of Linear Regression, Validating Linear Assumption.

Logistic Regression: Introduction, What Is Regression?, Introduction To Generalized Linear Model, Logistic Regression, Binary Logistic Regression, Diagnosing Logistic Regression, Multinomial Logistic Regression Model.

UNIT IV

Decision Tree: Introduction, What Is A Decision Tree?, Decision Tree Representation In R, Appropriate Problems For Decision Tree Learning, Basic Decision Tree Learning Algorithm, Measuring Features, Hypothesis Space Search In Decision Tree Learning, Inductive Bias In Decision Tree Learning, Why Prefer Short Hypotheses, Issues In Decision Tree Learning.

Time Series In R: Introduction, What Is Time Series Data, Reading Time Series Data, Decomposing Time Series Data, Forecasts Using Exponential Smoothing, ARIMA Models.

UNIT-V

Clustering: Introduction, What Is Clustering, Basic Concepts in Clustering, Hierarchical Clustering, K-Means Algorithm, CURE Algorithm, Clustering in Non-Euclidean Space, Clustering for Streams and Parallelism.

Association Rules: Introduction, Frequent Itemset, Data Structure Overview, Mining Algorithm Interfaces, Auxiliary Functions, Sampling from Transaction, Generating Synthetic Transaction Data, Additional Measures of Interestingness, Distance Based Clustering Transaction and Association.

Text Mining: Introduction, Definition of Text Mining, A Few Challenges in Text Mining, Text Mining Verses Data Mining, Text Mining In R, General Architectures of Text Mining Systems, Pre-Processing of Documents In R, Core Text Mining Operations, Using Background Knowledge for Text Mining, Text Mining Query Languages.

Mining Frequent Patterns, Associations and Correlations: Basic Concepts and Methods.
Frequent Itemset, Closed Itemset And Association Rules.
Frequent Itemset: Mining Methods, Pattern Evaluation Methods, Sentiment Analysis

Suggested Readings:

1. Data Analytics using R by Seema Acharya. McGraw Hill education.
2. Practical Data Science with R, Nina Zumel and John Mount, Manning Shelter Island.
3. The R book, Crawley, Michael J. John Wiley & Sons, Ltd