SRI VASAVI ENGINEERING COLLEGE (**AUTONOMOUS**)
(Sponsored by Sri Vasavi Educational Society)
(Approved by AICTE, New Delhi & Permanently affiliated to JNTUK, Kakinada)
(Accredited by NAAC with 'A' Grade ,Recognized by UGC under section 2(f) & 12(B))
Pedatadepalli, **TADEPALLIGUDEM – 534 101.** W.G.Dist**. (A.P)**
Department of Computer Science & Engineering (Accredited by NBA)

Minutes of the sixth Board of Studies held on 25/07/2022 at 02.00 PM through online mode

Date: 25.07.2022

The 6th Meeting of Board of Studies in Department of Computer Science and Engineering is held at 02.00 PM on 25-07-2022 through online mode using,

https://us02web.zoom.us/j/86544550258

The following members attended the meeting:

S.No.	Name of the Member	Designation	Role
1.	Dr. D Jaya Kumari	Professor, HoD-CSE, SVEC	Chairperson
2.	Dr.Krishna Mohan Ankala	Professor, UCEK, Kakinada	University Nominee
3.	Dr. R.B.V. Subramanyam	Professor, Department of CSE, NIT Warangal	Academic Expert
4.	Dr. S Pallam Setty	Professor, Department of CSE, Andhra University, Vishakapatnam	Academic Expert
5.	Sri. Srinivasa Raju Vuppalapati	Senior Consultant, MSR IT Services LLP, Hitech City, Hyderabad.	Industry Expert
6.	Mr.EEdala Rambabu	microfocus, Bangalore	Alumni
7.	Sri Ch. Apparao	Technical Director	Invited Member
8.	Dr. V. Venkateswara Rao	Professor	Member
9.	Dr. G Loshma	Professor	Member
10.	Dr. V S Naresh	Professor	Member
11.	Dr. Ch. Raja Ramesh	Associate Professor	Member
12.	Dr.K. Shirin Bhanu	Associate Professor	Member
13.	Dr. P Laxmikanth	Associate Professor	Member
14.	D Anjani Suputhri Devi	Sr. Assistant Professor	Member
15.	R. LeelaPhani Kumar	Assistant Professor	Member
16.	D Sasi Rekha	Assistant Professor	Member
17.	G.Sriram Ganesh	Assistant Professor	Member
18.	N.V.Murali Krishna Raja	Assistant Professor	Member
19.	N. Hiranmayee	Assistant Professor	Member
20.	K Lakshmi Narayana	Assistant Professor	Member
21.	D.S L Manikanteswari	Assistant Professor	Member
22.	M V V Krishna	Assistant Professor	Member
23.	M Baburao	Assistant Professor	Member
24.	P Rajesh	Assistant Professor	Member
25.	K Satyanarayana	Assistant Professor	Member
26.	M Sree Radha Mangamani	Assistant Professor	Member
27.	M Chilaka Rao	Assistant Professor	Member
28.	A Nageswara Rao	Assistant Professor	Member
29.	A NagaJyothi	Assistant Professor	Member
30.	G Prashanthi	Assistant Professor	Member
31.	M Yesu Shekharam	Assistant Professor	Member
32.	K Praveen Kumar	Assistant Professor	Member
33.	G Nagavallika	Assistant Professor	Member
34.	N V Ratna Kishore	Assistant Professor	Member
35.	G Jaya Raju	Assistant Professor	Member
36.	P Bhanu Rangarao	Assistant Professor	Member
37.	M N V Surekha	Assistant Professor	Member
38.	R Padmaja	Assistant Professor	Member

The following are the Minutes of the Meeting

Item No.1: Welcome note by the Chairperson BOS.

Chairperson BOS extended a formal welcome and introduced the members.

Item No.2: Progress Report of the Department

Chairperson BOS had given the Brief Progress Report of the Department.

Item No.3: Review and Approval of Course Structure for I to VIII Semesters of B.Tech(CSE) and B.Tech(CST) Programmes under V20 Regulation.

The approved Course Structure is given in **Annexure-I.**

SEM	Course Code	Suggestions	Inclusions / Modifications
VI	V20CSTPE08	Move CNS Course from VII SEM to VI SEM	Moved CNS Course VI SEM
VII	V20CSTPE13	Remove OOAD course and replace with DevOps / Design Patterns	OOAD Course replaced with Design Patterns
VII	V20CSTPE15	Replace Distributed systems course with Data Visualization / Reinforcement Learning	Distributed systems course replaced with Reinforcement Learning
Skill Oriented Course		Remove Source code Management in SOC Pool	Removed

The revised Course Structure is given in **Annexure-I.**

Item No.4: Approval of Syllabi for the Proposed Courses offered in V to VIII Semesters of B.Tech(CSE) and B.Tech(CST) Programme under V20 Regulation.

Approved the Syllabi for the courses offered in V to VIII semesters of B.Tech(CSE) and B.Tech(CST) Programme under V20 Regulation and suggested the following changes:

SEM	Course Code	Suggestions	Inclusions / Modifications
VI	V20CST14	In Machine Learning Course UNIT - II & III need to be revised	Revised
VI	V20CSTPE07	In Data Science Course merge UNIT-I & II and replace Regression methods with new concepts and add one more UNIT.	As per the Suggestion Changes has been done

The detailed syllabus is given in **Annexure-II**

Item No. 5: Approval of list of Courses offering under Job Oriented Elective-I to Job Oriented Elective-IV in V to VII Semesters of B.Tech(CSE) and B.Tech(CST) Programme respectively under V20 Regulation and the approval of their Syllabi.

Approved list of Courses under Job Oriented Elective-I to Job Oriented Elective-IV in V to VII Semesters of B.Tech(CSE) and B.Tech(CST) Programme respectively under V20 Regulation and their Syllabi.

The details are given in **Annexure-III.**

Item No. 6: Approval of list of Courses offering under Open Elective-I to Open Elective-IV in V and VII Semesters respectively under V20 Regulation for all other branches and the approval of their Syllabi.

Approved list of Courses under Open Elective-I to Open Elective-IV in V and VII Semesters respectively under V20 Regulation for all other branches and their Syllabi.

The details are given in **Annexure-IV.**

Item No. 7: Approval of B.Tech(Hons) & B.Tech(Minors) offered by CSE in line with the guidelines prescribed by APSCHE.

Approved B.Tech(Hons) & B.Tech(Minors) offered by CSE in line with the guidelines prescribed by APSCHE. However the list is not exhaustive. Before registering the course take the approval from HOD.

The details are given in **Annexure-V**

Chairperson of BOS
(Dr.D Jaya Kumari)

Head of the Department
Dept. of Computer Science & Engineering
Sri Vasavi Engineering College
TADEPALLIGUDEM-534 101

Annexure-I

SEMESTER – I (First Year)

S.No.	Code	Name of the Course		L	T	P	C
1	V20MAT01	Linear Algebra and Differential Equations	BSC	3	0	0	3
2	V20CHT01	Engineering Chemistry	BSC	3	0	0	3
3	V20ENT01	English for Professional Enhancement	HSS	3	0	0	3
4	V20MEL02	Engineering Workshop	ESC	1	0	4	3
5	V20CST01	Programming in 'C' for problem Solving	ESC	3	0	0	3
6	V20ENL01	Hone Your Communication Skills Lab -I	HSS	0	0	3	1.5
7	V20CHL01	Engineering Chemistry Lab	BSC	0	0	3	1.5
8	V20CSL01	Programming Lab in 'C' for problem Solving	ESC	0	0	3	1.5
Total:				**13**	**0**	**13**	**19.5**

Total Contact Hours: 26 **Total Credits: 19.5**

SEMESTER – II (First Year)

S.No.	Code	Name of the Course		L	T	P	C
1	V20MAT02	Numerical Methods and Vector Calculus	BSC	3	0	0	3
2	V20PHT01	Engineering Physics	BSC	3	0	0	3
3	V20ECT01	Switching Theory and Logic Design	ESC	3	0	0	3
4	V20CST02	Python Programming	ESC	3	0	0	3
5	V20MEL01	Engineering Graphics	ESC	1	0	4	3
6	V20PHL01	Engineering Physics Lab	BSC	0	0	3	1.5
7	V20CSL02	Python Programming Lab	ESC	0	0	3	1.5
8	V20ENL02	Hone Your Communication Skills Lab -II	HSS	0	0	3	1.5
9	V20CHT02	Environmental Science	MNC	0	0	0	0
			Total:	**13**	**0**	**13**	**19.5**

Total Contact Hours: 26 **Total Credits: 19.5**

SEMESTER-III (SECOND YEAR)

S.No.	Code	Name of the Course		L	T	P	C
1	V20MBT51	Managerial Economics and Financial Analysis	HSS	3	0	0	3
2	V20MAT07	Mathematical Foundation Of Computer Science	ESC	3	0	0	3
3	V20CST03	OOPs Through C++	PCC	3	0	0	3
4	V20CST04	Data Structures	PCC	3	0	0	3
5	V20CST05	Computer Organization and Architecture	ESC	3	0	0	3
6	V20CSL03	OOPs Through C++ Lab	PCC	0	0	3	1.5
7	V20CSL04	Data Structures Lab	PCC	0	0	3	1.5
8	V20CSL05	Linux Shell Scripting Lab	PCC	0	0	3	1.5
9	V20SOC01	**Skill Oriented Course-I**	**SOC**	**1**	**0**	**2**	**2**
10	V20ENT02	Professional Communication Skills -I	MNC	2	0	0	0
			Total:	**18**	**0**	**11**	**21.5**

Total Contact Hours: 29 **Total Credits: 21.5**

SEMESTER - IV (SECOND YEAR)

S.No.	Code	Name of the Course		L	T	P	C
1	V20CST06	Design and Analysis of Algorithms	PCC	3	0	0	3
2	V20CST07	Software Engineering	PCC	3	0	0	3
3	V20CST08	Database Management Systems	PCC	3	0	0	3
4	V20CST09	Java Programming	PCC	3	0	0	3
5	V20MAT04	Probability and Statistics	BSC	**3**	0	0	**3**
6	V20CSL06	Statistical Visualization using R Lab	BSC	0	0	3	1.5
7	V20CSL07	Database Management Systems Lab	PCC	0	0	3	1.5
8	V20CSL08	Java Programming Lab	PCC	0	0	3	1.5
9	V20SOC02	Skill Oriented Course-II	SOC	1	0	2	2
10	V20ENT03	Professional Communication Skills -II	**MNC**	**2**	0	0	**0**
			Total:	**18**	**0**	**11**	**21.5**

Total Contact Hours: 29 **Total Credits: 21.5**

V SEMESTER (THIRD YEAR)

S.No.	Code	Name of the Course		L	T	P	C
1	V20CST10	Operating Systems	PCC	3	0	0	3
2	V20CST11	Data Mining	PCC	3	0	0	3
3	V20CST12	Web Technologies	PCC	3	0	0	3
4		**Open Elective -I / Job Oriented Elective-I**	OEC	3	0	0	3
			JOE	0	0	6	3
5	**Professional Elective-I**		PEC	3	0	0	3
	V20CSTPE01	Software Testing Methodologies					
	V20CSTPE02	Principles of Programming Languages					
	V20CSTPE03	Artificial Intelligence					
	V20CSTPE04	Computer Graphics					
6	V20CSL09	Data Mining Lab	PCC	0	0	3	1.5
7	V20CSL10	Web Technologies Lab	PCC	0	0	3	1.5
8	V20SOC03	Skill Oriented Course-III (Soft Skills)	SO/SS	1	0	2	2
9	V20CSP01	Mini Project / Internship	Internship	0	0	3	1.5
10	V20ENT04	Professional Communication Skills -III	**MNC**	**2**	0	0	**0**
			Total:	**16**	**0**	**11**	**21.5**

Total Contact Hours: 27 **Total Credits: 21.5**

VI SEMESTER (THIRD YEAR)

S.No.	Code	Name of the Course		L	T	P	C
1	V20CST13	Computer Networks	PCC	3	0	0	3
2	V20CST14	Machine Learning	PCC	3	0	0	3
3	V20CST15	Automata and Compiler Design	PCC	3	0	0	3
4		**Open Elective -II / Job Oriented Elective-II**	OEC	3	0	0	3
			JOE	0	0	6	
5	**Professional Elective-II**		PEC	3	0	0	3
	V20CSTPE05	Object Oriented Software Engineering					
	V20CSTPE06	Advanced Data Structures					
	V20CSTPE07	Data Science					
	V20CSTPE08	Cryptography &Network Security					
6	V20CSL11	Computer Networks Lab	PCC	0	0	3	1.5
7	V20CSL12	Machine Learning Lab using Python	PCC	0	0	3	1.5
8	V20CSL13	Unified Modeling Language Lab	PCC	0	0	3	1.5
9	V20SOC04	Skill Oriented Course-IV	SO	**1**	**0**	**2**	**2**
10		Professional Ethics & Human Values	MNC	**2**	0	0	**0**
			Total:	**15**	**0**	**17**	**21.5**

Total Contact Hours: 32 **Total Credits: 21.5**

VII SEMESTER(FOURTH YEAR)

S.No.	Code	Name of the Course		L	T	P	C
1	**Professional Elective-III**						
	V20CSTPE09	Advanced Computer Architecture					
	V20CSTPE10	BigData Analytics	PEC	3	0	0	3
	V20CSTPE11	Deep Learning					
	V20CSTPE12	Human Computer Interaction					
2	**Professional Elective-IV**						
	V20CSTPE13	Design Patterns					
	V20CSTPE14	NoSQL Databases	PEC	3	0	0	3
	V20CSTPE15	Reinforcement Learning					
	V20CSTPE16	Cloud Computing					
3	**Professional Elective-V**						
	V20CSTPE17	Software Project Management					
	V20CSTPE18	Scripting Languages	PEC	3	0	0	3
	V20CSTPE19	Natural Language Processing					
	V20CSTPE20	Social Networks and Semantic Web					
4		**Open Elective -III / Job Oriented Elective -III**	OEC	3	0	0	3
			JOE	0	0	6	
5		**Open Elective -IV / Job Oriented Elective - IV**	OEC	3	0	0	3
			JOE	3	0	0	
6		Management Science	HSS	3	0	0	3
7	V20SOC05	Skill Oriented Course-V	SO	1	0	2	2
8	V20CSP02	Mini Project /Internship	Internship	0	0	6	3
			Total:	**16**	0	14	**23**

Total Contact Hours: 30 **Total Credits: 23**

VIII SEMESTER (FOURTH YEAR)

S.No.	Code	Name of the Course		L	T	P	C
1	V20CSP03	Internship/ Industrial Training /Practical training	PRO	0	0	4	2
2	V20CSP04	Major Project (6 Months)	PRO	0	0	20	10
			Total:	0	0	**24**	**12**

Total Contact Hours: 24 **Total Credits: 12**

SKILL ORIENTED COURSES

With reference to the Lr No: SVEC/Admn/Circular/2021-22/61 dated: 05/11/2021, regarding the V20 regulations in B.Tech III & IV Semesters about the offering of Skill Oriented Course in all branches, the course codes are as follows.

In III Semester: V20SOC01
In IV Semester: V20SOC02

If any proposed courses and course codes offered under Skill Oriented Course will be considered as Skill Oriented Course **(V20SOC01 or V20SOC02)** only.

S.No.	Name of the Course
1.	Mobile Application Development
2.	MEAN Stack Technologies
3.	Secure DevOps
4.	AWS Cloud Computing
5.	SDG -Web Development
6.	Web Development using Django
7.	Game Development using Buildbox
8.	Game Programming
9.	.NET Framework
10.	CCNA IT Essentials
11.	Augmented Reality and Virtual Reality
12.	Go Programming
13.	Applications of Python using NumPy & Pandas
14.	Ethical Hacking
Any advanced courses offered by industries / Professional bodies / APSSDC can be appended in future	

Annexure-II

Semester	**V**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	3	0	0	3	V20CST10
Name of the Course	**Operating Systems**					
Branch	Common to CSE and CST					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Describe Operating System Services and System Calls. **(K2)**
CO2: Illustrate Process Management Concepts and CPU Scheduling Algorithms. **(K3)**
CO3: Demonstrate Process Synchronization primitives and Process Deadlocks. **(K3)**
CO4: Illustrate Memory Management Techniques and Page Replacement Algorithms. **(K3)**
CO5: Describe File System Concepts and Mass Storage Structures. **(K2)**

UNIT-I: Introduction: Operating-System Structure, Operating-System Services, User and Operating System Interface, System Calls, Types of System Calls.

UNIT-II: Process Management: Process Concept, Process Scheduling, Operations on Processes, Inter process Communication. **Threads:** Overview, Multithreading Models
CPU Scheduling: Basic Concepts**,** Scheduling Criteria**,** Scheduling Algorithms.

UNIT-III: Process Synchronization: The Critical-Section Problem, Peterson's Solution, Synchronization Hardware, Mutex Locks, Semaphores, Classic Problems of Synchronization, Monitors. **Deadlocks:** System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock.

UNIT-IV: Memory Management: Main Memory: Swapping, Contiguous Memory Allocation, Segmentation, Paging, Structure of the Page Table.
Virtual Memory: Introduction, Demand Paging, Page Replacement Algorithms, Allocation of Frames, Thrashing.

UNIT-V: Storage Management: Overview of Mass-Storage Structure, Disk Scheduling, File Concept, Access Methods, Directory and Disk Structure, File-System Mounting, File Allocation Methods.

Text Book:

1. Operating System Concepts**,** Abraham Silberschatz, ,Peter Baer Galvin,Greg Gagne, 9th Edition, John Wiley and Sons Inc., 2012.

Reference Books:

1. Operating Systems – Internals and Design Principles, William Stallings, 7th Edition, Prentice Hall, 2012 .
2. Modern Operating Systems, Andrew S. Tanenbaum, Third Edition, Addison Wesley, 2007.

Semester	**V**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	3	0	0	3	V20CST11
Name of the Course	**Data Mining**					
Branch	Common to CSE and CST					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Explain the concept of Data Mining and its functionalities. **(K2)**
CO2: Discuss various Data Preprocessing Techniques. **(K3)**
CO3: Demonstrate Association Analysis Techniques. **(K3)**
CO4: Illustrate various Classification Techniques. **(K3)**
CO5: Use different Clustering techniques to cluster data. **(K3)**

UNIT-I: Introduction: Need for Data Mining, Knowledge Discovery from Data, Kinds of Data mined, Kinds of Patterns mined, Technologies used, Kinds of Applications targeted, Major Issues in Data Mining, Data Objects and Attribute Types, Basic Statistical Descriptions of Data, Measuring Data Similarity and Dissimilarity.

UNIT-II: Data Preprocessing: Overview of Data Preprocessing, Data Cleaning, Data Integration, Data Reduction, Data Transformation and Data Discretization.

UNIT-III: Mining Frequent Patterns, Associations, and Correlations: Basic Concepts, Frequent Itemset Mining Methods- Apriori Algorithm: Finding Frequent Itemsets by Confined Candidate Generation, Generating Association Rules from Frequent Itemsets, Improving the Efficiency of Apriori, Pattern-Growth Approach for Mining Frequent Itemsets.

UNIT-IV: Classification: Basic Concepts, Decision Tree Induction, Attribute Selection Measures, Tree Pruning.
Bayes Classification Methods: Bayes' Theorem, Naive Bayesian Classification.
Bayesian Belief Networks: Concepts and Mechanisms.

UNIT-V: Cluster Analysis: Basic Concepts and Methods, Partitioning Methods, Hierarchical Methods, Density Based Method-DBSCAN.

Text Books:
1. Data Mining Concepts and Techniques, Jiawei Han, Micheline Kamber, Jian Pei,3rd Edition, Morgan Kaufmann Publishers.

.

Reference Books:
1. Introduction to Data Mining, Pang-Ning Tan, Michael Steinbach, Vipin Kumar, 1st Edition, Pearson Education Inc.
2. Data Mining and Analysis,Mohammed J Zaki, Wagner Meira JR, 1st Edition ,Cambridge University Press.

Semester	**V**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	3	0	0	3	V20CST12
Name of the Course	**Web Technologies**					
Branch	Common to CSE and CST					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Illustrate the basic concepts of HTML and CSS. **(K2)**
CO2: Illustrate Extensible markup language and XML parsers **.** **(K3)**
CO3: Develop web applications using JDBC . **(K3)**
CO4: Build database driven web applications using JSP. **(K3)**
CO5: Illustrate the basic concepts of Angular and NODE JS. **(K2)**

UNIT-I: HTML :Basic Syntax, Standard HTML Document Structure, Basic Text Markup, Images, Hypertext Links, Lists, Tables, Frames Forms.
CSS: Cascading style sheets, Levels of Style Sheets, Style Specification Formats, Selector Forms, Property value forms, Font Properties, List Properties, color, Alignment of Text

UNIT-II: Working with XML: Introduction, The syntax of XML, XML Document Structure, Document type Definition (DTD), Namespaces, XML schemas, XSLT, **XML Parsers** - DOM and SAX

UNIT-III: WORKING WITH DATABASE: Getting started with JDBC , Defining ODBC, Introduction to JDBC, Components of JDBC, JDBC Architecture, Types of Drivers, Working with JDBC APIs, Creating a Simple Application, Working with Prepared Statement.

UNIT IV: Introduction to Servlets & JSP: Introduction to servlets, Life cycle of Servlet, Limitations of servlets, Java Server Pages: JSP Overview, Components of a JSP Page: Directives, comments, Expressions, Scriptlets , Declarations, implicit objects, Database Access, session tracking.

UNIT V: Fundamentals of NODE JS and Angular : Understanding Node.js, Installing Node.js, Working with Node Packages, Creating a Node.js Application, Understanding Angular, Modules, Directives, Data Binding, Dependency Injection, Services, Creating a Basic Angular Application.

Text Books:

1. Programming the World Wide Web, 7th Edition, Robet W Sebesta, Pearson, 2013.
2. Node.js, MongoDB and Angular Web Development, 2nd Edition,Brad Dayley Brendan Dayley Caleb Dayley,Pearson Education,2018
3. JSP:The Complete reference,Phil Hanna,The McGraw-Hill Copanies,2001.
4. JDBC, Servlets, and JSP, New Edition, Santhosh Kumar K, Kogent Learning Solutions Inc, Dreamtech Press, 2018.

Reference Books:

1. Web Technologies: HTML, JavaScript, PHP, Java, JSP, XML and AJAX, Black book, Dream Tech.
2. An Introduction to Web Design, Programming, Paul S Wang, Sanda S Katila, Cengage Learning
3. Web Technologies, 1st Edition 7th impression, Uttam K Roy, Oxford, 2012.

Semester	**V**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	3	0	0	3	V20CSTPE01
Name of the Course	**Software Testing Methodologies (Professional Elective-I)**					
Branch	Common to CSE and CST					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Describe Software testing objectives and methodology. **(K2)**
CO2: Apply various Software testing techniques. **(K3)**
CO3: Discuss Static testing techniques for software testing. **(K2)**
CO4: Distinguish Software testing and debugging process. **(K2)**
CO5: Explain modern Software testing tools to Support software testing. **(K2)**

UNIT-I: Introduction to Software Testing: Evolution of software Testing, Myths and Facts, Goals of software Testing, Definitions of Testing, Model for Software Testing, Software Testing Terminology, Software Testing Life Cycle.

UNIT-II: Verification and Validation: Verification & Validation Activities, Verification, Verification of Requirements, Verification of High level and low level designs, How to verify code, Validation. **Dynamic Testing I:** Black Box testing techniques: Boundary Value Analysis, Equivalence Class Testing, Decision Table based Testing,

UNIT-III: Dynamic Testing II: White-Box Testing: Need of White-Box Testing, Logic coverage criteria, Basis path testing, Loop testing. **Static Testing:** Inspections, Structured Walkthroughs, Technical reviews.

UNIT-VI: Regression Testing: Progressive Vs Regressive Testing, Regression testability, Objectives of regression testing, When is Regression Testing done? Regression Testing Types, Regression testing techniques. **Debugging:** Debugging process, Techniques, correcting bugs.

UNIT-V: Software Quality Management: Software quality concept, Quality control and Quality Assurance, Software Quality metrics. **Automation and Testing Tools:** Need for automation, categorization of Testing tools, selection of testing tools, Overview of some commercial testing tools.

Text Books:

1. Software Testing, Principles and Practices, Naresh Chauhan, 9th Edition, Oxford Publisher.

Reference Books:

1. Software testing techniques - Boris Beizer, 2nd Edition, Dreamtech publisher.
2. Foundations of Software testing, Aditya P Mathur, 2nd ed, Pearson.
3. Software Testing- Yogesh Singh, CAMBRIDGE.

Semester	**V**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	3	0	0	3	V20CSTPE02
Name of the Course	**Principles of Programming Languages (Professional Elective-I)**					
Branch	Common to CSE and CST					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Describe syntax and semantics of programming languages. **(K2)**
CO2: Explain data types and basic statements of programming languages **(K2)**
CO3: Design and implement subprogram constructs **(K3)**
CO4: Discussconcurrency process using OOP. **(K2)**
CO5: Develop programs in Scheme, ML, and Prolog. **(K3)**

UNIT-I:Syntax and semantics: Evolution of programming languages, describing syntax, context, free grammars, attribute grammars, describing semantics, lexical analysis, parsing, recursive – decentbottom - up parsing

UNIT-II:Data, Data types, and basic statements: Names, variables, binding, type checking, scope, scoperules, lifetime and garbage collection, primitive data types, strings, array types, associativearrays, pointers and references, Arithmetic expressions, overloadedoperators, type conversions, relational and boolean expressions , assignment statements , mixedmode assignments, control structures – selection, iterations, branching, guarded Statements

UNIT-III:Subprograms and implementations: Subprograms, design issues, local referencing, parameterpassing, overloaded methods, generic methods, design issues for functions, semantics of call andreturn, implementing simple subprograms, stack and dynamic local variables, nestedsubprograms, blocks, dynamic scoping.

UNIT- IV: Object- orientation, concurrency, and event handling: Object – orientation, design issues for OOP languages, implementation of object, oriented constructs, concurrency, semaphores, Monitors, message passing, threads, statement level concurrency, exception handling.

UNIT- V:Functional programming languages: Introduction to lambda calculus, fundamentals of functional programming languages, Programming with Scheme, Programming with ML, Logic programming languages: Introduction to logic and logic programming, Programming with Prolog, multi - paradigm languages

Text Books:

1. Robert W. Sebesta, “Concepts of Programming Languages”, Tenth Edition, Addison Wesley, 2012.
2. Programming Languages, Principles & Paradigms, 2ed, Allen B Tucker, Robert E Noonan, TMH

Reference Books:

1. R. Kent Dybvig, “The Scheme programming language”, Fourth Edition, MIT Press, 2009.
2. Jeffrey D. Ullman, “Elements of ML programming”, Second Edition, Prentice Hall, 1998.
3. Richard A. O'Keefe, “The craft of Prolog”, MIT Press, 2009.
4. W. F. Clocksin and C. S. Mellish, “Programming in Prolog: Using the ISO Standard”,5th Edition, Springer, 2003

Semester	**V**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	3	0	0	3	V20CSTPE03
Name of the Course	**Artificial Intelligence (Professional Elective-I)**					
Branch	Common to CSE & CST					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Discuss the foundations of AI. **(K2)**
CO2: Identify Search Strategies for Problem Solving. **(K2)**
CO3: Illustrate Adversarial Search for Game Playing. **(K2)**
CO4: Discuss Reasoning approaches. **(K2)**
CO5: Illustrate Knowledge Representation approaches. **(K2)**

UNIT-I: Introduction: What is AI? The Foundations of Artificial Intelligence, History of Artificial Intelligence, The State of the Art Applications.
Intelligent Agents: Agents and Environments, Good Behaviour: The Concept of Rationality, The Nature of Environments, The Structure of Agents.

UNIT-II: Solving Problems by Searching: Problem-Solving Agents, Example Problems, Searching for Solutions, Uninformed Search Strategies, Informed (Heuristic) Search Strategies, Heuristic Functions, Local Search Algorithms and Optimization Problems.

UNIT-III: Adversarial Search : Games, Optimal Decisions in Games, Alpha–Beta Pruning, Imperfect Real-Time Decisions, Stochastic Games, Partially Observable Games, State-of-the-Art Game Programs, Alternative Approaches.

UNIT-IV: Knowledge and Reasoning: Propositional Logic, Propositional Theorem Proving, Syntax and Semantics of First-Order Logic, Using First-Order Logic, Forward Chaining, Backward Chaining, Resolution.

UNIT-V: Knowledge Representation: Representations and Mappings, Approaches to Knowledge Representation-Simple Relational Knowledge, Inheritable Knowledge, Inferential Knowledge, Procedural Knowledge, Issues in Knowledge Representation, The Frame Problem.

Text Books:

1. Artificial Intelligence : A Modern Approach, Stuart J. Russell and Peter Norvig, 3rd Edition, Prentice Hall.
2. Artificial Intelligence, Elaine Rich,Kevin Knight, Shivashankar B Nair, 3rd Edition,Tata McGraw-Hill.

Reference Books:

1. Artificial Intelligence, George F Luger, Pearson Education Publications.
2. Artificial Intelligence, Saroj Kaushik, 1st Edition, Cengage Learning.

Semester	**V**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	3	0	0	3	V20CSTPE04
Name of the Course	**Computer Graphics (Professional Elective-I)**					
Branch	Common to CSE and CST					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Discuss the applications of computer graphics and learn basic algorithms. **(K2)**
CO2: Discuss the concepts of 2D graphics along with transformation techniques. **(K2)**
CO3: Demonstrate3D graphics and 3D object representation. **(K3)**
CO4: Discuss different visible surface detection methods and color models. **(K2)**
CO5: Illustrate different animation sequences . **(K2)**

UNIT-I: Introduction: Application of Computer Graphics, raster scan systems, random scan systems, raster scan display processors. Output Primitives : Points and lines, line drawing algorithms(Bresenham's and DDA Line derivations and algorithms), mid-point circle algorithms. **Filled area primitives:** Boundary-fill and flood-fill algorithms.

UNIT-II: 2-D geometrical transforms: Translation, scaling, rotation, reflection and shear transformations, and homogeneous coordinates, composite transforms.**2-D viewing:** The viewing pipeline, viewing coordinate reference frame, window to view-port coordinate transformation, viewing functions, Cohen-Sutherland line clipping, Sutherland–Hodgeman polygon clipping algorithm.

UNIT-III: 3-D Geometric transformations: Translation, rotation, scaling, reflection and shear transformations, composite transformations. **3-D object representation:** Polygon surfaces, quadric surfaces, spline representation, Bezier curve and B-Spline curves.

UNIT-IV: Visible surface detection methods: Classification, back-face detection, depth-buffer, scan-line, BSP tree methods, area sub-division. **Color Models** – RGB, YIQ, CMY, HSV.

UNIT-V: Computer Animation: Design of animation sequence, general computer animation functions, raster animation, computer animation languages, key frame systems**,** motion specifications.

Text Books:

1. Computer Graphics C version, Donald Hearn, M.Pauline Baker,Pearson
2. Computer Graphics, Schaum's outlines", Zhigandxiang,RoyPlastock, 2nd Edition,TataMc-Graw HillEdition.

Reference Books:

1. Computer Graphics Principles & practice, 2/e, Foley, VanDam, Feiner, Hughes,Pearson
2. Computer Graphics, Peter, Shirley, CENGAGE
3. Principles of Interactive Computer Graphics, Neuman ,Sproul,TMH.

Semester	**V**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	0	0	3	1.5	V20CSL09
Name of the Course	**Data Mining Lab**					
Branch	Common to CSE and CST					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Demonstrate Data Preprocessing techniques. **(K3)**
CO2: Demonstrate Association Rule Mining techniques. **(K3)**
CO3: Demonstrate Classification techniques. **(K3)**
CO4: Demonstrate the Clustering techniques. **(K3)**

List of Experiments (Weka Tool)

1. Demonstrate Data Preprocessing on predefined Weka dataset labor.arff
2. Create a student.arff dataset and Demonstrate Data Preprocessing on it
3. Demonstrate Association rule process on predefined Weka dataset contactlenses.arff using apriori algorithm.
4. Create an employee.arff dataset and demonstrate Association rule process on it using apriori algorithm
5. Demonstrate Classification process on student.arff dataset using j48 algorithm
6. Create a customer.arff dataset and demonstrate Classification process on it using j48 algorithm
7. Demonstrate Classification process on employee.arff dataset using id3 algorithm
8. Demonstrate Classification process on employee.arff dataset using Naïve Bayes algorithm
9. Demonstrate Clustering process on predefined Weka dataset iris.arff using simple k-means algorithm.
10. Demonstrate Clustering process on dataset student.arff using simple k- means algorithm.

Reference Books:

1. Data Mining: Practical Machine Learning Tools and Techniques, Ian H. Witten, Eibe Frank, Mark A. Hall, 3rd Edition, Morgan Kaufmann Publishers.
2. Data Mining Concepts and Techniques, Jiawei Han, Micheline Kamber, Jian Pei, 3rd Edition, Morgan Kaufmann Publishers.
3. Introduction to Data Mining, Pang-Ning Tan, Michael Steinbach, Vipin Kumar, 1st Edition, Pearson Education Inc.

Semester	V	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	0	0	3	1.5	V20CSL10
Name of the Course	**Web Technologies Lab**					
Branch	Common to CSE and CST					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Illustrate the basic concepts of HTML and CSS. **(K2)**
CO2: Illustrate Extensible markup language and XML parsers . **(K3)**
CO3: Develop web applications using JDBC . **(K3)**
CO4: Build database driven web applications using JSP. **(K3)**
CO5: Illustrate the basic concepts of Angular and NODE JS. **(K2)**

List of Experiments

Exercise 1: Design HTML fundamental constructs.
(i) Headings (ii) Links (iii) Paragraph (iv) Images (v) Tables

Exercise 2: Design HTML fundamental constructs.
(i) Frames (ii) Forms and HTML controls

Exercise 3: Design Cascading style sheets
(i) Internal (ii) External (iii) Inline

Exercise 5: Write an XML file which will display the Book information which includes the following:
(i) Title of the book (ii) Author Name (iii) ISBN number (iv) Publisher name
(v) Edition (vi) Price
(a) Write a Document Type Definition (DTD) to validate the above XML file.
(b) Write a XML Schema Definition (XSD)

Exercise 6: Create a simple JSP to print the current Date and Time.

Exercise 7: Develop JSP program calculates factorial values for an integer number, while the input is taken from an HTML form.

Exercise 8: Develop JSP program shows a Sample Order Form.

A Sample Order Form

Item	Price	Quantity	Total Price
DVD	19.99	2	39.98
CD	12.99	9	116.91
Diskette	1.99	24	47.76

Exercise 9: Create JSP to insert, delete, and update the details of student into the database using JDBC connectivity.

Exercise 10: Design a simple Angular JS form

Exercise 11: Design a simple Node JS application

Reference Books:

1. Web Technologies: HTML, JavaScript, PHP, Java, JSP, XML and AJAX, Black book, Dream Tech.
2. An Introduction to Web Design, Programming, Paul S Wang, Sanda S Katila, Cengage Learning
3. Web Technologies, 1st Edition 7th impression, Uttam K Roy, Oxford, 2012.

Semester	**VI**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	3	0	0	3	V20CST13
Name of the Course	**Computer Networks**					
Branch	Common to CSE and CST					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Discuss fundamentals of network concepts and Reference Models. **(K2)**
CO2: Discuss Communication media and switching techniques. **(K2)**
CO3: Demonstrate Error control and Data link layer protocols. **(K3)**
CO4: Apply Routing algorithms and congestion control algorithms. **(K3)**
CO5: Discuss Transport layer protocols and Application layer protocols. **(K2)**

UNIT-I: Introduction: Reference models: The OSI Reference Model- the TCP/IP Reference Model, Examples of Networks: Novell Networks, Arpanet, Internet, Network Topologies WAN, LAN, MAN.

UNIT-II: Physical Layer: Transmission Media, Multiplexing: FDM, WDM and TDM- LAN Technologies, introduction to switching: Circuit Switched Networks, Datagram Networks, and Virtual Circuit Networks.

UNIT–III: Data link layer: Design issues, Framing, Flow control, error control, error detection - Parity bit, CRC, Checksum, error correction- Hamming code. MAC: ALOHA, CSMA. Elementary Data Link Layer protocols: simplex protocol, Simplex stop and wait, Simplex protocol for Noisy Channel. Sliding window protocol: One bit, Go back N, Selective repeat-Stop and wait protocol, HDLC, point to point protocol (PPP).Piggybacking.

UNIT-IV: Network Layer : Network layer design issues- Algorithm shortest path routing, Flooding, Distance Vector Routing, Link State Routing, Hierarchical routing, Broad cast, Multi cast Routing algorithms-Congestion control and algorithms, Internet Protocol (IP) Addresses, Subnet masking. Quality of Service, QoS improving techniques: Leaky Bucket and Token Bucket algorithm.

UNIT–V:Transport Layer: Services, Primitives and sockets, Elements of transport protocols, Internet Transport protocols(TCP,UDP,RPC,RTTP/RTP,RTCP) Segment headers, Primitives, Control, Congestion control.
Application layer: DNS, SMTP, POP,FTP HTTP Presentation formatting. Network security: Cryptography, DES Public key and RSA private key cryptography Algorithms.

Text Books:

1. Computer Networks — Andrew S Tanenbaum, 4th Edition. Pearson Education/PHI
2. Data Communications and Networks – Behrouz A. Forouzan.Third Edition TMH

Reference Books:

1. An Engineering Approach to Computer Networks-S.Keshav, 2nd Edition, Pearson Education
2. Understanding communications and Networks, 3rd Edition, W.A. Shay, Thomson

Semester	VI	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	3	0	0	3	V20CST14
Name of the Course	**Machine Learning**					
Branch	Common to CSE and CST					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Explain the Basics of Machine Learning **(K2)**

CO2: Demonstrate Classification and Clustering Techniques. **(K3)**

CO3: Construct Decision Trees and Random Forest. **(K3)**

CO4: Illustrate the Working of Neuron and Perceptron Algorithm. **(K2)**

CO5: Demonstrate the working of Multi-Layer Perceptron algorithm. **(K3)**

UNIT-I: Introduction: Learning**:** Machine Learning, Types of Machine Learning, Supervised Learning: Classification & Regression, The Machine Learning Process, Weight Space, The Curse of Dimensionality, Overfitting, Training, Testing, Validation Sets, The Confusion Matrix, Accuracy, ROC Curve, Unbalanced Datasets, Measurement Precision, Thae Bias-Variance Tradeoff.

UNIT-II: Classification: The General Problem, Probabilistic Classifiers: The Bayes Classifier, Logistic Regression, Non-Probabilistic Classifiers: K-Nearest Neighbors, Support Vector Machines, Assessing Classification Performance: Accuracy-0/1 Loss, Sensitivity, Specificity. **Clustering**: The General Problem, K-Means Clustering.

UNIT-III: Learning With Trees: Using Decision Trees, Constructing Decision Trees: Entropy in Information Theory, ID3 Algorithm, CART-Gini Impurity. **Ensemble Learning:** Boosting: Adaboost, Stumping; Bagging, Random Forests.

UNIT-IV: Neuron & Neural Network: The Brain And The Neuron: Hebb's Rule, McCulloch and Pitts Neuron and Its Limitations, Neural Networks, The Perceptron: The Learning Rate, Bias Input, The Perceptron Learning Algorithm, Linear Separability, Linear Regression.

UNIT-V: Multi-Layer Perceptron: Going Forward: Biases, Back-Propagation and Error, The Multi-Layer Perceptron Algorithm, Initializing the Weights, Activation Functions, Sequential and Batch Training, Local Minima, Picking up momentum, Minibatches and Stochastic Gradient Descent, The Multi-Layer Perceptron In Practice: Amount of Training Data, Number of Hidden Layers, When to Stop Learning.

Text Books:

1. Machine Learning: An Algorithmic Perspective, Stephen Marsland, 2nd Edition, CRC Press.
2. A First Course in Machine Learning, Simon Rogers & Mark Girolami, 2nd Edition, CRC Press.

Reference Books:

1. Machine Learning, Tom Mirchel,Mcgraw Hill.
2. Machine Learning: The Art and Science of Algorithms that Make Sense of Data, Peter Flash, Cambridge University Press.

Semester	VI	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	3	0	0	3	V20CST15
Name of the Course	**Automata and Compiler Design**					
Branch	Common to CSE and CST					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Construct Finite Automata and Regular Expressions. **(K3)**
CO2: Describe the Compilation Process and Lexical Analysis. **(K2)**
CO3: Construct Topdown and Bottom up Parsing Techniques. **(K3)**
CO4: Produce Intermediate Code Generation and Runtime Environments**.** **(K3)**
CO5: Explain Code Optimization and Code Generation. **(K2)**

UNIT I: Formal Language and Regular Expressions: Alphabet, Strings, Language, Finite Automaton-Design of DFA, Design of NFA, Equivalence between NFA and DFA, Finite Automata with ϵ-Transition, Equivalence between NFA and ϵ-NFA. **Regular Expression**: Regular expressions Equivalence between Regular Expressions and Finite Automata ,Chomsky Hierarchy.

UNIT II: Compiler: Definition, Structure of a compiler. **Lexical Analysis**: The Role of the Lexical Analyzer, Specification of Tokens, Recognition of Tokens and the Lexical-Analyzer Generator-Lex. **Context Free grammars**: Context free grammars, derivation, parse trees, Ambiguous Grammar, Writing a Grammar- Elimination of Left Recursion, Left Factoring.

UNIT III: Top Down Parsing: First and Follow, LL(1) Grammars, **Bottom-Up Parsing**: Bottom Up Parser Classification, Reductions, Handle Pruning, Shift-Reducing, Constructing SLR Parsing Tables, construction of CLR (1), LALR Parsing tables, Comparison of all Bottom Up approaches.

UNIT IV: Semantic Analysis: Syntax Directed Definitions, Evaluation Orders for SDD‘s
Intermediate Code Generation: Variants of Syntax Trees, Three-Address Code, Basic blocks and Flow graphs, Control Flow. **Run-Time Environments:** Storage Organization, Stack Allocation of Space, Heap Management

UNIT V: Code optimization: Machine Independent Optimization. The principle sources of Optimization, optimization of Basic blocks, peep hole Optimization, Introduction to Date flow Analysis.
Code generation: Issues in design of code generation, The target Language, Address in the target code, A Simple Code generation.

Text Books:

1) Introduction to AutomataTheory, Languages and Computation, J.E.Hopcroft, R.MotwaniandJ.D.Ullman, 3rdEdition, Pearson, 2008.
2) Compilers Principles, Techniques and Tools Aho, Ullman, Ravisethi, Pearson Education.

Reference Books:

1) Louden: “Compiler Construction, Principles & Practice”, 1st Edition, Thomson Press, 2006.
2) Tremblay J P, Sorenson G P: “The Theory & Practice of Compiler writing”, 1st Edition, BSP Publication, 2010.
3) Theory of Computation, V. Kulkarni, Oxford University Press, 2013.

Semester	**VI**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	3	0	0	3	V20CSTPE05
Name of the Course	**Object Oriented Software Engineering (Professional Elective-II)**					
Branch	Common to CSE and CST					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Describe Software process and different life cycle models. **(K2)**
CO2: Discuss Project Planning, and organization. **(K2)**
CO3: Apply OO concepts along with their applicability contexts. **(K3)**
CO4: Demonstrate object oriented analysis and design. **(K3)**
CO5: Describe Implementation, Integration and Maintenance phases. **(K2)**

UNIT I: Introduction to Classical software Engineering: Introduction to OO Paradigm.Different phases in structured paradigm and OO Paradigm. Software Process and different life cycle models and corresponding strengths and weaknesses.

UNIT II: Planning and Estimation: Estimation of Duration and Cost, COCOMO components of software. Project Management plan. Planning Object-Oriented Projects.Project Organization & communication concepts and their activities.

UNIT III: Modules to objects: Cohesion and Coupling, Data Encapsulation and Information hiding aspects of Objects. Inheritance, Polymorphism and Dynamic Binding aspects.Cohesion and coupling of objects. Reusability, Portability and Interoperability aspects.

Introduction to testing, with focus on Utility, Reliability, Robustness, Performance, Correctness.

UNIT IV: Requirement phase: Rapid Prototyping method, Specification phase, Specification Document, Formal methods of developing specification document, Examples of other semi - formal methods of using Finite-State- Machines, Petri nets and E- Language.

Analysis phase: Use case Modeling, Class Modeling, Dynamic Modeling, Testing during OO Analysis.

UNIT V: Design phase: Data oriented design, Object Oriented design, and Formal techniques for detailed design. Challenges in design phase.**IIM Phases:** Implementation, Integration and maintenance phases, OOSE aspects in these phases.

Text Books:

1. Object oriented and Classical Software Engineering, **7/e**, Stephen R. Schach, TMH
2. Object oriented and classical software Engineering, Timothy Lethbridge, Robert Laganiere, TMH, **Second Edition**.

Reference Books:

1. Component-based software engineering: 7th international symposium, **CBSE 2004**, IvicaCrnkovic, Springer.

Semester	**VI**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	3	0	0	3	V20CSTPE06
Name of the Course	**Advanced Data Structures (Professional Elective-II)**					
Branch	Common to CSE and CST					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Explain external sorting method. **(K2)**
CO2: Discuss pattern matching Algorithms. **(K2)**
CO3: Illustrate various hash functions with appropriate examples. **(K3)**
CO4: Illustrate various priority queues with appropriate examples. **(K3)**
CO5: Construct self-balanced tree with appropriate examples. **(K3)**

UNIT-I: Sorting: Introduction - External Sorting- K-way Merging - Buffer Handling for parallel Operation Run Generation- Optimal Merging of Runs.

UNIT-II: String Matching Algorithms: The Navi String matching algorithms – The Robin-Krap algorithm – String Matching algorithm using finite automata – The Knuth Morris Pratt algorithm.

UNIT-III: Hashing: Dictionaries –-Hash Table Representation: Ideal hashing – Hash functions and tables -Linear probing- Hashing with Chains
Priority Queues (HEAPS): Definition and Applications – ADT – Linear lists – Heaps : Definition – Max heap and Min heap operations, Applications – Heap Sort – Huffman Codes.

UNIT-IV:Efficient Binary Search Trees :ADT-Introduction to AVL Trees- Red-Black Trees-DefinitionRepresentation of a Red- Black Tree- Searching a Red-Black Tree- Inserting into a Red Black Tree- Deletion from a Red-Black Tree- Joining Red-Black Trees, Splitting a Red-Black tree – Splay Trees – Introduction – operation – Amortized complexity.

UNIT-V:Multiway Search Trees : ISAM - M-Way Search Trees, Definition and PropertiesSearching an M-Way Search Tree, B-Trees, Definition and Properties- search Elements in a B-treeInsertion into B-Tree- Deletion from a B-Tree- Node Structure.

Text Books:
1. Data Structures, Algorithms and Applications in C++; SartajSahni; UniverstiyPress ; 2 nd Edition.
2. Introduction to Algorithms By Thomas H Cormen, Charless E leiserson, Ronald L Rivest and Cliford Stein PHI publication Third Edition (UNIT – II)

References:
1. Data Structures, a Pseudocode Approach, Richard F Gilberg, BehrouzAForouzan, Cengage.
2. An Introduction to Data Structures with applications By Jean Paul Trembly and Paul G Sorenson Tata McGraw Hill Second Edition
3. Fundamentals of Data Structures and algorithms by C V Sastry, RakeshNayak, Ch. Raja Ramesh, IK Publications, new Delhi.

Semester	**VI**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	3	0	0	3	V20CSTPE07
Name of the Course	**Data Science (Professional Elective-II)**					
Branch	Common to CSE and CST					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Discuss the fundamental concepts of Data Science. **(K2)**
CO2:Illustrate Exploratory Data Analysis. **(K2)**
CO3: Explain the Concepts of Recommendation Engines. **(K2)**
CO4: Explain various Anomaly Detection Techniques. **(K2)**
CO5: Discuss Feature Selection techniques. **(K2)**

UNIT-I: Introduction:AI, Machine Learning and Data Science, What is Data Science? Case for Data Science, Data Science Classification, Data Science Algorithms.
Data Science Process: Prior Knowledge, Data Preparation, Modeling-Training and Testing Datasets,Learning Algorithms,Evaluation of the Model, Ensemble Modeling, Application, Knowledge.

UNIT-II: Data Exploration: Objectives of Data Exploration, Datasets- Types of Data, Descriptive Statistics-Univariate Exploration, Multivariate Exploration, Data Visualization, Roadmap for Data Exploration.

UNIT-III: Recommendation Engines: Need, Applications, Concepts, Types, Collaborative Filtering-Neighborhood-Based Methods, Matrix Factorization; Content-Based Filtering- Building an Item Profile, User Profile Computation, Implementation Steps,Hybrid Recommenders.

UNIT-IV: Anomaly Detection: Concepts - Causes of Outliers, Anomaly Detection Techniques; Distance-Based Outlier Detection- Working, Implementation Steps; Density-Based Outlier Detection-Working, Implementation Steps; Local Outlier Factor- Working, Implementation Steps.

UNIT-V: Feature Selection: Classifying Feature Selection Methods, Principal Component Analysis, Information Theory-Based Filtering, Chi-Square-Based Filtering, Wrapper-Type Feature Selection-Backward Elimination.

Textbook:

1. Data Science Concepts and Practice, Vijay Kotu, BalaDeshpande, 2nd Edition, Morgan Kaufmann Publishers.

Reference Books:

1. An Introduction to Data Science, Jeffrey S. Saltz,Jeffrey M. Stanton, Sage Publications.
2. The Art of Data Science, Roger D Peng, Elizabeth Matsui, Lean Publishing.
3. Data Science for Business, Foster Provost, Tom Fawcett,O'Reilly Media.

Semester	**VII**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	3	0	0	3	V20CSTPE08
Name of the Course	**Cryptography and Network Security (Professional Elective-II)**					
Branch	Common to CSE and CST					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Discuss fundamentals and mathematical support of Cryptography and Network Security. (**K2)**

CO2: Discuss symmetric and asymmetric cryptosystems. (**K2)**

CO3 : Discuss about HASH functions & Digital Signatures to provide authentication and integrity. **(K2)**

CO4: Demonstrate various methods of Mutual trust and mail security**. (K3)**

CO5: Review the Network& Internet Security Scenarios. **(K2)**

UNIT-I: Overview: Security attacks, Services, Mechanisms, A model for network security, Symmetric cipher model. **Classical encryption techniques**: Substitution Techniques, Transposition Techniques. **Number Theory**: Prime numbers, Fermat's theorem, Euler's Theorem, the Chinese Remainder Theorem.

UNIT-II: Block Cipher: Principles, DES, Strength of DES, AES, Block cipher Modes of Operations. **Public Key Cryptography:** Principles, Public Key Crypto system, RSA Algorithm, Diffie Hellman Key Exchange.

UNIT-III: Cryptographic Hash Functions: Application of Cryptographic Hash Functions, Requirements & Security, SHA-512, Message Authentication Functions, Requirements, HMAC. **Digital Signatures**: Properties, Attacks and Forgeries, Requirements, Digital Signature Standards, NIST Digital Signature Algorithm.

UNIT-IV: Key Management and Distribution: Symmetric Key Distribution Using Symmetric Encryption, Asymmetric Key Distribution Using Symmetric Encryption, Distribution of Public Keys, X.509 Certificates. **User Authentication**: Remote User Authentication Principles, Kerberos. **Electronic Mail Security:** Pretty Good Privacy (PGP) And S/MIME.

UNIT-V: IP Security: Two modes, two security protocols Authentication Header, Encapsulating Security Payload. **Transport Level Security:** Secure Socket Layer (SSL) and Transport Layer Security (TLS). **HTTPS:** Connection Initiation Connection Closure.

Text Books:

1. William Stallings, "Cryptography and Network Security, Principles and Practices", Pearson Education, Sixth Edition.
2. Cryptography and Network Security, Behrouz A Forouzan, Debdeep Mukhopadhyay, (3e) Mc Graw Hill.

Reference Books:

1. Charlie Kaufman, Radia Perlman and Mike Speciner, "Network Security – Private Communication in a Public World" Pearson/PHI.

Semester	**VI**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	0	0	3	1.5	V20CSL11
Name of the Course	**Computer Networks Lab**					
Branch	Common to CSE and CST					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Implement Error detection technique and Sliding window protocol. **(K3)**
CO2: Implement Routing and congestion control Algorithms. **(K3)**
CO3: Implement socket programming. **(K3)**

List of Experiments

(Implement using C/C++/Java/Python)

1. Study of basic network commands and Network configuration commands.
   a) Ping
   b) Tracert / Traceroute
   c) Ipconfig / ifconfig
   d) Hostname
   e) Nslookup
   f) Netstat
2. Construct Detecting error using CRC-CCITT.
3. Implementation of Bit Stuffing
4. Implementation of Character Stuffing
5. Implementation of stop and wait protocol.
6. Implementation of Dijkstra's algorithm
7. Implementation Distance vector algorithm
8. Implementation of Congestion control using leaky bucket algorithms
9. Implementation using Socket TCP both client and server programs.
10. Implementation using Socket UDP both client and server programs

Text Books:

1. Computer Networks — Andrew S Tanenbaum, 4th Edition. Pearson Education/PHI.
2. Data Communications and Networks – Behrouz A. Forouzan.Third Edition TMH.

Reference Boks:

1. An Engineering Approach to Computer Networks-S.Keshav, 2nd Edition, Pearson Education.
2. Understanding communications and Networks, 3rd Edition, W.A. Shay, Thomson.

Semester	**VI**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	0	0	3	1.5	V20CSL12
Name of the Course	**Machine Learning Lab using Python**					
Branch	Common to CSE and CST					

Syllabus Details

Course Outcomes: After successful completion of the course, the student will be able to:

CO1: Identify various Python libraries used in Machine Learning. **(K2)**
CO2: Implement probabilistic classifiers using Python Programming. **(K3)**
CO3: Construct non-probabilistic classifiers using Python Programming. **(K3)**
CO4: Demonstrate the process of clustering using the K-Means algorithm. **(K3)**
CO5: Illustrate the working of a Multi-layer perceptron network. **(K3)**

List of Experiments

1. Introduction to required python libraries such as Numpy, Pandas, Scipy, Matplotliband Scikit-learn.
2. Import, preprocess, and split the datasets using scikit-learn.
3. Construct a classification model using the Bayes classifier using Python Programming.
4. Implement a Logistic Regression algorithm for binary classification using Python Programming.
5. Implement the KNN algorithm for classification and demonstrate the process of finding out optimal "K" value using Python Programming.
6. Construct an SVM classifier using python programming.
7. Demonstrate the process of the Decision Tree construction for classification problems using python programming.
8. Implement an Ensemble Learner using Random Forest Algorithm using python programming.
9. Implement an Ensemble Learner using Adaboost Algorithm using Python programming.
10. Demonstrate the working of Multi-layer perceptron with MLPClassifier() using Python programming.
11. Demonstrate the K-Means algorithm for the given data set using Python programming.

Text Books:

1. Introduction to Machine Learning with Python, Andreas C. Muller and Sarah Guido, First Edition, O'Reilly.

Reference Books:

1. Practical Machine Learning with Python, Dipanjan Sarkar, Raghav Bali and Tushar Sharma, First Edition, A Press.

Semester	**VI**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	0	0	3	1.5	V20CSL13
Name of the Course	**Unified Modeling Language Lab**					
Branch	Common to CSE and CST					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Develop Class diagrams. **(K3)**
CO2: Develop Use case diagrams. **(K3)**
CO3: Construct Interaction diagrams. **(K3)**
CO4: Develop State chart, Activity diagrams. **(K3)**
CO5: Develop Component and Deployment diagrams. **(K3)**

List of Experiments

1. Draw basic class diagrams to identify and describe key concepts like classes, and their relationships.
2. Draw Use Case diagrams for capturing and representing requirements of the system.
3. Draw sequence diagrams OR communication diagrams with advanced notation for system to show objects and their message exchanges.
4. Draw activity diagrams to display either business flows or like flow charts.
5. Develop State chart diagrams.
6. Draw component diagrams assuming that build the system reusing existing components along with a few new ones.
7. Draw deployment diagrams to model the runtime architecture of system.
8. Design Case study on Library Management System.
9. Design Case Study on Hospital Management System.
10. Design Case study-Railway Reservation System.

Text Books:

1. The Unified Modeling Language User Guide, Grady Booch, James Rumbaugh, Ivar Jacobson, Pearson Education.

.

Reference Books:

1. UML 2 Toolkit, Hans-Erik Eriksson, Magnus Penker, Brian Lyons, David Fado, WILEY-Dreamtech India Pvt. Ltd.
2. Fundamentals of Object Oriented Design in UML, Meilir Page-Jones, Pearson Education.
3. Modeling Software Systems Using UML2, Pascal Roques, WILEY- Dreamtech India Pvt. Ltd.

Semester	**VII**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	3	0	0	3	V20CSTPE09
Name of the Course	**Advanced Computer Architecture (Professional Elective-III)**					
Branch	Common to CSE and CST					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Explain the different types of parallel computer models. **(K2)**
CO2: Describe various Processor and Memory organizations. **(K2)**
CO3: Illustrate Pipelining, Multiprocessors and Multicomputers concepts. **(K2)**
CO4: Explain Multivector, SIMD Computers and Multithreaded, Dataflow Architectures. **(K2)**
CO5: Illustrate the Parallel Programming models and instruction level parallelism. **(K2)**

UNIT-I: Parallel computer models: The state of computing, Multiprocessors and Multicomputers, Multivector and SIMD computers. **Program and network properties:** Conditions of parallelism, Program flow mechanisms.

UNIT-II: Processors: Advanced Processor Technology, Superscalar and Vector Processors, **Memory Hierarchy, Cache and Shared Memory:** Hierarchical Memory Technology, Virtual Memory Technology, Cache Memory Organizations, Shared-Memory Organizations.

UNIT-III: Pipelining: Linear Pipeline Processors, Nonlinear Pipeline Processors, Instruction Pipeline Design, Arithmetic Pipeline Design. **Multiprocessors and Multicomputers:** Multiprocessor System Interconnects, Cache Coherence and Synchronization Mechanisms, Message Passing Mechanisms.

UNIT-IV: Multivector and SIMD Computers: Vector Processing Principles, Compound Vector Processing. **Scalable, Multithreaded, Dataflow Architectures:** Latency-Hiding Techniques, Principles of Multithreading.

UNIT-V: Parallel Models, Languages: Parallel Programming Models, Parallel Languages and Compilers.**Instruction Level Parallelism:** Problem Definition, Model of a Typical Processor, Compiler-detected Instruction Level Parallelism, Operand Forwarding, Reorder Buffer, Register Renaming, Tomasulo's Algorithm, Branch Prediction.

Text Book:

1. Advanced Computer Architecture: Parallelism, Scalability, Programmability, Kai Hwang, Naresh Jotwani, 2nd Edition, Tata McGraw Hill Education

Reference Books:

1. Computer Organization and Design, David A. Patterson and John. L. Hennessy, 5th Edition, Morgan Kaufmann Series.
2. Computer Architecture and Organization, John P. Hayes, 3rd Edition, McGraw Hill Education.
3. Computer Architecture and Organization: Design Principles and Applications, B. Govindarajulu, 2nd Edition, McGraw Hill Education.

Semester	**VII**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	3	0	0	3	V20CSTPE10
Name of the Course	**BigData Analytics (Professional Elective-III)**					
Branch	Common to CSE and CST					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Discuss the challenges of Big Data using Hadoop. **(K2)**
CO2: Apply data modelling techniques to large data sets using map reduce programs. **(K3)**
CO3: Describe the Hadoop I/O classes. **(K2)**
CO4: Examine the use of Pig Framework to work with Big Data. **(K3)**
CO5: Develop a data analytical system using HIVE. **(K3)**

UNIT-I: Introduction to Big Data & Hadoop: What is Big Data, Why Big Data is Important, Data Storage and Analysis, Comparison with other systems. A brief history of Hadoop, Meet Hadoop Data, Apache Hadoop and the Hadoop Ecosystem.
Working with Big Data & HDFS: Google File System, Hadoop Distributed File System (HDFS) – Building blocks of Hadoop (Namenode, Datanode, Secondary Namenode, JobTracker, TaskTracker).

UNIT-II: Introducing and Configuring Hadoop cluster: Local distributed mode, Pseudo-distributed mode, Fully Distributed mode, Configuring XML files.
Writing Map Reduce Programs: Analyzing the Data with Hadoop-Map Reduce, Basic programs of Hadoop Map Reduce, Driver code, Mapper code, Reducer code, Record Reader, Combiner functions. Map Reduce Types, Input Format class Hierarchy.

UNIT-III:Hadoop I/O: The Writable Interface, Writable Comparable and Comparators.
Writable Classes: Writable wrappers for Java primitives, Text & Bytes Writable, Null Writable, Object Writable and Generic Writable, Writable collections.
Implementing a Custom Writable: Implementing a Raw Comparator for speed, Custom comparators

UNIT-IV: Pig - Hadoop Programming Made Easier: Admiring the Pig Architecture, Going with the Pig Latin Application Flow, Working through the ABCs of Pig Latin, Evaluating Local and Distributed Modes of Running Pig Scripts, Checking out the Pig Script Interfaces, Scripting with Pig Latin.

UNIT-V: Applying Structure to Hadoop Data with Hive: Saying Hello to Hive, Seeing How the Hive is Put Together, Getting Started with Apache Hive, Examining the Hive Clients, Working with Hive Data Types, Creating and Managing Databases and Tables, Seeing How the Hive Data Manipulation Language Works, Querying and Analyzing Data.

Text Books:

1. Hadoop: The Definitive Guide, Tom White, O'Reilly, 3rd Edition, 2012.
2. Hadoop in Action, Chuck Lam, MANNING Publ., 2016.
3. Hadoop for Dummies, Dirk deRoos, Paul C.Zikopoulos, Roman B.Melnyk, Bruce Brown, Rafael Coss, 2014.

Reference Books:

1. Hadoop in Practice, Alex Holmes, MANNING Publ., 2014.
2. Hadoop Map Reduce Cookbook, Srinath Perera, Thilina Gunarathne, PACKT, 2013.

Semester	**VII**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	3	0	0	3	V20CSTPE11
Name of the Course	**Deep Learning (Professional Elective-III)**					
Branch	Common to CSE and CST					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Describe the fundamentals of deep learning. **(K2)**
CO2: Illustrate the working of deep feedforward neural networks. **(K2)**
CO3: Discuss regularization and optimization techniques used in deep neural networks. **(K2)**
CO4: Illustrate the working of convolution neural networks. **(K2)**
CO5: Explain about recurrent and recursive neural networks. **(K2)**

UNIT-I: Introduction: Historical Trends in Deep Learning**,** The Many Names and Changing Fortunes of Neural Networks, Increasing Dataset Sizes, Increasing Model Sizes, Increasing Accuracy, Complexity and Real-World Impact.

UNIT-II: Deep Feed forward Networks: Learning XOR, Gradient-Based Learning, Hidden Units, Architecture Design, Back Propagation and Other Differentiation Algorithms.

UNIT-III: Regularization for Deep Learning: Parameter Norm Penalties, Early Stopping, Dropout; **Optimization for Training Deep Models:** How Learning Differs from Pure Optimization, Challenges, Basic Algorithms, Parameter Initialization Strategies, Algorithms with Adaptive Learning Rates, Optimization Strategies and Meta-Algorithms.

UNIT-VI: Convolution Networks: The Convolution Operation, Motivation, Pooling, Convolution and Pooling as an Infinitely StrongPrior, Variants of Basic Convolution Functions, Structured Outputs, Data Types, Efficient Convolution Algorithms, Random or Unsupervised Features, The Neuroscientific Basis for Convolutional Networks, Convolutional Networks and the History of Deep Learning.

UNIT-V: Sequence Modeling- Recurrent and Recursive Nets: Unfolding Computational Graphs, Recurrent Neural Networks, Bidirectional RNNs, Encoder-Decoder Sequence-to-Sequence Architectures, Deep Recurrent Networks, Recursive Neural Networks, The Challenge of Long-Term Dependencies, Echo State Networks, Leaky Units and Other Strategies for Multiple Time Scales, LSTM and Other Gated RNNs, Explicit Memory.

Textbooks:

1. Deep Learning, Ian Goodfellow, YoshuaBengio, and Aaron Courville, MIT Press.

Reference Books:

1. Neural Networks and Deep Learning, Charu C. Aggarwal, Springer.
2. Fundamentals of Deep Learning, Nikhil Buduma, 1st Edition, O'Reilly

Semester	**VII**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	3	0	0	3	V20CSTPE12
Name of the Course	Human Computer Interaction **(Professional Elective-III)**					
Branch	Common to CSE and CST					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Describe the principles and characteristics of GUI. **(K2)**
CO2:Describe how a computer system may be modified to include human diversity. **(K2)**
CO3: Select an effective style and screen design for a specific business application. **(K2)**
CO4: Discuss System Menus & Navigation Schemes**.** **(K2)**
CO5:Select Device and Screen based controls. **(K2)**

UNIT I:The User Interface: Introduction, Importance of the User Interface, Importance and benefits of Good Design, Characteristics of Graphical and Web User Interface Graphical User Interface, popularity of graphics, concepts of Direct Manipulation, Graphical System advantage and disadvantage, Characteristics of GUI, Characteristics of Web Interface, Principles of User Interface Design.

UNITII:TheUserInterfaceDesignProcess:ObstaclesandPitfallinthedevelopmentProcess,Usability, The Design Team, Human Interaction with Computers, Important Human Characteristics in Design, Human Consideration in Design, Human Interaction Speeds, Performance versus Preference, Methods for Gaining and Understanding of Users.

UNIT III: Understanding Business Functions: Business Definitions& Requirement analysis, Determining Business Functions. **Principles of Good Screen Design:** Human considerations in screen Design, interface design goals, screen meaning and purpose, Technological considerations in Interface Design.

UNIT IV: System Menus and Navigation Schemes: Structure, Functions, Context, Formatting, PhrasingandSelecting,NavigatingofMenus,KindsofGraphicalMenusWindowsInterface:Windowscharacteristic,ComponentsofWindow,WindowsPresentationStyles,TypesofWindows,WindowManagement,

UNITV: Device and Screen-Based Control: Device based controls, Operable Controls, Textentry/read-Only Controls, Section Controls, Combining Entry/Selection Controls Presentation Controls, Selecting proper controls.

Text Books:

1. "The Essential Guide to User Interface Design", Wilbert O. Galitz, 2nd edition,2002,Wiley India Edition.
2. Prece, Rogers, "Sharps Interaction Design", Wiley India.
3. "Designing the user interfaces". Ben Shneidermann 3rd Edition, Pearson Education Asia.

Reference Books:

1. "User Interface Design" , SorenLauesen, Pearson Education
2. "Essentials of Interaction Design", Alan Cooper, Robert Riemann, David Cronin, Wiley
3. "HumanComputer Interaction", Alan Dix, Janet Fincay, GreGoryd, Abowd, Russell, Bealg, Pearson Education.

Semester	**VII**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	3	0	0	3	V20CSTPE13
Name of the Course	**Design Patterns (Professional Elective-IV)**					
Branch	Common to CSE and CST					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Describe the design patterns view and its applications . **(K2)**
CO2: Demonstrate Creational Patterns. **(K3)**
CO3: Construct Structural Patterns for a given Scenario. **(K3)**
CO4: Construct Behavioural Patterns for a given Scenario. **(K3)**
CO5: Examine various Case Studies in utilizing Software Architectures **(K3)**

UNIT I: Introduction: What Is a Design Pattern?, Design Patterns in Smalltalk MVC, Describing Design Patterns, The Catalog of Design Patterns, Organizing the Catalog, How Design Patterns Solve Design Problems, How to Select a Design Pattern, How to Use a Design Pattern

UNIT II: Creational Patterns: Abstract factory, Builder, Factory method, Prototype, Singleton.

UNIT III: Structural Patterns: Adapter, Bridge, Composite, Decorator, Façade, Flyweight, and PROXY.

UNIT VI: Behavioural Patterns: Chain of responsibility, command, Interpreter, iterator, mediator, memento, observer, state, strategy, template method, visitor.

UNIT V: Case Studies A-7E – A case study in utilizing architectural structures, The World Wide Web - a case study in Interoperability, Air Traffic Control – a case study in designing for high availability, Celsius Tech – a case study in product line development.

Text Books:

1. Software Architecture in Practice, second edition, Len Bass, Paul Clements & Rick Kazman, Pearson Education, 2003.
2. Design Patterns, Erich Gamma, Pearson Education, 1995.

Reference Books:

1. Beyond Software architecture, Luke Hohmann, Addison wesley, 2003.
2. Software architecture, David M. Dikel, David Kane and James R. Wilson, Prentice Hall PTR, 2001
3. Software Design, David Budgen, second edition, Pearson education, 2003
4. Head First Design patterns, Eric Freeman & Elisabeth Freeman, O'REILLY, 2007.
5. Design Patterns in Java, Steven John Metsker & William C. Wake, Pearson education, 2006.

Semester	**VII**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	3	0	0	3	V20CSTPE14
Name of the Course	**NOSQL Database (Professional Elective-IV)**					
Branch	Common to CSE and CST					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Discuss four types of NoSQL Databases (Document-oriented, KeyValue Pairs, Column oriented and Graph). **(K2)**

CO2: Illustrate Replication and sharding. **(K2)**

CO3: Explain NoSQL Key/Value databases using MongoDB. **(K2)**

CO4: Demonstrate Column- oriented NoSQL databases using Apache HBASE. **(K3)**

CO5: Explain Graph NoSQL databases using Neo4. **(K3)**

UNIT I: Introduction: Overview and History of NoSQL Databases Definition of the Four Types of NoSQL Database, The Value of Relational Databases, Getting at Persistent Data, Concurrency, Integration, Impedance Mismatch, Application and Integration Databases, Attack of the Clusters, The Emergence of NoSQL, Key Points, Comparison of relational databases to new NoSQL stores, MongoDB, Cassandra, HBASE, Neo4j use and deployment, Application, RDBMS approach, Challenges NoSQL approach, Key-Value and Document Data Models, Column-Family Stores, Aggregate-Oriented Databases.

UNIT II: Replication and sharding, MapReduce on databases. Distribution Models, Single Server, Sharding, Master-Slave Replication, Peer-to-Peer Replication, Combining Sharding and Replication.

UNIT III: NoSQL Key/Value databases using MongoDB, Document Databases, What Is a Document Database? Features, Consistency, Transactions, Availability, Query Features, Scaling, Suitable Use Cases, Event Logging, Content Management Systems, Blogging Platforms, Web Analytics or Real-Time Analytics, E-Commerce Applications, When Not to Use, Complex Transactions Spanning Different Operations, Queries against Varying Aggregate Structure.

UNIT IV: Column- oriented NoSQL databases using Apache HBASE, Column-oriented NoSQL databases using Apache Cassandra, Architecture of HBASE, What Is a Column-Family Data Store? Features, Consistency, Transactions, Availability, Query Features, Scaling, Suitable Use Cases, Event Logging, Content Management Systems, Blogging Platforms, Counters, Expiring Usage, When Not to Use

UNIT V: Graph NoSQL databases using Neo4,NoSQL database development tools and programming languages, Graph Databases, What Is a Graph Database? Features, Consistency, Transactions, Availability, Query Features, Scaling, Suitable Use Cases, Connected Data, Routing, Dispatch, and Location-Based Services, Recommendation Engines, When Not to Use

Textbooks:

1. NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence , **1st Edition, 2012.** Authors: Sadalage, P. & Fowler, Publication: Pearson Education.
2. The Definitive Guide to MongoDB: A complete guide to dealing with Big Data using MongoDB, **3rd Edition, December, 2015**. Authors: Eelco Plegge, David Hows, Peter Membrey, Tim Hawkins, Apress Publishers

Reference Books:

1. Redmond, E. ,Wilson, Perkins: Seven Databases in Seven Weeks: A Guide to Modern Databases and the NoSQL Movement Edition**: 2nd Edition,2018**, O′Reilly Publishers.

Semester	**VII**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	3	0	0	3	V20CSTPE15
Name of the Course	**Reinforcement Learning (Professional Elective-IV)**					
Branch	Common to CSE and CST					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Discuss Elements of Reinforcement Learning and Multi-armed Bandits. **(K2)**
CO2: Illustrate Finite Markov Decision Process and Dynamic Programming. **(K2)**
CO3: Explain Monte Carlo Methods and *n*-step Bootstrapping. **(K2)**
CO4: Explain Off-policy Methods with Approximation. **(K2)**
CO5: Discuss Policy Gradient Methods. **(K2)**

UNIT I: Introduction: Reinforcement Learning, Examples, Elements of Reinforcement Learning, Limitations and Scope. **Multi-armed Bandits:** A k-armed Bandit Problem, Action-value methods, The 10-armed Testbed, Incremental Implementation, Tracking a Non stationary Problem, Optimistic Initial Values, Upper –Confidence-Bound Action Selection, Gradient Bandit Algorithm.

UNIT II:Finite Markov Decision Process: The Agent-Environment Interface, Goals and Rewards, Returns and Episodes, Unified Notataion for Episodic and Continuing Tasks, Policies and Value Functions. **Dynamic Programming:** Policy Evaluation, Policy Improvement, Policy Iteration, Value Iteration, Asynchronous Dynamic Programming, Generalized Policy Iteration, Efficiency of Dynamic Programming.

UNIT III: Monte Carlo Methods: Monte Carlo Prediction, Monte Carlo Estimation of Action Values, Monte Carlo Control, Monte Carlo Control without Exploring Starts, Off-policy Prediction via Importance Sampling, Incremental Implementation, Discontinuing-aware Importance Sampling, Per-decision Importance Sampling. ***n*-step Bootstrapping:** *n*-step TD Prediction, *n*-step Sarsa, *n*-step Off-policy Learning, Per-decision methods with Control Variables, A Unifying Algorithm: *n*-step Q(σ).

UNIT IV:
Off-policy Methods with Approximation: Semi-gradient Methods, Examples of Off-policy Divergence, The Deadly Triad, Linear Value-function Geometry, Gradient Descent in the Bellman Error, The Bellman Error is not Learnable, Gradient-TD methods, Emphatic-TD methods, Reducing Variance.
Eligibility Traces: The λ-return, TD(λ), *n*-step Truncated λ-return methods, Online λ –return Algorithm, True Online TD(λ), Dutch Traces in Monte Carlo Learning, Sarsa(λ), Variable λ and γ, Off-policy Traces with Control Variables, Watkins's Q(λ) to Tree-Backup(λ).

UNIT V:
Policy Gradient Methods: Policy Approximation and its Advantages, The Policy Gradient Theorem, REINFOECE: Monte Carlo Policy Gradient, REINFORCE with Baseline, Actor-Critic Methods, Policy Gradient for Continuing Problems, Policy Parameterization fr Continuous Actions.

Text Books:

1. R. S. Sutton and A. G. Bart,. "Reinforcement Learning - An Introduction," MIT Press, 2018.

Reference Books:

1. Szepesvári, Csaba, "Algorithms for Reinforcement Learning," United States: Morgan & Claypool, 2010.
2. Puterman, Martin L., "Markov Decision Processes: Discrete Stochastic Dynamic Programming," Germany: Wiley, 2014.

Semester	**VI**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	3	0	0	3	V20CSTPE16
Name of the Course	**Cloud Computing (Professional Elective-IV)**					
Branch	Common to CSE and CST					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Explain the basic concepts of cloud computing. **(K2)**
CO2: Describe the Virtualization and Migration concepts of Cloud. **(K2)**
CO3: Explain the Cloud Application Design methodologies. **(K2)**
CO4: Illustrate the Security aspects of Cloud. **(K2)**
CO5: Illustrate theSLA Management aspects of Cloud. **(K2)**

UNIT-I: Introduction to Cloud Computing: Definition of Cloud Computing, Layers and Types Of Clouds, Desired Features of a Cloud, Infrastructure as a Service Providers, Platform as a Service Providers, Challenges and Risks.

UNIT-II: Cloud Concepts & Technologies: Virtualization, Load Balancing, Replication, Software Defined Networking, Network Function Virtualization (NFV).
Migrating into a Cloud: The Seven-Step Model of Migration into a Cloud, Migration Risks and Mitigation

UNIT-III: Cloud Application Design: Design Considerations for Cloud Applications, Reference Architectures for Cloud Applications, Cloud Application Design Methodologies: SOA, Cloud Component Model, MVC, Data Storage Approaches.

UNIT-IV: Cloud Security: Cloud Security Architecture (CSA), Authentication, Authorization, Identity, Access Management, Data Security, Key Management.

UNIT-V: SLA Management in Cloud Computing: Service Level Agreements (SLA), Traditional Approaches to SLO Management, Types of SLA, Life Cycle of SLA, SLA Management in Cloud.

Text Books:

1. Cloud Computing: Principles and Paradigms,RajkumarBuyya, James Broberg, Andrzej Goscinski, Wiley Publication.
2. Cloud Computing: A Hands-on Approach, ArshdeepBahga, Vijay Madisetti,Universities Press.

Reference Books:

1. Cloud Computing – Web-Based Applications That Change the way you Work and Collaborate Online, Michael Miller, PearsonEducation.
2. Cloud Computing: A Practical Approach,Anthony T. Velte, Toby J. Velte, Robert Elsenpeter, McGraw-Hill,(2010).

Semester	**VII**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	3	0	0	3	V20CSTPE17
Name of the Course	**Software Project Management (Professional Elective-V)**					
Branch	Common to CSE and CST					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Describe Software Project Management Terminology. **(K2)**

CO2: Explain various Software development process Models and software Life cycle phases. (**K2)**

CO3: Illustrate various Effort Estimation Techniques and activity network models for Software Project Planning. **(K3)**

CO4: Demonstrate Risk Management Concepts and resource allocation. **(K3)**

CO5: Explain the importance of Project monitoring and control for accomplishing project goals and software Quality. **(K2)**

UNIT-I: Introduction to Software Project Management: Software Project versus other types of projects, Activities covered by Software Project Management, Categorizing projects ,Stakeholders, Objectives& goals, what is management. **Project Planning:** Step-wise planning, Identify Project Scope and objectives, Infrastructure, Project Products &deliverables, Project activities, Effort estimation.

UNIT-II: Project Approach: Build or buy, process models: waterfall model, Prototyping, Incremental delivery model, **Agile methods**: Extreme Programming, Atern method, selecting an appropriate process model. **Lifecycle phases:** Engineering and Production stages, Inception, Elaboration, Construction, Transition phases.

UNIT-III: Software effort estimation and Activity planning: Overview of Effort Estimation techniques, Function Point analysis, COCOMO. **Activity planning:** Objectives, Network planning models, forward pass and backward pass, Identify Critical path and activities.

UNIT-VI: Risk Management and Resource Allocation: Introduction, Risk and its categories, Identification, Assessment, Risk Planning and management, applying PERT technique.
Resource Allocation: Types of Resources, Identifying resource requirements, Resource scheduling.

UNIT-V: Project Monitoring and Control: Creating framework for monitoring& control, Collecting Data, Visualizing Progress, Cost monitoring, Earned value Analysis.
Software Quality: Defining Quality, Importance of quality, ISO 9126, Product QualityVs Process Quality management.
Process Capability Models: Capability Maturity Model, Enhancing software Quality.

Text Books:

1. Software Project Management, Bob Hughes & Mike Cotterell, 6 th edition, TATA Mcgraw-Hill 2. Software Project Management, WalkerRoyce 2nd edition, Pearson Education.

Reference Books:

1. Software Project Management in practice, PankajJalote, 9th edition, Pearson Education.
2. Software Project Management, Joel Henry, 3rd edition, Pearson Education.

Semester	**VII**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	3	0	0	3	V20CSTPE18
Name of the Course	**Scripting Languages (Professional Elective-V)**					
Branch	Common to CSE and CST					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Develop dynamic webpages and validate with java Script. **(K3)**
CO2: Discuss fundamentals of PHP. **(K2)**
CO3: Develop web applications using PHP. **(K3)**
CO4: Demonstrate Perl Programming concepts. **(K3)**
CO5: Illustrate AngularJS frame work. **(K2)**

UNIT – I: JavaScript: Overview of JavaScript, General Syntactic Characteristics, Primitives Operations and Expressions, Screen output and Keyboard Input, Control Statements, Object creation and Modification, Arrays, Functions, Constructors, Pattern Matching using Regular Expressions, Events and Event Handling. **DHTML:** Positioning Moving and Changing Elements.

UNIT – II: PHP Basics- Features, Embedding PHP Code in your Web pages, Outputting the data to the browser, Data types, Variables, Constants, expressions, string interpolation, control structures, Function, Creating a Function, Function Libraries, Arrays, strings and Regular Expressions.

UNIT - III: Advanced PHP Programming: PHP and Web Forms, Files, PHP Authentication and Methodologies - Database Based, Login Administration, Uploading Files with PHP, Sending Email using PHP.

UNIT – IV: Introduction to PERL and Scripting: Scripts and Programs, Origin of Scripting , Scripting Today, Characteristics of Scripting Languages, Uses for Scripting Languages, Web Scripting, and the universe of Scripting Languages. PERL- Names and Values, Variables, Scalar Expressions, Control Structures, arrays, list, hashes, strings, pattern and regular expressions, subroutines.

UNIT- V: AngularJS - Overview, environment Setup, MVC Architecture, Creating AngularJS Application, Directives, Expressions, Controllers, Filters, Tables, HTML DOM, Modules, Forms.

Text Books:
1. The World of Scripting Languages, David Barron, Wiley Publications.
2. A Journey to Angular Development, by Sukesh Marla, bpb publisher

Reference Books:
1. Open Source Web Development with LAMP using Linux, Apache, MySQL, Perl and PHP, J.Lee and B.Ware (Addison Wesley) Pearson Education.
2. Perl by Example, E.Quigley, Pearson Education.
3. Programming Perl, Larry Wall T.Christiansen and J.Orwant, O'Reilly, SPD.
4. Tcl and the Tk Toolkit, Ousterhout, Pearson Education.
5. Pearl Power, J.P. Flynt, Cengage Learning.
6. Learn Angular in 24 Hours A Step-by-Step Approach, Lakshmi Kamala Thota.

Semester	**VII**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	3	0	0	3	V20CSTPE19
Name of the Course	**Natural Language Processing (Professional Elective-V)**					
Branch	Common to CSE and CST					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Illustrate Natural Language Processing tasks in syntax, semantics, and pragmatics. **(K2)**
CO2: Classify Morphology and Finite State Transducers, Markov Models and Entropy Models. **(K2)**
CO3: Explain about Statistical parsing and probabilistic CFGs. **(K2)**
CO4: Demonstrate semantic analysis. **(K2)**
CO5: Explain Discourse Analysis and Lexical Resources. **(K2)**

UNIT-I : Introduction: Natural Language Processing tasks in syntax, semantics, and pragmatics– Issues–Applications- The role of machine learning - Probability Basics–Information theory– Collocations-N-gram Language Models - Estimating parameters and smoothing – Evaluating language models.

UNIT-II : Morphology And Part Of Speech Tagging: Linguistic essentials - Lexical syntax-Morphology and Finite State Transducers - Part of speech Tagging - Rule-Based Part of Speech Tagging - Markov Models - Hidden Markov Models – Transformation based Models - Maximum Entropy Models. Conditional Random Fields.

UNIT-III : Syntax Parsing: Syntax Parsing - Grammar formalisms and treebanks - Parsing with Context Free Grammars ,Features and Unification-Statistical parsing and probabilistic CFGs(PCFGs)-Lexicalized PCFGs.

UNIT-IV : Semantic Analysis: Representing Meaning – Semantic Analysis - Lexical semantics – Word-sense disambiguation- Supervised – Dictionary based and Unsupervised Approaches - Compositional semantics- Semantic Role Labeling and Semantic Parsing – Discourse Analysis.

UNIT-V : Discourse Analysis and Lexical Resources: Discourse segmentation, Coherence – Reference Phenomena, Anaphora Resolution using Hobbs and Centering Algorithm – Coreference Resolution – Resources: Porter Stemmer, Lemmatizer, Penn Treebank, Brills Tagger, WordNet, PropBank, FrameNet, Brown Corpus, British National Corpus (BNC). NLP Applications**:** Named entity recognition and relation extraction- IE using sequence labeling-Machine Translation (MT) .

Text Books:

1. Daniel Jurafsky and James H. Martin Speech and Language Processing (2nd Edition), Prentice Hall; 2 edition,2008
2. FoundationsofStatisticalNaturalLanguageProcessingbyChristopherD.Manningand Hinrich Schuetze, MIT Press,1999
3. Steven Bird, Ewan Klein and Edward Loper Natural Language Processing with Python, O'Reilly Media; 1 edition,2009 Roland R. Hausser, Foundations of Computational Linguistics: Human-Computer Communication in Natural Language, Paperback, MIT Press,2011

Reference Books:

1. Pierre M. Nugues, An Introduction to Language Processing with Perl and Prolog: An Outline of Theories, Implementation, and Application with Special Consideration of English, French, and German (Cognitive Technologies) Softcover reprint,2010
2. James Allen, Natural Language Understanding, Addison Wesley; 2 edition 1994
NLTK – Natural Language Tool Kit -http://www.nltk.org/

Semester	**VII**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	3	0	0	3	V20CSTPE20
Name of the Course	Social Networks and Semantic Web **(Professional Elective-V)**					
Branch	Common to CSE and CST					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Demonstrate knowledge by explaining the three different "named" generations of the web. **(K3)**
CO2: Construct a social network. **(K3)**
CO3: Relate knowledge representation methods for semantic web. **(K3)**
CO4: Describe web services and its Applications. **(K2)**
CO5: Develop "Linked Data" Applications using Semantic Web Technologies. **(K3)**

UNIT-I: The Semantic web: Limitations of the current Web, The semantic solution, Development of the Semantic Web, The emergence of the social web.

UNIT-II: Social Network Analysis: What is network analysis? Development of Social Network Analysis, Key concepts and measures in network analysis. Electronic sources for network analysis: Electronic discussion networks, Blogs and online communities, Web-based networks.

UNIT-III: Knowledge Representation on the Semantic Web: Ontologies and their role in the Semantic Web, Ontology languages for the semantic Web.

Modeling and Aggregating Social Network Data: State of the art in network data representation, Ontological representation of Social individuals, Ontological representation of social relationships, Aggregating and reasoning with social network data.

UNIT-IV: Developing social semantic applications: Building Semantic Web applications with social network features, Flink- the social networks of the Semantic Web community, Open academia: distributed, semantic-based publication management.

UNIT-V: Evaluation of Web-Based Social Network Extraction: Differences between survey methods and electronic data extraction, context of the empirical study, Data collection, Preparing the data, optimizing goodness of fit, Comparison across methods and networks, Predicting the goodness of fit, Evaluation through analysis.

Text Books:

1. Social Networks and the Semantic Web, PeterMika, Springer,2007.
2. Semantic Web Technologies, Trends and Research in Ontology basedsystems, J.Davies,RudiStuder,PaulWarren,JohnWiley&Sons.

Reference Books:

1. Semantic Web and Semantic Web Services –Liyang Lu Chapman and Hall/CRC Publishers, (Taylor & Francis Group).
2. Information Sharing on the semantic Web – Heiner Stuckenschmidt; Frank Van Harmelen, Springer Publications.

Annexure-III

List of Job Oriented Elective Courses

S.No.	Course Code	Name of the Course
1.	V20CSTJE01	Master Coding and Competitive Programming - Part-1
2.	V20CSTJE02	Master Coding and Competitive Programming - Part-2
3.	V20CSTJE03	Full Stack Technologies
4.	V20CSTJE04	DevOps
5.	V20CSTJE05	Blockchain Technologies

NOTE: All the Job oriented can be theory / Lab Course

Semester	**V**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	0	0	6	3	V20CSTJE01
Name of the Course	**Master Coding and Competitive Programming - Part-1 (Job Oriented Elective-I)**					
Branch	Common to CSE and CST					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Apply Mathematical reasoning and number theory to solve real world problems in linear time. **(K3)**
CO2: Use of modular arithmetic, to solve complex problems in linear time , logarithmic. **(K3)**
CO3: Use of Prime Factorization and complex solve problems. **(K3)**
CO4: Analyse different techniques including sieve to find prime numbers and evaluate efficiency of these methods. **(K4)**
CO5: Experiment with Hashing and searching techniques to solve problems on Arrays in Linear time. **(K3)**

List of Experiments

1. Develop Programs to solve problems based on Mathematical logic, Reasoning and number theory
2. Develop programs using different techniques to find prime number
3. Develop programs using Sieve method and optimize Complexity of finding prime number
4. Develop Programs based on series, patterns
5. Develop programs on concept of Fibonacci series
6. Develop programs on strings including palindrome and anagram concepts
7. Develop programs to search pattern in a string
8. Develop programs for String Processing.

Text Books:

1. Java The Complete Reference - Eleventh Edition, Herbert Schildt, Oracle
2. Guide to Competitive Programming by Antti Laaksonen
3. Programming challenges by Steven S Skiena

Tools:

1. practice.geeksfoegeeks.com
2. leetcode.com
3. codingninjas.com
4. Hackerrank.com
5. Interviewbit.com

Semester	**VI**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	0	0	6	3	V20CSTJE02
Name of the Course	**Master Coding and Competitive Programming - Part-2 (Job Oriented Elective-II)**					
Branch	Common to CSE and CST					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Apply Divide and Conquer algorithm technique to solve complex in logarithmic time. **(K3)**

CO2: Apply Greedy method to solve Optimization and decision making problems. **(K3)**

CO3: Apply Backtracking Algorithm technique to find combinatorial problems. **(K4)**

CO4: Experiment with Dynamic Programming Algorithm technique to solve Problems that uses Optimal substructures. **(K3)**

CO5: Develop programs using LinkedList Graphs, DFS and BFS techniques. **(K3)**

List of Experiments

1. Develop Programs to solve problems based on Divide and Conquer Algorithm Technique.
2. Develop programs using two pointer and sliding window algorithms.
3. Problem Solving using Greedy Algorithm technique.
4. Problem Solving using Backtracking.
5. Develop programs using Dynamic Programming and Kadane Algorithm.
6. Develop programs using Linked List and its applications.
7. Develop programs using Graphs and Graph Searching Techniques.

Text Books:

1. Introduction to Algorithms, Second Edition, Thomas H. Cormen Charles E. Leiserson.

2. Data Structures and Algorithms Made Easy: Narasimha Karumanchi .

3. The Algorithm Design Manual, Springer series, Steven Skiena.

Tools:

1. practice.geeksfoegeeks.com
2. leetcode.com
3. codingninjas.com
4. Hackerrank.com
5. Interviewbit.com

Semester	**VII**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	0	0	6	3	V20CSTJE03
Name of the Course	**Web Application Development Using Java Full Stack (Job Oriented Elective)**					
Branch	Common to CSE and CST					

Syllabus Details

After Successful completion of the Course, the students will be able to:

CO1: Demonstrate IDE tolls Installation. **(K3)**
CO2: Develop programs using servlets. **(K3)**
CO3: Illustrate MVC architecture. **(K3)**
CO4: Demonstrate applications of Hibernate. **(K3)**
CO5: Illustrate Spring MVC Framework. **(K3)**

Exercise 1: Basic Installation of IDEs and Development Tools (use any one of the following IDEs).

The Student should know about installing IDEs (Integrated Development Environment) in the system such as IntelliJ, Eclipse, NetBeans, Macromedia Dream Viewer and Databases such as My-SQL, Oracle, SQL Server etc.

Additional Tasks:

- How we can import project files into IDEs.
- How we can import eclipse (Java IDE) projects.
- How to Create new project in IDEs.
- How to Save the Project using packages.
- How to Compile the Project or Program in IDE.
- How to Build the Project or Program in IDE.
- How to Debug the Errors in IDE.

Exercise 2: Understanding about Servlets: Create Example programs Using the below concepts

- Introduction to Servlets.
- Write Servlet application to print current date & time.
- Write Servlet program to link Html & Servlet Communication.
- Write Servlet program to Auto refresh a page.
- Demonstrate session tracking using small program.
- Write Servlet program to insert/delete/update the record into database.
- Write Servlet program to add cookie to selected value.

Exercise 3: Understanding about Model View Controller : Create Example programs Using the below concepts

- Introduction to MVC in java.
- Create sample program on Model Layer in MVC Using Java.
- Create sample program on View Layer in MVC Using Java
- Create sample program on Controller Layer in MVC Using Java
- Demonstrate MVC Deployment in java.
- Rules for MVC Mapping in Server Side.
- How to use Web Server for MVC Deployment.

Exercise 5: Understanding about Hibernate : Create Example programs Using the below concepts

- Introduction to Hibernate.
- What is ORM
- Demonstrate the components of Hibernate
- How to persist objects using Hibernate
- How to use map using XML and Annotations
- How to implement Inheritance in Hibernate
- Working with relationship between entities - association
- Transactions in Hibernate
- Querying with HQL (Hibernate Query Language)
- Various other forms of querying - Criteria, QBE etc.

Exercise 4: Understanding about Spring MVC Framework: Create Example programs Using the Below concepts

- Introduction to Spring MVC.
- Demonstrate the usage of Dispatcher Servlet in Spring MVC.
- Load the spring jar files or add dependencies in the case of Maven
- Create the controller class.
- Provide the entry of controller in the web.xml file.
- Define the bean in the separate XML file.
- Display the message in the JSP page.
- Start the server and deploy the project.
- Execute the application on webserver using Spring MVC.

Exercise 6: Understanding Some Debugging Tools in Java : The Student should know about how to debug the java codes using some debugging tools such as:

- NetBeans.
- Eclipse.
- IntelliJ IDEA.
- Visual Studio Code.

Reference Books:

1. Jeffrey C. Jackson, "Web Technologies--A Computer Science Perspective", Pearson Education, 2006.
2. Murach's Java Servlets and JSP, 3rd Edition by (Murach: Training & Reference) 3rd Edition.
3. Spring and Hibernate Paperback – 1 July 2017 by K. Santosh Kumar.
4. Full Stack Java Development with Spring MVC, Hibernate, jQuery, and Bootstrap by Mayur Ramgir,Wiley.

Semester	VII	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	3	0	0	3	V20CSTJE04
Name of the Course	DEVOPS **(Job Oriented Elective)**					
Branch	Common to CSE and CST					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Discuss the traditional software development. **(K2)**
CO2: Discuss the concepts of rise of agile methodologies. **(K2)**
CO3: Discuss the concept of DevOps and Agile. **(K2**)
CO4: Demonstrate the purpose of DevOps. **(K3)**
CO5: Illustrate the Operations of CAMS. **(K2)**

UNIT-I: Traditional Software Development: The Advent of Software Engineering - Waterfall method - Developers vs IT Operations conflict.

UNIT-II: Rise of Agile Methodologies: Agile movement in 2000 - Agile Vs Waterfall Method - Iterative Agile SoftwareDevelopment - Individual and team interactions over processes and tools – Workingsoftware over - comprehensive documentation - Customer collaboration over contract negotiation - Responding to change over following a plan.

UNIT-III: Definition of DevOps: Introduction to DevOps - DevOps and Agile.

UNIT-IV: Purpose of DevOps: Minimum Viable Product - Application Deployment - Continuous Integration - Continuous Delivery.

UNIT-V: **CAMS (Culture, Automation, Measurement And Sharing)**: CAMS – Culture - CAMS – Automation - CAMS – Measurement - CAMS – Sharing - Test-Driven Development - Configuration Management - Infrastructure Automation - Root Cause Analysis – Blamelessness - Organizational Learning.

Text Books:
1. The DevOps Handbook - Book by Gene Kim, Jez Humble, Patrick Debois, and Willis Willis.

Reference Books:
1. What is DevOps? - by Mike Loukides.

Semester	VII	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	3	0	0	3	V20CSTJE05
Name of the Course	**Blockchain Technology (Job Oriented Elective)**					
Branch	Common to CSE and CST					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Discuss the Cryptographic primitives used in Blockchain. **(K2)**
CO2: Discuss about various technologies borrowed in Blockchain. **(K2)**
CO3: Illustrate various models for Blockchain. **(K2)**
CO4: Discuss about Ethereum. **(K2)**
CO5: Discuss about Hyperledger Fabric. **(K2)**

UNIT-I: Introduction: History of Bitcoin and origins of Blockchain, Fundamentals of Blockchain and key components, Permission and Permission-less platforms, Cryptography, SHA256 and ECDSA, Hashing and Encryption, Symmetric/ Asymmetric keys, Private and Public Keys.

UNIT-II: Technologies Borrowed in Blockchain: Technologies Borrowed in Blockchain–hash pointers--Digital cashetc.-Bitcoin Blockchain-Wallet–Blocks Merkley Tree - hardness of mining - Transaction verifiability - Anonymity -forks - Double spending - Mathematical analysis of properties of Bitcoin -Bitcoin-the challenges and solutions.

UNIT-III: Consensus Mechanisms: Consensus Algorithms: Proof of Work(PoW) as random oracle-Formal treatment of consistency-Liveness and Fairness-Proof of Stake(PoS)based Chains -Hybrid models (PoW + PoS), Byzantine Models of fault tolerance.

UNIT-IV: Ethereum: Ethereum- Ethereum Virtual Machine(EVM)-Wallets for Ethereum-Solidity-Smart Contracts-The Turing Completeness of Smart Contract Languages and verification challenges- Using smart contracts to enforce legal contracts-Comparing Bitcoin scripting vs. Ethereum Smart Contracts-Some attacks onsmart contracts.

UNIT-V: Hyperledger Fabric: Hyperledger fabric- the plug and play platform and mechanisms inpermissioned block chain - Beyond Cryptocurrency – applications of blockchain in cyber security-integrity of information-E-Governance and other contract enforcement mechanisms-Limitations of blockchain as a technology and myths vs reality of Blockchain technology.

Textbooks:

1. S.Shukla,M.Dhawan,S.Sharma,S.Venkatesan"BlockchainTechnology:CryptocurrencyandApplications",OxfordUniversityPress2019.
2. Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller andStevenGoldfeder,"Bitcoinandcryptocurrencytechnologies:acomprehensiveintroduction",PrincetonUniversityPress,2016.

Reference Books:

1. Joseph Bonneau et al, SoK: "Research perspectives and challenges forBitcoinandcryptocurrency",IEEESymposiumonsecurityandPrivacy,2015
2. J.A.Garayetal,"Thebitcoinbackboneprotocol-analysisandapplications",EUROCRYPT2015,Volume2.
3. R.Passetal,"AnalysisofBlockchainprotocolinAsynchronousnetworks",EUROCRYPT2017.
4. Passetal,"Fruitchain-afairblockchain",PODC2017.

Annexure-IV

Open Electives

The following courses are offered to the students of other departments.

S.No.	Course Code	Name of the Course
1.	V20CSTOE01	Python Programming Lab
2.	V20CSTOE02	Advanced Python Programming Lab
3.	V20CSTOE03	Operating Systems
4.	V20CSTOE04	Software Engineering
5.	V20CSTOE05	Object Oriented Programming through Java Lab
6.	V20CSTOE06	Computer Graphics
7.	V20CSTOE07	Software Testing Methodologies
8.	V20CSTOE08	Linux Shell Scripting Lab
9.	V20CSTOE09	Computer Networks
10.	V20CSTOE10	Cryptography and Network Security
11.	V20CSTOE11	Database Management Systems Lab
12.	V20CSTOE12	Human Computer Interaction

Semester	**V to VII SEM**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	0	0	6	3	V20 CSTOE01
Name of the Course	Python Programming Lab **(Open Elective)**					
Branch	Common to CIVIL,MECH,EEE,ECE,ECT,CAI & AIM					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Demonstrate Basic Python Programs **(K3)**
CO2: Construct control structures in python. **(K3)**
CO3: Demonstrate functions and packages. **(K3)**
CO4: Construct python programs using structured data types. **(K3)**
CO5: Demonstrate Text Files. **(K3)**

Syllabus:

Basics of python programming: Features of python – History of Python - The Future of Python installation and execution - Data types – Identifiers - variables – type conversions- Literal Constants – Numbers – Strings. I/O statements. Operators and expressions, operator precedence – expression evaluation.

Exercise 1 - Basics

a) A sample Python Script using command prompt, Python Command Line and IDLE
b) A program to purposefully raise an Indentation Error and correct it

Exercise 2 - Operations

a) A program to compute distance between two points taking input from the user (Pythagorean Theorem)
b) A program on add.py that takes 2 numbers as command line arguments and prints its sum.

Decision Control statements: conditional (if), alternative (if-else), chained conditional (if-elif-else); **Iteration:** while loop, for loop, nested for loop, range function, break, continue and pass statements.

Exercise - 3 Control Flow

a) A Program to implement for checking whether the given number is a even number or not.
b) A program to construct reverse the digits of a given number and add it to the original, If the sum is not a palindrome repeat this procedure.
c) A program using a while loop that asks the user for a number, and prints a countdown from that number to zero.

Exercise 4 - Control Flow – Continued

a) A program to construct the following pattern, using a nested for loop.

```
*
* *
* * *
* * * *
* * * * *
* * * *
* * *
* *
*
```

b) By considering the terms in the Fibonacci sequence whose values do not exceed four million, find the sum of the even-valued terms.

Functions and modules : Introduction - Function Definition - Function Call – argument types- Scope and Lifetime - The return statement - More on Defining Functions - Lambda Functions or Anonymous Functions.

Exercise - 5 – Problem Solving using Functions
a) Find mean, median, mode for the given set of numbers passed as arguments to a function
b) Develop a function nearly_equal to test whether two strings are nearly equal. Two strings a and b are nearly equal when a can be generated by a single mutation on b.
c) Develop a Recursive Function to find the Factorial of a given number .
d) Develop function to compute gcd, lcm of two numbers. Each function shouldn't exceed one line.

Lists: list operations, list slices, list methods, mutability, cloning lists, list parameters. **Tuples**: tuple assignment, tuple as return value. **Set:** Set Creation, Set Operations. **Dictionaries**: Creation, operations; comprehension, operations on strings.

Exercise - 6 Structured Data types
a) a program to count the number of strings where the string length is 2 or more and the first and last character are same from a given list of strings.
b) a program to develop unzip a list of tuples into individual lists and convert them into dictionary.

Exercise – 7 Structured Data types Continued
a) A program to count the numbers of characters in the string and store them in a dictionary data structure
b) a program to use split and join methods in the string and trace a birthday with a dictionary data structure.

Documentation Strings- Modules – Packages

Exercise - 8– Modules
a) Install packages requests, flask and explore them using (pip)
b) A program to implement a script that imports requests and fetch content from the page. Eg. (Wiki)
c) Develop a simple script that serves a simple HTTP Response and a simple HTML Page

Introduction - Types of files - Text files - reading and writing files

Exercise - 9 Files
a) a program to count frequency of characters in a given file. Can you use character frequency to tell whether the given file is a Python program file, C program file or a text file?
b) a program to compute the number of characters, words and lines in a file.

Classes, Methods, Constructor, Inheritance, Overriding Methods, Data hiding

Exercise - 10 OOP
a) Class variables and instance variable and illustration of self variable
i) Robot
ii) ATM Machine

Text Books:

1. "Python Programming using problem solving Approach" ReemaThareja, Oxford University Press – 2017.
Python with Machine Learning by A.Krishna Mohan, Karunakar&T.Murali Mohan by S. Chand Publisher.

Semester	V to VII SEM	L	T	P	C	COURSE CODE
Regulation	V20	0	0	6	3	V20CSTOE02
Name of the Course	Advanced Python Programming Lab (Open Elective)					
Branch	Common to CIVIL,MECH,EEE,ECE,ECT,CAI & AIM					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Develop Python Programs using regular expressions. (**K3**)
CO2: Develop programs using GUI. (**K3**)
CO3: Construct programs using Numpy Arrays. (**K3**)
CO4: Develop python programs using pandas. (**K3**)
CO5: Develop charts using matplotlib. (**K3**)

LIST OF EXPERIMENTS

1. Regular expressions & GUI:

a) Develop a python program to create regular expression to replace a string with a new string.
b) Develop a python program to create regular expression to retrieve all the words starting with 'a' in a given string and other create other regular expression to retrieve all the words with size 5.
c) A Python Program to create a regular expression to search for string using search() , findall() , match()
d) A Python Program to create a regular expression to extract E-mail id from files.
e) A Python Program to create a regular expression to extract Phone number from files.
f) A Python Program to create a regular expression to extract words whose length is greater than 4 from files **GUI**
g) A Python Program to draw different shapes on canvas
h) A Python Program to create a push button and bind it with an event handler function using command option
i) A Python Program to design a simple calculator
j) A Python Program to create check boxes and display the content of selected boxes

2. Numpy

a. A Python Program to split arrays using numpy module.
b. A Python Program to test all aggregate functions in numpy module
c. A Python Program to generate a matrix of random numbers within range and print its Transpose
d. A Python Program that calculates variance, co variance, correlation by taking a sample statistical data.
e. Write a python program to find rank, determinant, and trace of an array.
f. Write a python program to find eigenvalues of matrices
g. Write a python program to find matrix and vector products (dot, inner, outer, product), matrix exponentiation.
h. Write a python program to solve a linear matrix equation, or system of linear scalar equations.

3. Pandas

a. Write a python program to implement Pandas Series with labels, dictionary and Numpy
b. Write a program to creating a Pandas DataFrame using dictionary and two dimensional array.
c. Write a program which make use of following Pandas methods
d. i) describe() ii) head() iii) tail()
e. Perform insert, delete row operations on data frame.

4. Pandas Library: Visualization

a. Write a program which use pandas inbuilt visualization to plot following graphs:
   i. Bar plots
   ii. Histograms
   iii. Line plots
   iv. Scatter plots
b. Write a program to demonstrate use of groupby() method.
c. Write a program to demonstrate pandas Merging, Joining and Concatenating
d. Creating data frames from csv and excel files.

Text Books:

1. Core Python Programming Dr. R Nageswara Rao Dreamtech publications.
2. Problem solving and python programming fundametals and application: Numpy, Pandas and Matplotlib. HarshaBhasin.

Semester	**V to VII SEM**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	3	0	0	3	V20 CSTOE03
Name of the Course	**Operating Systems (Open Elective)**					
Branch	Common to CIVIL,MECH,EEE,ECE,ECT,CAI & AIM					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Describe Operating System Services and System Calls. **(K2)**
CO2: Illustrate Process Management Concepts and CPU Scheduling Algorithms. **(K3)**
CO3: Demonstrate Process Synchronization primitives and Process Deadlocks. **(K3)**
CO4: Illustrate Memory Management Techniques and Page Replacement Algorithms. **(K3)**
CO5: Describe File System Concepts and Mass Storage Structures. **(K2)**

UNIT-I: Introduction: Operating-System Structure, Operating-System Services, User and Operating System Interface, System Calls, Types of System Calls.

UNIT-II: Process Management: Process Concept, Process Scheduling, Operations on Processes, Inter process Communication. **Threads:** Overview, Multithreading Models
CPU Scheduling: Basic Concepts**,** Scheduling Criteria**,** Scheduling Algorithms.

UNIT-III: Process Synchronization: The Critical-Section Problem, Peterson's Solution, Synchronization Hardware, Mutex Locks, Semaphores, Classic Problems of Synchronization, Monitors. **Deadlocks:** System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock.

UNIT-IV: Memory Management: Main Memory: Swapping, Contiguous Memory Allocation, Segmentation, Paging, Structure of the Page Table.
Virtual Memory: Introduction, Demand Paging, Page Replacement Algorithms, Allocation of Frames, Thrashing.

UNIT-V: Storage Management: Overview of Mass-Storage Structure, Disk Scheduling, File Concept, Access Methods, Directory and Disk Structure, File-System Mounting, File Allocation Methods.

Text Book:

1. Operating System Concepts**,** Abraham Silberschatz, ,Peter Baer Galvin,Greg Gagne, 9th Edition, John Wiley and Sons Inc., 2012.

Reference Books:

1. Operating Systems – Internals and Design Principles, William Stallings, 7th Edition, Prentice Hall, 2012 .
2. Modern Operating Systems, Andrew S. Tanenbaum, Third Edition, Addison Wesley, 2007.

Semester	**V to VII SEM**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	3	0	0	3	V20CSTOE04
Name of the Course	**Software Engineering (Open Elective)**					
Branch	Common to CIVIL,MECH,EEE,ECE,ECT,CAI & AIM					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Demonstrate the Software Development life cycle Models. **(K3)**
CO2: Illustrate the Requirements engineering process and SRS document. **(K3)**
CO3: Develop the Software Architecture and Design Modeling. **(K3)**
CO4: Apply the Coding & Testing techniques and Risk management strategies. **(K3)**
CO5: Describe Project estimation techniques and Quality Management& Metrics. **(K2)**

UNIT-I: Software and Software Engineering: The Nature of Software, Software Engineering, Software Process, Software Engineering Practice, Software Myths. **Software process models:** Waterfall model, Prototyping, Iterative development, Unified process, RAD model, Spiral model, and agile process.

UNIT-II: Software Requirements: Functional and non-functional requirements, User requirements, System requirements, Interface specification, SRS document. **Requirements engineering process:** Feasibility studies, Requirements elicitation and analysis, Requirements validation, Requirements management.

UNIT-III: Software Architecture: Role of software architecture, Architecture views, components and connector view, Cohesion and Coupling, documenting architecture design. **Design:** Design concepts, Function-oriented design, object-oriented design, UML diagrams, and Data flow diagram.

UNIT-IV: Coding and Testing: Programming principles and guidelines, incrementally developing code. Testing concepts, testing process, Black-box & White-box testing.**Risk management**: Reactive vs. Proactive Risk strategies, Software risks, Risk identification, Risk projection, Risk refinement, RMMM Plan.

UNIT-V: Software Project Estimation& Maintenance: Decomposition techniques, Empirical Estimation Models, Maintenance Process, Reengineering, Configuration Management. Metrics for Products & Quality Management: Software Measurement, Metrics for software quality, Quality concepts, Software Reviews, Formal technical reviews, SEI-CMM Model, Six Sigma and ISO 9000 quality standards.

Text Books:

1. Software Engineering, A practitioner's Approach- Roger S.Pressman, 7th Edition, McGrawHill International Edition.
2. Software Engineering- Ian Sommerville, 9th Edition, Pearson education.Software Engineering, A Precise approach, PankajJalote, Wiley

Reference Books:

1. CMMI and Six Sigma: Partners in Process Improvement, Jeannine M. Siviy,M. Lynn Penn, Robert W. Stoddard, 1st edition, Addison Wesley;
2. SoftwareEngineeringprinciplesandpractice,WSJawadekar, 3rdEdition,TMH.

Semester	**V to VII SEM**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	0	0	6	3	V20CSTOE04
Name of the Course	**Object-Oriented Programming through Java Lab (Open Elective)**					
Branch	Common to CIVIL,MECH,EEE,ECE,ECT,CAI & AIM					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Use code editors and JDK tools to write, compile, and run Java programs. **(K3)**
CO2: Use control statements and arrays while programming. **(K3)**
CO3: Develop programs using classes and objects. **(K3)**
CO4: Use inheritance, interfaces and packages while developing programs in Java. **(K3)**
CO5: Develop exception-handling and multithreaded programs. **(K3)**

Syllabus:

CYCLE–I:Overview of Object-oriented Programming: Introduction to Object-oriented Programming, Principles of Object-oriented Programming Languages, and Applications of OOP.

Introduction to Java: History of Java, Java Features, Java Virtual Machine, Java Program Structure, Literals, Identifiers, Primitive Data types, Variables, Operators and Expressions, Operator Precedence and Associativity, Type Conversion and Casting.

Exercises

a) Develop a Java program to display the default values of all primitive data types of Java.
b) Construct a Java program that calculates the area of a triangle, given the lengths of all three sides.

Area =$\sqrt{(S\,(S-a)\,(S-b)\,(S-c))}$, where S = (a+b+c)/2.

CYCLE– II: Control Statements: Conditional Statements - if, switch; Iteration Statements - while, do-while, for, for-each version of for; Jump Statements - break, continue, return.

Arrays: Introduction to Arrays, Array Declaration and Initialization, One-Dimensional Arrays, Multi-Dimensional Arrays, Basic String Handling.

Exercises

a) Develop a Java program that displays
   i) The roots of a quadratic equation $ax^2+bx+c=0$
   ii) The nature of roots by calculating the discriminate D.
b) N bikers compete in a race such that they drive at a constant speed, which may or may not be the same as the other. To qualify the race, the speed of a racer must be more than the average speed of all N racers. Take as input, the speed of each racer and print back the speed of qualifying racers.
c) Develop a Java program that displays the name of the day, based on the value of day, using the switch statement.
d) Develop a Java program to search for an element in a given list of elements using Linear Search.
e) Develop a Java program to perform multiplication of two matrices.
f) Develop a Java program using StringBuffer to perform various operations on a string.

CYCLE– III: Introduction to Classes and Objects: General Form of a Class, Methods, Declaring Objects using new, Constructors, this Keyword, Understanding static, Method and Constructor Overloading, Using Command-Line Arguments, Garbage Collection.

Exercises

a) Construct a Java program to demonstrate class mechanism - Create a class that contains variables, methods, constructors and invoke those methods inside main().
b) Develop a Java program demonstrating the use of static variables, methods.
c) Develop a Java program demonstrating the use of this keyword.
d) Develop a Java program that implements method overloading.
e) Develop a Java program that implements constructor overloading.
f) Develop a Java program demonstrating the use of command-line arguments.

CYCLE– IV: Inheritance: Access Control, Introduction to Inheritance, Types of Inheritance, Using super, Method Overriding and Dynamic Method Dispatch, Using final, Abstract Classes.
Interfaces: Defining and Implementing Interfaces. **Packages:** Creating Packages, Importing Packages, Importance of CLASSPATH.

Exercises

a) Construct a Java program to demonstrate single inheritance.
b) Construct a Java program to demonstrate multi-level inheritance.
c) Construct a Java program that illustrates the use of super.
d) Develop a Java program that illustrates runtime polymorphism.
e) Develop a Java program that uses an abstract class to find areas of different shapes.
f) Develop a Java program using interfaces. In addition, use interfaces to achieve multiple inheritance.
g) Construct a Java program that creates a user-defined package. Use the package by importing it in another Java program.

CYCLE– V: Exception Handling: Exception-Handling Fundamentals, Using try and catch, Using throw, Using throws and finally, User-defined Exceptions.

Exercises

a) Develop a Java program to demonstrate exception-handling mechanism using try/catch. Use multiple catch clauses.
b) Construct a Java program for illustrating the use of throw.
c) Construct a Java program for illustrating the use of finally.
d) Construct a java program for demonstrating the creation and use of user-defined exceptions.

CYCLE– VI: Multithreading: Introduction to Multithreading, Creation of Threads, Thread Life Cycle, isAlive() and join(), Thread Synchronization, and Interthread Communication.

Exercises

a) Construct a Java program that creates threads by extending Thread class. The first thread displays "Good Morning" every 1 second, the second thread displays "Hello" every 2 seconds and thethird displays "Welcome" every 3 seconds.
b) Use Runnable to develop a Java program for the above problem.
c) Construct a java program illustrating isAlive() and join().
d) Develop a Java program to solve producer consumer problem using thread synchronization.

Text Books:

1. Java: The Complete Reference; 8th edition; Herbert Schildt; TMH.
2. Programming in Java;2nd edition; SachinMalhotra, SaurabhChoudhary; Oxford University Press.
3. Core JAVA, An Integrated Approach; Dr. R. Nageswara Rao; Dreamtech Press.

Semester	**V to VII SEM**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	3	0	0	3	V20CSTOE06
Name of the Course	**Computer Graphics (Open Elective)**					
Branch	Common to CIVIL,MECH,EEE,ECE,ECT,CAI & AIM					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Discuss the applications of computer graphics and learn basic algorithms. **(K2)**
CO2: Discuss the concepts of 2D graphics along with transformation techniques. **(K2)**
CO3: Demonstrate3D graphics and 3D object representation. **(K3)**
CO4: Discuss different visible surface detection methods and color models. **(K2)**
CO5: Illustrate different animation sequences . **(K2)**

UNIT-I: Introduction: Application of Computer Graphics, raster scan systems, random scan systems, raster scan display processors. Output Primitives : Points and lines, line drawing algorithms(Bresenham's and DDA Line derivations and algorithms), mid-point circle algorithms. **Filled area primitives:** Boundary-fill and flood-fill algorithms.

UNIT-II: 2-D geometrical transforms: Translation, scaling, rotation, reflection and shear transformations, and homogeneous coordinates, composite transforms.**2-D viewing:** The viewing pipeline, viewing coordinate reference frame, window to view-port coordinate transformation, viewing functions, Cohen-Sutherland line clipping, Sutherland–Hodgeman polygon clipping algorithm.

UNIT-III: 3-D Geometric transformations: Translation, rotation, scaling, reflection and shear transformations, composite transformations. **3-D object representation:** Polygon surfaces, quadric surfaces, spline representation, Bezier curve and B-Spline curves.

UNIT-IV: Visible surface detection methods: Classification, back-face detection, depth-buffer, scan-line, BSP tree methods, area sub-division. **Color Models** – RGB, YIQ, CMY, HSV.

UNIT-V: Computer Animation: Design of animation sequence, general computer animation functions, raster animation, computer animation languages, key frame systems**,** motion specifications.

Text Books:

1. Computer Graphics C version, Donald Hearn, M.Pauline Baker,Pearson
2. Computer Graphics, Schaum's outlines", Zhigandxiang,RoyPlastock, 2nd Edition,TataMc-Graw HillEdition.

Reference Books:

1. Computer Graphics Principles & practice, 2/e, Foley, VanDam, Feiner, Hughes,Pearson
2. Computer Graphics, Peter, Shirley, CENGAGE
3. Principles of Interactive Computer Graphics, Neuman ,Sproul,TMH.

Semester	**V to VII SEM**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	3	0	0	3	V20CSTOE07
Name of the Course	**Software Testing Methodologies (Open Elective)**					
Branch	Common to CIVIL,MECH,EEE,ECE,ECT,CAI & AIM					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Describe Software testing objectives and methodology. **(K2)**
CO2: Apply various Software testing techniques. **(K3)**
CO3: Discuss Static testing techniques for software testing. **(K2)**
CO4: Distinguish Software testing and debugging process. **(K2)**
CO5: Explain modern Software testing tools to Support software testing. **(K2)**

UNIT-I: Introduction to Software Testing: Evolution of software Testing, Myths and Facts, Goals of software Testing, Definitions of Testing, Model for Software Testing, Software Testing Terminology, Software Testing Life Cycle.

UNIT-II: Verification and Validation: Verification & Validation Activities, Verification, Verification of Requirements, Verification of High level and low level designs, How to verify code, Validation. **Dynamic Testing I:** Black Box testing techniques: Boundary Value Analysis, Equivalence Class Testing, Decision Table based Testing,

UNIT-III: Dynamic Testing II: White-Box Testing: Need of White-Box Testing, Logic coverage criteria, Basis path testing, Loop testing. **Static Testing:** Inspections, Structured Walkthroughs, Technical reviews.

UNIT-VI: Regression Testing: Progressive Vs Regressive Testing, Regression testability, Objectives of regression testing, When is Regression Testing done? Regression Testing Types, Regression testing techniques. **Debugging:** Debugging process, Techniques, correcting bugs.

UNIT-V: Software Quality Management: Software quality concept, Quality control and Quality Assurance, Software Quality metrics. **Automation and Testing Tools:** Need for automation, categorization of Testing tools, selection of testing tools, Overview of some commercial testing tools.

Text Books:

1. Software Testing, Principles and Practices, Naresh Chauhan, 9th Edition, Oxford Publisher.

Reference Books:

1. Software testing techniques - Boris Beizer, 2nd Edition, Dreamtech publisher.
2. Foundations of Software testing, Aditya P Mathur, 2nd ed, Pearson.
3. Software Testing- Yogesh Singh, CAMBRIDGE.

Semester	**V to VII SEM**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	0	0	6	3	V20CSTOE08
Name of the Course	**Linux Shell Scripting Lab (Open Elective)**					
Branch	Common to CIVIL,MECH,EEE,ECE,ECT,CAI & AIM					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Demonstrate the basic knowledge of Linux commands and utilities by using Linux shell environment. **(K3)**

CO2: Experiment with the Concept of shell Programming on Files and Directories. **(K3)**

CO3: Experiment with the Concept of shell Programming on File Permissions. **(K3)**

CO4: Experiment with the Concept of shell Programming on Conditional Statements. **(K3)**

CO5: Experiment with the Concept of shell Programming on Looping Statements. **(K3)**

LIST OF EXPERIMENTS

1. Experiment the following Unix Commands:
    a) **General Purpose Utilities:** cal, date,man,who.
    b) **Directory Handling Commands:** pwd,cd,mkdir,rmdir.
    c) **File Handling Utilities:** cat,cp,ls,rm,nl,wc
    d) **Displaying Commands:** head, tail
    e) **Filters:** cmp,comm.,diff,sort,uniq
    f) **Disk Utilities:** du,df
2. Develop a Shell Program to Display all the words which are entered as command line arguments.
3. Develop a shell script that Changes Permissions of files in PWD as rwx for users.
4. Develop a shell script to print the list of all sub directories in the current directory.
5. Develop a Shell Program which receives any year from the keyboard and determine whether the year is leap year or not. If no argument is supplied the current year should be assumed.
6. Develop a shell script which takes two file names as arguments-If their contents are same then delete the second file.
7. Develop a shell script to print the given number in the reversed order.
8. Develop a shell script to print first 25 Fibbonacci numbers.
9. Develop a shell script to print the Prime numbers between the specified range.
10. Develop a shell script to delete all lines containing the word 'unix' in the files supplied as arguments.
11. Develop a shell script Menu driven program which has the following options.
    i) contents of /etc/passwd ii) list of users who have currently logged in.
    iii) present working directory. iv) exit.

Text Books:

1. UNIX and Shell Programming: A Textbook, Behrouz A. Forouzan | Richard F. Gilberg, Cengage Learning.
2. UNIX: Concepts and Applications, Sumithaba Das, 4th Edition, Tata McGrawHill.
3. Unix & Shell Programming, M.G.Venkatesh Murthy, Pearson Education.
4. UNIX shells by example, 4th Edition Ellie Quigley, Pearson Education.

Semester	**V to VII SEM**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	3	0	0	3	V20CSTOE09
Name of the Course	**Computer Networks (Open Elective)**					
Branch	Common to CIVIL,MECH,EEE,ECE,ECT,CAI & AIM					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Discuss fundamentals of network concepts and Reference Models. **(K2)**
CO2: Discuss Communication media and switching techniques. **(K2)**
CO3: Demonstrate Error control and Data link layer protocols. **(K3)**
CO4: Apply Routing algorithms and congestion control algorithms. **(K3)**
CO5: Discuss Transport layer protocols and Application layer protocols. **(K2)**

UNIT-I: Introduction: Reference models: The OSI Reference Model- the TCP/IP Reference Model, Examples of Networks: Novell Networks, Arpanet, Internet, Network Topologies WAN, LAN, MAN.

UNIT-II: Physical Layer: Transmission Media, Multiplexing: FDM, WDM and TDM- LAN Technologies, introduction to switching: Circuit Switched Networks, Datagram Networks, and Virtual Circuit Networks.

UNIT–III: Data link layer: Design issues, Framing, Flow control, error control, error detection - Parity bit, CRC, Checksum, error correction- Hamming code. MAC: ALOHA, CSMA. Elementary Data Link Layer protocols: simplex protocol, Simplex stop and wait, Simplex protocol for Noisy Channel. Sliding window protocol: One bit, Go back N, Selective repeat-Stop and wait protocol, HDLC, point to point protocol (PPP).Piggybacking.

UNIT-IV: Network Layer : Network layer design issues- Algorithm shortest path routing, Flooding, Distance Vector Routing, Link State Routing, Hierarchical routing, Broad cast, Multi cast Routing algorithms-Congestion control and algorithms, Internet Protocol (IP) Addresses, Subnet masking. Quality of Service, QoS improving techniques: Leaky Bucket and Token Bucket algorithm.

UNIT–V:Transport Layer: Services, Primitives and sockets, Elements of transport protocols, Internet Transport protocols(TCP,UDP,RPC,RTTP/RTP,RTCP) Segment headers, Primitives, Control, Congestion control.
Application layer: DNS, SMTP, POP,FTP HTTP Presentation formatting. Network security: Cryptography, DES Public key and RSA private key cryptography Algorithms.

Text Books:

1. Computer Networks — Andrew S Tanenbaum, 4th Edition. Pearson Education/PHI.
2. Data Communications and Networks – Behrouz A. Forouzan.Third Edition TMH.

Reference Books:

1. An Engineering Approach to Computer Networks-S.Keshav, 2nd Edition, Pearson Education.
2. Understanding communications and Networks, 3rd Edition, W.A. Shay, Thomson.

Semester	**V to VII SEM**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	3	0	0	3	V20CSTOE10
Name of the Course	Cryptography and Network Security **(Open Elective)**					
Branch	Common to CIVIL,MECH,EEE,ECE,ECT,CAI & AIM					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Discuss fundamentals and mathematical support of Cryptography and Network Security. (**K2)**

CO2: Discuss symmetric and asymmetric cryptosystems. (**K2)**
CO3 : Discuss about HASH functions & Digital Signatures to provide authentication and integrity. **(K2)**
CO4: Demonstrate various methods of Mutual trust and mail security**.** **(K3)**
CO5: Review the Network& Internet Security Scenarios. **(K2)**

UNIT-I: Overview: Security attacks, Services, Mechanisms, A model for network security, Symmetric cipher model. **Classical encryption techniques**: Substitution Techniques, Transposition Techniques. **Number Theory**: Prime numbers, Fermat's theorem, Euler's Theorem, the Chinese Remainder Theorem.

UNIT-II: Block Cipher: Principles, DES, Strength of DES, AES, Block cipher Modes of Operations.
Public Key Cryptography: Principles, Public Key Crypto system, RSA Algorithm, Diffie Hellman Key Exchange.

UNIT-III:Cryptographic Hash Functions: Application of Cryptographic Hash Functions, Requirements & Security, SHA-512, Message Authentication Functions, Requirements, HMAC.
Digital Signatures: Properties, Attacks and Forgeries, Requirements, Digital Signature Standards, NIST Digital Signature Algorithm.

UNIT-IV:Key Management and Distribution: Symmetric Key Distribution Using Symmetric Encryption, Asymmetric Key Distribution Using Symmetric Encryption, Distribution of Public Keys, X.509 Certificates. **User Authentication**: Remote User Authentication Principles, Kerberos. **Electronic Mail Security:** Pretty Good Privacy (PGP) And S/MIME.

UNIT-V: IP Security: Two modes, two security protocols Authentication Header, Encapsulating Security Payload. **Transport Level Security:** Secure Socket Layer (SSL) and Transport Layer Security (TLS). **HTTPS:** Connection Initiation Connection Closure.

Text Books:

1. William Stallings, "Cryptography and Network Security, Principles and Practices", Pearson Education, Sixth Edition.
2. Cryptography and Network Security, Behrouz A Forouzan, Debdeep Mukhopadhyay, (3e) Mc Graw Hill.

Reference Books:

1. Charlie Kaufman, Radia Perlman and Mike Speciner, "Network Security – Private Communication in a Public World" Pearson/PHI.

Semester	**V to VII SEM**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	0	0	6	3	V20CSTOE11
Name of the Course	**Database Management System Lab (Open Elective)**					
Branch	Common to CIVIL,MECH,EEE,ECE,ECT,CAI & AIM					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1:	Construct SQL queries to perform different database operations.	(K3)
CO2:	Experiment with various constraints and Database Indexing Techniques.	(K3)
CO3:	Construct PL/SQL Cursors and Exceptions.	(K3)
CO4:	Develop PL/SQL Functions and Procedures.	(K3)
CO5:	Develop PL/SQL Packages.	(K3)

LIST OF EXPERIMENTS

1. Construct SQL queries to facilitate acquaintance of Built-In Functions, String Functions, Numeric Functions, Date Functions and Conversion Functions.
2. Construct SQL queries using Operators.
3. Construct SQL queries to Retrieve and Change Data: Select, Insert, Delete and Update
4. Construct SQL queries using Group By, Order By and Having Clauses.
5. Construct SQL queries on Controlling data: commit, rollback and savepoint
6. Construct report using SQL*PLUS
7. Construct SQL queries for Creating, Dropping and Altering Tables, Views and Constraints
8. Construct SQL queries on Joins and Correlated Subqueries
9. Demonstrate Index, Sequence and Synonym.

PL/SQL

11. DemonstrateBasicVariables,AnchoredDeclarations,andUsageofAssignmentOperation Using PL SQL block
12. Demonstrate Bind and Substitution Variables using PL SQL block
13. Demonstrate Control Structures in PL SQL
14. Demonstrate Cursors, Exception and Composite Data Types in PL SQL.
15. Demonstrate Procedures, Functions, and Packages in PLSQL.

Textbooks:

1. OracleDatabase11gTheCompleteReferencebyOraclePress,Kevin Loney
2. DatabaseSystemsUsingOracle,NileshShah,2ndEdition, PHI.
3. IntroductiontoSQL,RickFVanderLans, 4thEdition, Pearson Education.

Reference Books:

1. OraclePL/SQLInteractiveWorkbook,B.RosenzweigandE.Silvestrova,2ndEdition, Pearsoneducation.
2. SQL&PL/SQLforOracle10 g,BlackBook, Dr.P.S.Deshpande, DreamTech.

Semester	**V to VII SEM**	**L**	**T**	**P**	**C**	**COURSE CODE**
Regulation	V20	3	0	0	3	V20CSTOE12
Name of the Course	Human Computer Interaction **(Open Elective)**					
Branch	Common to CIVIL,MECH,EEE,ECE,ECT,CAI & AIM					

Syllabus Details

Course Outcomes: After Successful completion of the Course, the student will be able to:

CO1: Describe the principles and characteristics of GUI. **(K2)**
CO2:Describe how a computer system may be modified to include human diversity. **(K2)**
CO3: Select an effective style and screen design for a specific business application. **(K2)**
CO4: Discuss System Menus & Navigation Schemes**.** **(K2)**
CO5: Select Device and Screen based controls. **(K2)**

UNIT I:The User Interface: Introduction, Importance of the User Interface, Importance and benefits of Good Design, Characteristics of Graphical and Web User Interface Graphical User Interface, popularity of graphics, concepts of Direct Manipulation, Graphical System advantage and disadvantage, Characteristics of GUI, Characteristics of Web Interface, Principles of User Interface Design.

UNITII:TheUserInterfaceDesignProcess:ObstaclesandPitfallinthedevelopmentProcess,Usability, The Design Team, Human Interaction with Computers, Important Human Characteristics in Design, Human Consideration in Design, Human Interaction Speeds, Performance versus Preference, Methods for Gaining and Understanding of Users.

UNIT III: Understanding Business Functions: Business Definitions& Requirement analysis, Determining Business Functions. **Principles of Good Screen Design:** Human considerations in screen Design, interface design goals, screen meaning and purpose, Technological considerations in Interface Design.

UNIT IV: System Menus and Navigation Schemes: Structure, Functions, Context, Formatting, PhrasingandSelecting,NavigatingofMenus,KindsofGraphicalMenusWindowsInterface:Windowscharacteristic,ComponentsofWindow,WindowsPresentationStyles,TypesofWindows,WindowManagement,

UNITV: Device and Screen-Based Control: Device based controls, Operable Controls,Textentry/read-Only Controls, Section Controls, Combining Entry/Selection Controls Presentation Controls, Selecting proper controls.

Text Books:

1. "The Essential Guide to User Interface Design", Wilbert O. Galitz, 2nd edition,2002,Wiley India Edition.
2. Prece, Rogers, "Sharps Interaction Design", Wiley India.
3. "Designing the user interfaces". Ben Shneidermann 3rd Edition, Pearson Education Asia.

.**Reference Books:**

1. "User Interface Design" , SorenLauesen, Pearson Education
2. "Essentials of Interaction Design", Alan Cooper, Robert Riemann, David Cronin, Wiley
3. "Human Computer Interaction", Alan Dix, Janet Fincay, GreGoryd, Abowd, Russell, Bealg, Pearson Education.

Honors Degree

List of Courses for B.Tech(Hons) in Computer Science

S.No.	Course Name	Number of Weeks	Credits	
1.	Data Science for Engineers	8	2	**Students have to acquire a minimum of 16 credits by completing MOOC/NPTEL Courses from this Pool**
2.	Big Data Computing	8	2	
3.	Introduction to Haskell Programming	8	2	
4.	Distributed Systems	8	2	
5.	Computer Graphics	8	2	
6.	Data Analytics With Python	12	3	
7.	Learning Analytics Tools	12	3	
8.	Introduction to Artificial Intelligence	12	3	
9.	Deep Learning	12	3	
10.	Natural Language Processing	12	3	
11.	Reinforcement Learning	12	3	
12.	Computer Vision	12	3	
13.	Deep Learning for Computer Vision	12	3	
14.	Cloud Computing	12	3	
15.	Advanced Distributed systems	12	3	
16.	Software Testing Methodologies	12	3	
17.	Privacy and Security in Online Social Media	12	3	
18.	Blockchain Architecture Design And Use Cases	12	3	
19.	Introduction To Internet Of Things	12	3	
20.	Introduction to Biomedical Imaging Systems	12	3	
21.	Digital Forensics	12	4	
22.	Web based Technologies and Multimedia Applications	12	4	
23.	Introduction to Information Technology	12	4	
Project Work			4	4
Total				**20 Credits**

NOTE: However the list is not exhaustive. Before registering the Course take the approval from HOD.

Minor Degree in Computer Science and Engineering

List of Courses for B.Tech(Minors) - Artificial Intelligence

S.No.	Course Name	Number of Weeks	Credits	
1	Data Science for Engineers	8	2	**Students have to acquire a minimum of 16 credits by completing MOOC/NPTEL Courses from this Pool**
2*	Introduction to Machine Learning (IITKGP)	8	2	
	Introduction to Machine Learning (IITM)	12	3	
3	Machine Learning for Earth System Sciences	8	2	
4	Scalable Data Science	8	2	
5*	Business Analytics and Text Mining Modeling using Python	8	2	
	Data Analytics With Python	12	3	
6	Introduction to Artificial Intelligence	12	3	
7	Essential Mathematics for Machine Learning	12	3	
8	Deep Learning	12	3	
9	Computer Vision	12	3	
10	Deep Learning for Computer Vision	12	3	
11	Natural Language Processing	12	3	
Project Work			4	4
Total				**20 Credits**

***Students can opt only one course from this set.**

NOTE: However the list is not exhaustive. Before registering the Course take the approval from HOD.

List of Courses for B.Tech(Minors) - Networks and Cyber Security

S.No.	Course Name	Number of Weeks	Credits	
1	Computer Networks and Internet Protocol	12	3	**Students have to acquire a minimum of 16 credits by completing MOOC/NPTEL Courses from this Pool**
2*	Cyber Security **(NPTEL)**	12	3	
	Cyber Security **(CEC)**	15	4	
	Cyber Security Tools, Techniques and Counter measures **(IGNOU)**	12	4	
3	Cryptography and Network Security	12	3	
4	Information Security and Cyber Forensics (CEC)	12	4	
5	Blockchain and its Applications	12	3	
6	Cloud Computing	12	3	
7	Introduction to Internet of Things	12	3	
8	Hardware Security	12	3	
9*	Digital Forensics **(IGNOU)**	12	4	
	Digital Forensics **(CEC)**	16	4	
Project Work			4	4
Total				**20 Credits**

***Students can opt only one course from this set.**

NOTE: However the list is not exhaustive. Before registering the Course take the approval from HOD.