SHRI RAMDEOBABA COLLEGE OF ENGINEERING AND MANAGEMENT, NAGPUR – 440013

An Autonomous College affiliated to
Rashtrasant Tukadoji Maharaj Nagpur University,
Nagpur, Maharashtra (INDIA)

PROGRAMME SCHEME & SYLLABI
2020 – 2021

B. E. (COMPUTER SCIENCE & ENGINEERING)

Shri Ramdeobaba College of Engineering and Management, Nagpur

Published By

Dr. R. S. Pande

Principal
Shri Ramdeobaba College of Engineering & Management
Ph. : 0712-2580011   Fax : 0712 - 2583237
ISO 9001 : 2015 CERTIFIED ORGANISATION

About the Department

The department was established in 2002 and is well equipped with latest and state-of-the-art hardware and software/ The department hosts 300 computers with internet facility. The 24x7 network managed with Cyberoam UTM firewall, and CISCO router offers intranet and internet connectivity. The computer laborites have high-end servers of IBM and WIPRO along with industry standard software, viz., Oracle, NetSim, Wireshark, AIX, Robotics Platform, IOT kit and MSDN. The department promotes computing through Open Source technologies. The department also has a NVIDIA DX-2 Server with 128 GB GPU capable of accelerating the execution capabilities for Machine learning and Deep learning Algorithms.

The department has been instrumental in condition and management of Smart India Hackathon since 2017. This year 04 teams from the department are selected for Grand Finale of Smart India Hackathon-2020 (Software Edition-Online Platform) held on 01-03 August, 2020. The team of Seventh Semester CSE, students attempted a solution named "ReMedize" to the problem- "Primary health care in remote areas using Cloud medicine and Diagnostic" of Government of Uttarakhand bagging a prize money of INR One Lakh.

For the Third Smart India Hackathon, SIH-2019 held on March 02-03, 2019, Dr. M. B. Chandak, HOD, CSE was SPoC nodal Centre for the Grand Finale of SIH 2019 ar RCOEM Nodal Centre. 10 studetn teams of RCOEM participated at the Grand Finale of SIH-2019. All the 03 winning teams successfully solved the problems of "complicated" category to bag cash prize of INR 75,000 each under the able guidance of Professors for CSE Department.

Students of CSE are inclined to competitive coding and are actively contributing through Code Chef, Hacker Rank, Hacker Earth, Geeks Coding Challenge and others. In ICPC-2019 the world renowned prestigious international Collegiate Programming Contest by ACM - CSE, RCOEM was able to participate for Onsite Contests at all four Asia Regional Contests organized at sites- IIT Kharagpur, Kanpur-Kanpur, Gwalior-Pune & Amritapuri. Our teams have been regularly participating at ICPC Regional Contests in past years.

A grant of INR 45 lakh has been received from Department of Science and Technology, Govt. of India under Fund for Improvement of S&T Infrastructure (FIST- Level0).

Departmental Vision

To continually improve the education environment, in order to develop graduates with strong academic and technical background needed to achieve distinction in the discipline. The excellence is expected in various domains like workforce, higher studies or lifelong learning.

To strengthen links between industry through partnership and collaborative development works.

Department Mission

To develop strong foundation of theory and practices of computer science amongst the students to enable them to develop into knowledgeable, responsible professionals, lifelong learners and implement the latest computing technologies for the betterment of the society.

Programme Educational Objectives

I. **To prepare graduates to apply the broad set of techniques, tools, and skills from science, mathematics and engineering required to solve problems in Computer Science and Engineering.**

The field of Computer Science & Engineering is a fast evolving field and caters to multiple disciplines. The focus is to imbibe necessary skill set amongst the students and develop competencies to solve basic computer science & engineering problem.

II. **To prepare graduates to address practices in computer science and engineering using software development life cycle principles.**

The department aims to develop good analytical and designing skills amongst students, while emphasizing on theoretical and practical aspects of computer science.

III. **To provide adequate training & opportunities to work as teams in multidisciplinary projects.**

The department aims at encouraging team spirit through projects which are multi- disciplinary in nature.

IV. **To prepare the graduates to exhibit professionalism, communication skills, ethical attitude, and practice their profession with high regard to legal and ethical responsibilities.**

The department recognizes the need for effective communication in students and strives to enhance this aspect. The department feels that apart from curricular studies, it is necessary to impart good moral values in the students so that they are aware of their social responsibilities.

V. **To prepare graduates for engaging in life-long learning, such as post graduate study & certification courses.**

The department encourages the students for higher studies and certification courses to keep track with the pace of technology.

Program outcomes (POs)

1. Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

4. Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

5. Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

6. The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

7. Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

8. Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

9. Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

10. Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

11. Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

12. Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change.

Programme Specific Outcomes (PSOs)

1. Foundation of Computer System: Ability to understand fundamental concepts of computer science & engineering, operating system, networking & data organization systems, hardware & software aspects of computing,

2. Software development Ability: Ability to understand the software development life cycle. Possess professional skills and knowledge of software design process. Familiarity and algorithmic competence with a broad range of programming languages and open source platforms.

3. Research Ability: Ability to apply knowledge base to identify research gaps in various domains, model real world problems, solve computational tasks, to provide solution for betterment of society with innovative ideas.

TEACHING SCHEME FOR FIRST YEAR (SEMESTER I & II) BACHALOR OF ENGINEERING

GROUP 1: SEMESTER-I/ GROUP 2: SEMESTER-II

Sr. No.	Code	Course	Hours/week			Credits	Maximum Marks			ESE Duration (Hours)
			L	T	P		Continual Assessment	End Sem Examination	Total	
1.	PHT156	Semiconductor Physics	3	1	0	4	40	60	100	03
2.	PHP156	Semiconductor Physics Lab	0	0	3	1.5	25	25	50	--
3.	MAT152/ MAT151	Differential Equations, Linear Algebra, Statistics & Probability / Calculus	3	0/1	0	3/4	40	60	100	03
4.	MAP151	Computational Mathematics Lab	0	0	2	1	25	25	50	--
5.	EET151	Basic Electrical Engineering	3	1	0	4	40	60	100	03
6.	EEP151	Basic Electrical Engineering Lab	0	0	2	1	25	25	50	--
7.	MET151	Engineering Graphics & Design	1	0	0	1	40	60	100	03
8.	MEP151	Engineering Graphics & Design Lab	0	0	4	2	50	50	100	--
9.	HUT152	Constitution of India	2	0	0	0	--	--	--	--
10	PEP151	Yoga/Sports	0	0	2	0	--	--	--	--
		Total	12	2/3	13	17.5/18.5			650	

GROUP 1: SEMESTER-II/ GROUP 2: SEMESTER-I

Sr. No.	Code	Course	Hours/week			Credits	Maximum Marks			ESE Duration (Hours)
			L	T	P		Continual Assessment	End Sem Examination	Total	
1.	CHT152	Chemistry	3	1	0	4	40	60	100	03
2.	CHP152	Chemistry Lab	0	0	3	1.5	25	25	50	--
3.	MAT151/ MAT152	Calculus / Differential Equations, Linear Algebra, Statistics & Probability	3	1/0	0	4/3	40	60	100	03
4.	CST151	Programming for Problem Solving	4	0	0	4	40	60	100	03
5.	CSP151	Programming for Problem Solving Lab	0	0	2	1	25	25	50	--
6.	IDT151	Creativity, Innovation & Design Thinking	1	0	0	1	20	30	50	1.5
7.	INT151	Workshop/Manufacturing Practices	1	0	0	1	20	30	50	1.5
8.	INP151	Workshop/Manufacturing Practices Lab	0	0	2	1	25	25	50	--
9.	HUT151	English	2	0	0	2	40	60	100	03
10.	HUP151	English Lab	0	0	2	1	25	25	50	--
		Total	14	2/1	9	20.5/19.5			700	

Program Scheme and Syllabi for B. E. (Computer Science & Engineering)

Scheme of Teaching & Examination of Bachelor of Engineering (Computer Science Engineering); Semester : III

Sr. No.	Category	Course code	Course Name	Hours/week			Credits	Maximum marks			ESE duration (Hrs)
				L	T	P		Continuous evaluation	End Sem Exam	Total	
1	PCC	CST251	Fundamentals of Digital Logic and Computer Architecture	4	0	0	4	40	60	100	3 Hrs
2	PCC	CSP251	Fundamentals of Digital Logic and Computer Architecture Lab	0	0	2	1	25	25	50	-
3	PCC	CST252	Data Structures & Algorithms	3	0	0	3	40	60	100	3 Hrs
4	PCC	CSP252	Data Structures & Algorithms Lab	0	0	4	2	25	25	50	- -
5	PCC	CSP253	Systems Lab-I	0	0	4	2	25	25	50	- -
6	BSC	MAT252	Linear Algebra and Statistics	2	1	0	3	40	60	100	3 Hrs
7	HSSM	HUT253	Business Communication	3	0	0	3	40	60	100	3 Hrs
8	HSSM	HUT257	Cyber Laws & Ethics in IT	2	0	0	2	40	60	100	3 Hrs
	Total			**14**	**1**	**10**	**20**	**275**	**375**	**650**	

Scheme of Teaching & Examination of Bachelor of Engineering (Computer Science Engineering); Semester : IV

Sr. No.	Category	Course code	Course Name	Hours/week			Credits	Maximum marks			ESE duration (Hrs)
				L	T	P		Continuous evaluation	End Sem Exam	Total	
1	PCC	CST254	Discrete Mathematics and Graph Theory	3	1	0	4	40	60	100	3 Hrs
2	PCC	CST255	Operating Systems	3	0	0	3	40	60	100	3 Hrs
3	PCC	CSP255	Operating Systems Lab	0	0	4	2	25	25	50	-
4	PCC	CST256	Object Oriented Programming	3	0	0	3	40	60	100	3 Hrs
5	PCC	CSP256	Object Oriented Programming Lab	0	0	2	1	25	25	50	-
6	PCC	CST257	Formal Languages & Automata Theory	3	0	0	3	40	60	100	3 Hrs
7	PCC	CST258	System Programming & Device Drivers	3	0	0	3	40	60	100	3 Hrs
8	PCC	CSP258	System Programming & Device Drivers Lab	0	0	2	1	25	25	50	-
9	PCC	CSP259	Systems Lab-II	0	0	4	2	40	60	100	-
10	OEC	CST299	Open Elective-I	3	0	0	3	40	60	100	3 Hrs
11	MC	CHT252	Environmental Sciences	2	-	-	0	-	-	-	-
				20	**1**	**12**	**25**	**355**	**495**	**850**	-

Department of Computer Science and Engineering

Scheme of Teaching & Examination of Bachelor of Engineering (Computer Science Engineering); Semester : V

Sr. No.	Category	Course code	Course Name	Hours/week			Credits	Maximum marks			ESE duration (Hrs)
				L	T	P		Continuous evaluation	End Sem Exam	Total	
1	PCC	CST351	Database Management Systems	3	0	0	3	40	60	100	3 Hrs
2	PCC	CSP351	Database Management Systems Lab	0	0	4	2	25	25	50	-
3	PCC	CST352	Design & Analysis of Algorithms	3	1	0	4	40	60	100	3 Hrs
4	PCC	CSP352	Design & Analysis of Algorithms Lab	0	0	2	1	25	25	50	-
5	PCC	CST353	Computer Networks	3	0	0	3	40	60	100	3 Hrs
6	PCC	CSP353	Computer Networks Lab	0	0	2	1	25	25	50	-
7	PCC	CSP354	Mobile Prog. Lab	0	0	4	2	25	25	50	-
8	OEC	CST398	Open Elective-II	3	0	0	3	40	60	100	3 Hrs
9	PEC	CST355	Elective-I	3	0	0	3	40	60	100	3 Hrs
10	MC	HUT353	Indian Traditional Knowledge	2	-	-	0	-	-	-	-
				17	**1**	**12**	**22**	**300**	**400**	**700**	**-**

Course Code	ELECTIVE – I
CST355-1	Computer Graphics
CST355-2	Embedded Systems
CST355-3	Information Theory & Coding
CST355-4	Design Patterns

Scheme of Teaching & Examination of Bachelor of Engineering (Computer Science Engineering); Semester : VI

Sr. No.	Category	Course code	Course Name	Hours/week			Credits	Maximum marks			ESE duration (Hrs)
				L	T	P		Continuous evaluation	End Sem Exam	Total	
1	PCC	CST356	Artificial Intelligence	3	0	0	3	40	60	100	3 Hrs
2	PCC	CSP356	Artificial Intelligence Lab	0	0	2	1	25	25	50	-
3	PCC	CST357	Software Engineering	3	0	0	3	40	60	100	3 Hrs
4	PCC	CSP357	Software Engineering Lab	0	0	2	1	25	25	50	-
5	PCC	CST358	Complier Design	3	0	0	3	40	60	100	3 Hrs
6	PCC	CSP358	Complier Design Lab	0	0	4	2	25	25	50	-
7	PEC	CST359	Elective-II	3	0	0	3	40	60	100	3 Hrs
8	OEC	CST399	Open Elective-III	3	0	0	3	40	60	100	3 Hrs
9	PR	CSP360	Project-1	0	0	6	3	25	25	50	-
10	PCC	CSP361	Comprehensive Viva	0	0	2	1	25	25	50	-
				15	0	16	23	325	425	750	-

Course Code	ELECTIVE – II
CST359-1	Advanced Algorithms
CST359-2	Distributed Systems
CST359-3	Digital Signal Processing
CST359-4	Data Warehousing & Mining

Department of Computer Science and Engineering

Scheme of Teaching & Examination of Bachelor of Engineering (Computer Science Engineering); Semester : VII

Sr. No.	Category	Course code	Course Name	Hours/week			Credits	Maximum marks			ESE duration (Hrs)
				L	T	P		Continuous evaluation	End Sem Exam	Total	
1	PCC	CST451	Elective-III	3	0	0	3	40	60	100	3 Hrs
2	PCC	CSP451	Elective-III Lab	0	0	2	1	25	25	50	-
3	PCC	CST452	Elective-IV	3	0	0	3	40	60	100	3 Hrs
4	PCC	CSP452	Elective-IV Lab	0	0	2	1	25	25	50	-
5	OEC	CST498	Open Elective-IV	3	0	0	3	40	60	100	3 Hrs
6	BSC	IDT451	Bio-informatics	2	1	0	3	40	60	100	3 Hrs
7	PR	CSP454	Project-2	0	0	12	6	50	50	100	-
8	PR	CSP455	Industry Internship Evaluation	0	0	2	0	-	-	-	-
				11	**1**	**18**	**20**	**260**	**340**	**600**	-

Course Code	ELECTIVE - III	Course Code	ELECTIVE - IV
CST451-1	Machine Learning	CST452-1	Digital Image & Video Processing
CST451-2	Web Intelligence and Big Data	CST452-2	Distributed and Parallel Database
CST451-3	Data Visualization & Analytics	CST452-3	Game Theory
CST451-4	Fundamentals of Augmented Reality	CST452-4	Cloud Computing

Scheme of Teaching & Examination of Bachelor of Engineering (Computer Science Engineering); Semester : VIII

Sr. No.	Category	Course code	Course Name	Hours/week			Credits	Maximum marks			ESE duration (Hrs)
				L	T	P		Continuous evaluation	End Sem Exam	Total	
1	PEC	CST456	Elective-V	3	0	0	3	40	60	100	3 Hrs
2	PEC	CST457	Elective-VI	3	0	0	3	40	60	100	3 Hrs
3	PR	CSP458	Project-3/Industry Internship	0	0	12	6	50	50	100	-
				6	0	12	12	130	170	300	-

Course Code	ELECTIVE - V	Course Code	ELECTIVE - VI
CST456-1	Neural Network & Deep Learning	CST457-1	Information Retrieval
CST456-2	Robotics: Perception & Estimation	CST457-2	Natural Language Processing
CST456-3	Multi Agent Intelligent Systems	CST457-3	Data Warehousing for Business Intelligence
CST456-4	Cryptography & Network Security	CST457-4	Internet of Things

Course Code	Open Elective	Course Name
CST299-1	Open Elective - I	Java Programming and UI Design Concepts
CST299-2	Open Elective - I	Design Thinking for Innovation
CST398-1	Open Elective – II	Python and Data Analysis
CST399-1	Open Elective – III	Recent Trends in Computing
CST498-1	Open Elective - IV	Data Analytics for Business Applications

Total Credits (III to VIII Semester): 122

Department of Computer Science and Engineering
Honors & Minor Curriculum Design

Honors Scheme

Sr. No.	Semester	Course code	Course Name	Hours per Week	Credits	Maximum marks			ESE Duration in (Hrs)
						Continuous Evaluation	End Sem Exam	Total	
1	IV	CSTH41	Programming for Advanced Computing	4	4	40	60	100	3
2	V	CSTH51	Pattern Recognition / MOOC	4	4	40	60	100	3
3	VI	CSTH61	Graph Mining / MOOC	4	4	40	60	100	3
4	VII	CSTH71	Statistical Machine Learning / MOOC	4	4	40	60	100	3
5	VIII	CSTH81	Big Data Analysis / MOOC	4	4	40	60	100	3

Note : All the courses mentioned above are provided by the PG department of CSE. If any of the course is not provided by the PG department of CSE, that course can be completed through MOOC or any other sources with prior permission from HOD, CSE.

Minor Scheme

Sr. No.	Semester	Course code	Course Name	Hours per Week	Credits	Maximum marks			ESE Duration in (Hrs)
						Continuous Evaluation	End Sem Exam	Total	
1	IV	CSTM41	Data Structures & Algorithms	4	4	40	60	100	3
2	V	CSTM51	Software Engineering & Project Management	4	4	40	60	100	3
3	VI	CSTM61	AI and Machine Learning	4	4	40	60	100	3
4	VII	CSTM71	Mobile Application Programming	4	4	40	60	100	3
5	VIII	CSTM81	Database Management System	4	4	40	60	100	3

Note :

1. If any of the subjects is offered by the parent department, then with the prior permission of HOD, CSE the student can opt for -

a. ONE/TWO Program Electives (for same/more credits) offered by CSE OR

b. MOOC courses (for same/more credits)

2. Students cannot opt for an open elective course of any departments which are aligned with the courses offered in Minor Scheme.

Syllabus for Semester I / II

(Electronics Engineering, Electronics and Communication Engineering, Computer Science Engineering, Information Technology, Computer Science Engineering (Data Science))

Course Code : PHT156 **Course : PHYSICS : Semiconductor Physics (Theory)**

Category : Basic Science Course

L: 3 Hrs. T: 1 Hrs. P: 0 Hrs. Per week **Total Credits : 4**

Course Objectives

1. To introduce ideas of quantum mechanics necessary to begin understanding semiconductor devices;
2. To familiarize prospective engineers with fundamental concepts of semiconductors and their interaction with light and resulting devices

Course Outcomes

After successful completion of the course students will

1. Have an elementary understanding of quantum behaviour of electrons in solids;
2. Have a grasp of band structure and its consequences for semiconductors;
3. Should be able to use band structure to explain effects of doping,on the properties of junctions between semiconductors and metals;
4. Have an elementary understanding of working of optoelectronic devices

Module 1: Quantum Mechanics Introduction

Wave-particle duality, Heisenberg uncertainty relations, the quantum state wave function and its probability interpretation, Schrodinger's equation, Energies and wave functions of a single electron in one-dimensional infinite potentials: formulae, function graphs, number of bound states, tunneling, One electron atom, periodic table, Quantum confinement effects in nanosystems

Module 2: Electronic Materials

Free electron theory, Extension of idea of energy level splitting in molecules to bonding in solids, Energy bands in solids, Kronig-Penny model (to better demonstrate origin of band gaps), Band gap based classification of electronic materials: metals, semiconductors, and insulators, E-k diagram, Direct and indirect bandgaps, Valence and conduction bands, Density of states, Fermi-Dirac statistics: Occupation probability of states, Fermi level, Effective mass.

Module 3: Intrinsic and Extrinsic Semiconductors

Intrinsic and extrinsic semiconductors, Dependence of Fermi level on carrier-concentration and temperature (equilibrium carrier statistics), Carrier transport: diffusion and drift

Module 4: Non-Equilibrium Semiconductors

Carrier generation and recombination, Continuity equation, Ambipolar transport equation, Quasi-Fermi Energy levels, Excess Carrier Lifetime, Qualitative introduction to recombination mechanisms, Shorkley - Read-Hall Recombination, Surface Recombination

Module 5: Junction Physics

p-n junction, Zero applied bias, forward bias, reverse bias, Metal-semiconductor junction, Shottky barrier, Ideal junction properties, Ohmic contacts, ideal non-rectifying barrier, tunneling barrier, Heterojunctions, Nanostructures, Energy band diagram, two dimensional electron gas

Module 6: Light - Semiconductors Interaction

Optical absorption in semiconductors, Light emitting diodes, Principles, Device Structures, Materials, High Intensity LEDs, Characteristics, LASERS, Stimulated emission and photon amplification, Einstein Coefficients, Laser oscillation conditions, Laser diode, Solar Energy Spectrum, photovoltaic device principles, Solar Cells

Text Book(s)

Modules 1-5

1. Semiconductor Physics and Devices (Fourth Edition), Donald A. Neamen, McGraw-Hill 2012.

Reference

1. Physics of Semiconductor Devices, S. M. Sze, 2nd Edition, Willey-Interscience Publication 1986

Modules 6

1. Online course: Semiconductor Optoelectronics by M. R. Shenoy on NPTEL
2. Optoelectronics and Photonics: Principles and Practices by S. O. Kasap, Prentice Hall 2001

Syllabus for Semester I / II

(Electronics Engineering, Electronics and Communication Engineering, Computer Science Engineering, Information Technology, Computer Science Engineering (Data Science))

Course Code : PHP156 **Course : Semiconductor Physics (Lab)**

Category : Basic Science Course

L: 0 Hrs. T: 0 Hrs. P: 3 Hrs. Per week **Total Credits : 1.5**

Course Outcomes

The Physics Lab course consists of experiments illustrating the principles of physics relevant to the study of science and engineering. At the end of the Course the students will learn to:

1. Develop skills to impart practical knowledge in real time.
2. Understand principle, concept, working and application of areas in physics and compare the results obtained with theoretical calculations.
3. Understand measurement technique, and report the results obtained through proper graph plotting and error analysis.

In addition to the General physics experiments, the Lab turns will be utilized for performing the experiments based on the following lists as specific to Program

General Physics

1. Error analysis and graph plotting
2. Newton's law of cooling
3. Simple Pendulum
4. Magnetic flux using deflection magnetometer
5. Dispersive power and determination of Cauchy's constants
6. Data analysis using Mathematica.
7. Cathode Ray Oscilloscope

Semiconductor Physics and Devices

1. Energy gap of semiconductor/thermister
2. Study of Hall Effect
3. Parameter extraction from I-V characteristics of a PN junction diode
4. Parameter extraction from I-V characteristics of a zener diode

5. Study of diode rectification
6. Parameter extraction from I-V characteristics of a transistor in common-emitter configuration.
7. V-I Characteristics of Light Emitting Diodes
8. Study of a photodiode
9. Solar Cell (Photovoltaic cell)
10. Resistivity measurement by Four Probe method

A minimum of 8 experiments to be performed from the following list of experiments

Syllabus for B.E. Semester I / II

Bachelor of Engineering

Course Code : MAT151 **Course : Calculus**

L: 3 Hrs., T: 1 Hrs., P: 0 Hrs., Per week **Total Credits : 04**

Course Objective

The objective of this course is to familiarize the prospective engineers with techniques in Calculus and multivariate analysis. It aims to equip the students with standard concepts and tools at an intermediate to advanced level that will serve them well towards tackling more advanced level of mathematics and applications that they would find useful in their disciplines.

Course Outcomes

On successful completion of the course, the students will learn:

1 . The fallouts of Mean Valve Theorems that is fundamental to application of analysis to Engineering problems , to deal with functions of several variables that are essential in most branches of engineering.

2 . Basics of improper integrals, Beta and Gamma functions, Curve Tracing , tool of power series and Fourier series for learning advanced Engineering Mathematics.

3 . Multivariable Integral Calculus and Vector Calculus and their applications to Engineering problems.

Syllabus

Module 1: Calculus: (7 hours)

Rolle's theorem, Mean value theorems, Taylor's and Maclaurin's series expansions; Indeterminate forms and L'Hospital's rule; radius of curvature (Cartesian form), evolutes and involutes

Module 2:Multivariable Calculus (Differentiation) (8 hours)

Limit, continuity and partial derivatives, Eulers Theorem, chain rule, total derivative, Jacobians, Maxima, minima and saddle points; Method of Lagrange multipliers.

Module 3 Calculus: (6 hours)

Evaluation of definite and improper integrals; Beta and Gamma functions and their properties; Tracing of curves(Cartesian form)

Module 4:Sequences and series: (7 hours)

Convergence of sequence and series, tests for convergence, power series, Fourier series: Half range sine and cosine series, Parseval's theorem.

Module 5: Multivariable Calculus (Integration) (7 hours)

Multiple Integration: Double and triple integrals (Cartesian and polar), change of order of integration in double integrals, Change of variables (Cartesian to polar), Applications: areas and volumes by double integration Center of mass and Gravity (constant and variable densities).

Module 6 : Vector Calculus (7 hours)

Vector Differentiation, Directional derivatives, total derivative, Gradient, Curl and Divergence. Vector integration , Theorems of Green, Gauss and Stokes.

Topics for self learning

Maxima and minima for function of one variable, Geometrical interpretation of Partial Differentiation(Tangent plane and Normal line) , Applications of definite integrals to evaluate perimeter, area, surface areas and volumes of revolutions.

Textbooks/References

1. Erwin Kreyszig, Advanced Engineering Mathematics, 9th Edition, John Wiley & Sons, 2006.
2. Veerarajan T., Engineering Mathematics for first year, Tata McGraw-Hill, New Delhi, 2008.
3. N.P. Bali and Manish Goyal, A text book of Engineering Mathematics, Laxmi Publications, Reprint, 2010.
4. B.S. Grewal, Higher Engineering Mathematics, Khanna Publishers, 35th Edition, 2000.
5. Ramana B.V., Higher Engineering Mathematics, Tata McGraw Hill New Delhi, 11th Reprint, 2010.
6. A text book of Applied Mathematics Volume I & II, by P. N. Wartikar and J. N. Wartikar, Pune Vidhyarthi Griha Prakashan, Pune - 411030 (India).

Syllabus for B.E. Semester I / II

Course No. MAT152 **Course : Differential Equations, Linear Algebra, Statistics & Probability**

L: 3 Hrs., T: 0 Hrs., P: 0 Hrs., Per week **Total Credits : 03**

Course Objective

The objective of this course is to familiarize the prospective engineers with techniques in Ordinary differential equation, statistics, probability and Matrices.

It aims to equip the students to deal with advanced level of mathematics and applications that would be essential for their disciplines.

Course Outcomes

On successful completion of the course, the students will learn:

1 . The effective mathematical tools for the solutions of ordinary differential equations that model physical processes.

2 . The essential tool of matrices in a comprehensive manner.

3 . The ideas of probability and various discrete and continuous probability distributions and the basic ideas of statistics including measures of central tendency, correlation and regression.

Syllabus

Module 1:First order ordinary differential equations(7 hours)

Exact, linear and Bernoulli's equations, Euler's equations, Equations not of first degree : equations solvable for p, equations solvable for y, equations solvable for x and Clairaut's type.

Module 2: Ordinary differential equations of higher orders (8 hours)

Second order linear differential equations with constant and variable coefficients, method of variation of parameters, Cauchy-Euler equation.

Module 3:Basic Statistics: (7 hours)

Curve fitting by the method of least squares- fitting of straight lines, second degree parabolas and more general curves, correlation and regression – Rank correlation, Multiple regression and correlation.

Module 4: Basic Probability: (8 hours)

Probability spaces, conditional probability, independence; Discrete random variables, Binomial distribution, Poisson distribution, Normal distribution. Relation between binomial, Poisson and Normal distributions.

Module 5: Matrices (10 hours)

Algebra of matrices, Inverse and rank of a matrix, rank-nullity theorem; System of linear equations; Symmetric, skew-symmetric and orthogonal matrices; Eigen values and eigenvectors; Diagonalization of matrices; Cayley-Hamilton Theorem, Orthogonal transformation and quadratic to canonical forms.

Topics for Self Learning

Application of Differential Equations.

Textbooks/References

1. Erwin Kreyszig, Advanced Engineering Mathematics, 9th Edition, John Wiley & Sons, 2006.
2. W. E. Boyce and R. C. DiPrima, Elementary Differential Equations and Boundary Value Problems, 9th Edition, Wiley India, 2009.
3. S. L. Ross, Differential Equations, 3rd Ed., Wiley India, 1984.
4. E. A. Coddington, An Introduction to Ordinary Differential Equations, Prentice Hall India, 1995.
5. E. L. Ince, Ordinary Differential Equations, Dover Publications, 1958.
6. B.S. Grewal, Higher Engineering Mathematics, Khanna Publishers, 35th Edition, 2000.
7. Theory and Problems of probability and statistics : 2nded :J. R. Spiegal ,Schaum series
8. A text book of Applied Mathematics Volume I & II, by P. N. Wartikar and J. N. Wartikar, Pune Vidhyarthi Griha Prakashan, Pune-411030 (India).
9. S. Ross, A First Course in Probability, 6th Ed., Pearson Education India, 2002.

Syllabus of Mathematics Computational Lab for Semester I/II, B.E. (2018-19)

Course Code : MAP151 **Course : Computational Mathematics Lab**

L:0 Hr., T:0Hrs., P:2 Hrs., Per week **Total Credits : 1**

Course Outcomes

The Computational Mathematics Lab course will consist of experiments demonstrating the principles of mathematics relevant to the study of science and engineering. Students will show that they have learnt laboratory skills that will enable them to properly acquire and analyze the data in the lab and draw valid conclusions. At the end of the Course the students will learn to:

1. Develop skills to impart practical knowledge in real time.
2. Understand principle, concept, working and application of areas in mathematics and compare the results obtained with theoretical calculations.
3. Understand basics of mathematics, and report the results obtained through proper programming.

The Lab turns will be utilized for performing the experiments based on the following list:

1. Calculus
2. Ordinary Differential Equations
3. Statistics
4. Linear Algebra

Suggested References

1. Computational Mathematics Lab Manual written by the Teaching Faculty of Mathematics Department, RCOEM.

A minimum of 8 experiments to be performed based on the above list.

Syllabus of Group 1 - Semester I and Group 2 - Semester II, Bachelor of Engineering

Course Code : EET151 Course : Basic Electrical Engineering

Course Outcomes

At the end of this course, students will demonstrate the ability

CO1: Understand and analyze basic ac and dc electric circuits and magnetic circuits

CO2: Understand working principles of electrical machines: Transformer, Induction motor, DC machines

CO3: Apply the knowledge of power converter for suitable applications

CO4: Introduce and identify the components of power systems and low-voltage electrical Installations.

Module 1: Introduction to Power system (2 hours)– CO4:

Introduction to Power Generation (Thermal, Hydro, Nuclear, Wind, and Solar) with block schematic presentation only. Single line diagram for Generation, Transmission & Distribution through different voltage levels.

Module 2 : DC Circuits & Magnetic Circuits(8 hours) - CO1:

Electrical circuit elements (R, L and C), voltage and current sources, Kirchoff's current and voltage laws, analysis of simple circuits with dc excitation, Time-domain analysis of first order RL and RC circuits, Magnetic materials, BH characteristics, Basics of Magnetic circuits.

Module 3: Single Phase AC Circuits (6 hours) - CO1:

Representation of sinusoidal waveforms, peak and rms values, phasor representation, real power, reactive power, apparent power, power factor. Analysis of single-phase ac circuits consisting of R, L, C, RL, RC, RLC combinations (series and parallel), resonance.

Module 4: Three Phase AC Circuits (4 hours) - CO1:

Three phase Ac generation, Three phase balanced circuits, voltage, and current relations in star and delta connections. Power factor improvement.

Module 5: Transformers (6 hours) - CO2:

Ideal and practical transformer, Equivalent circuit, losses in transformers, regulation, and efficiency. Auto transformer and three-phase transformer connections.

Module 6: Electrical Machines (8 hours) - CO2:

Generation of rotating magnetic fields, Construction and working of a three-phase induction motor, Significance of torque-slip characteristic. Loss components, efficiency, starting of induction motor. Single- phase induction motor. Construction, working, torque-speed characteristic, and speed control of separately excited dc motor.

Module 7: Power Converters (4 hours) - CO3:

Block schematic introduction to power converters and its practical applications (DC-DC, DC-AC, AC-DC, AC- AC), Types of Batteries, Important Characteristics for Batteries and battery backup.

Module 8: Electrical Installations (4 hours) - CO4:

Components of LT Switchgear: Switch Fuse Unit (SFU), MCB, ELCB, MCCB, Types of Wires and Cables, Earthing. Elementary calculations for energy consumption, energy tariff.

Text / References

1. D. P. Kothari and I. J. Nagrath, “Basic Electrical Engineering”, Tata McGraw Hill, 2010.
2. D. C. Kulshreshtha, “Basic Electrical Engineering”, McGraw Hill, 2009.
3. L. S. Bobrow, “Fundamentals of Electrical Engineering”, Oxford University Press, 2011.
4. E. Hughes, “Electrical and Electronics Technology”, Pearson, 2010.
5. V. D. Toro, “Electrical Engineering Fundamentals”, Prentice Hall India, 1989.
6. Electrical Technology: B. L. Thereja, S. Chand Publications.
7. Basic Electrical Engineering: S. B. Bodkhe, N. M. Deshkar, P. P. H. Pvt. Ltd.

Course Outcomes

Upon completion of this course, the students shall be able to,

CO1: Co-relate, analyze and apply the fundamental principles of science and engineering to understand the laboratory experimental work.

CO2: Connect the electric circuit, perform the experiment, analyze the observed data and make valid conclusion.

CO3: Write report based on the performed experiments (journal) with effective presentation of diagrams and characteristics/graphs.

CO4: Carry out survey of electrical energy consumption at home and calculate monthly energy bill as per the tariff of power Distribution Company.

List of Experiments

1. To verify Kirchhoff's laws for D.C. Circuits
2. Verification of Kirchhoff's laws to AC circuit(RLC series)
3. Verification of Kirchhoff's laws to AC circuit (RLC parallel).
4. To study speed control of D.C. shunts motor by:
   a) Armature voltage Control method.
   b) Field current/flux control method.
5. To study the balanced Three phase system for star and delta connected balanced load.
6. Improvement of power factor by using static capacitors
7. To determine regulation and efficiency of a single phase transformer by open circuit (o.c) and short circuit (s.c.) tests.
8. To determine regulation and efficiency of a single phase transformer by direct loading test

Demonstration/ Study experiment

9. To study B-H curve for different magnetic material
10. To study Buck converter
11. To study Boost converter

Demonstration of cut out sections of machines:

i. DC Machine

ii. Three phase squirrel cage induction motor

iii. Synchronous machine

Syllabus of Department of Mechanical Engineering

Course Code : MET151 **Course: Engineering Graphics and Design**

L:1 Hr., T:0Hrs., P:0 Hrs., Per week **Total Credits : 01**

Course Outcomes

The expected learning outcome is that, the students shall be able to

1. Draw and interpret technical drawing
2. Convert 2-D to 3-D drawing and vice versa.
3. Represent the various positions of planes and solids in different orientations.
4 Develop the solid surface for sheet metal working.

UNIT 1 : Introduction to Engineering Drawing

Principles of Engineering Graphics and their significance, usage of drawing instruments, Lettering and dimensioning.

UNIT 2 : Orthographic Projections

Principles of Orthographic Projections -Conventions : Projections of Points and lines (line inclined to both planes) Projections of planes (inclined to both the planes), Introduction to Auxiliary Planes;

UNIT 3 : Projections of Solids

Inclined to both the Planes - Auxiliary Views; Draw simple annotation, dimensioning and scale. Floor plans that include : windows, doors, and fixtures such as WC, bath, sink, shower, etc.

UNIT 4 : Sections and Sectional Views of Right Angular Solids

Prism, Cylinder, Pyramid Cone-Auxiliary Views; Development of surface of Right Regular solids - Prism, Pyramid, Cylinder and Cone; Draw the sectional orthographic views of geometrical solids, objects from industry and dwellings (foundation to slab only)

UNIT 5 : Isometric Projections

Principles of Isometric projection - Isometric Scale, Isometric Views, Conventions; Isometric Views of Simple Solids; Conversion of Orthographic views to Isometric Views / Projection.

Suggested Text / Reference Books

i) Bhatt N. D. Panchal V.M. & Ingle P.R., (2014) Engineering Drawing, Charotar Publishing House.

ii) Jolhe D. A. (2016) Engineering Drawing with an Introduction to Auto CAD", Tata McGraw- Hill Publishing Co. Ltd., New Delhi.

iii) Narayana K. L. & P. Kannaiah (2008), Text book on Engineering Drawing, Scitech Publishers.

iv) Shah M. B. & Rana B. C. (2008), Engineering Drawing and Computer Graphics, Pearson Education.

v) Agrawal B & Agrawal C. M. (2012), Engineering Graphic, TMH Publication.

vi) Corresponding set of CAD Software Theory and User Manuals.

Syllabus of Department of Mechanical Engineering

Course Code : MEP151 Course : Engineering Graphics & Design Lab

L:0 Hr., T:0Hrs., P:4 Hrs., Per week Total Credits : 02

Course Outcomes

Students are prepared for actual work situations through practical training in a new state of the art computer designed CAD laboratory using engineering software. The student will learn to :

1. Draw and interpret technical drawing
2. Plan the sheet layout for the given drawing
3. Convert 2-D to 3-D drawing and vice versa
4. Represent the various positions of planes and solids in different orientations.
5. Develop the solid surface for sheet metal working
6. Use & demonstrate drafting package.

UNIT 1 : Introduction to Engineering Drawing

Conic sections including the Rectangular Hyperbola (General method only); Cycloid, Epicycloids, Hypocycloid and involutes; Introduction to Scales.

UNIT 2 : Orthographic Projections

Principles of Orthographic Projections -Conventions - Projections of Points and lines inclined to both planes; Projections of planes - Auxiliary Planes.

UNIT 3 : Projections of Solids

Inclined to both the Planes Auxiliary Views; Draw simple annotation, dimensioning and scale, Floor plans that include: windows, doors, and fixtures such as WC, bath, sink, shower, etc.

UNIT 4 : Sections and Sectional Views of Right Angular Solids

Prism Cylinder, Pyramid, Cone - Auxilary Views; Development of surfaces of Right Regular Solids Prism, Pyramid, Cylinder and Cone; Draw the sectional orthographic views of geometrical solids, objects from industry and dwellings (foundation to slab only).

UNIT 5 : Isometric Projections

Principles of Isometric projection - Isometric Scale, Isometric Views, Conventions; Isometric Views of Simple Solids; conversion of Orthographic views to Isometric views / Projection.

UNIT 6 : Overview of Computer Graphics

Demonstrating knowledge of the theory of CAD software such as (the Menu System Toolbars Standard, Object Properties, Draw, Modify and Dimension), Drawing Area (Background, crosshairs, Coordinate Systems), Dialog boxes and windows, Shortcut menus (Button Bars), The command Line (wherever applicable), The Status Bar, Different methods of zoom as used in CAD, select and erase objects; Isometric Views of lines, Planes, Simple and compound solids).

UNIT 7 : Customization & CAD Drawing

Setting up drawing page and the printer, including scale settings, Setting up of units and Drawing limits; ISO and ANSI standards for coordinate dimensioning; Orthographic constraints, map to objects, manually and automatically, Producing drawings by using various coordinate input entry methods to draw straight lines, Applying various ways of drawing circles.

UNIT 8 : Annotations Layering & Other Functions

Applying dimensions to objects, applying annotations to drawings; Changing line lengths through modifying existing lines (extend/lengthen); Printing documents to paper using the print command; orthographic projection techniques.

UNIT 9 : Demonstration of a simple team design project that illustrates

Geometry and Topology of Engineered Components Creation of Engineering models and their presentation in standard 2D blueprint form and as 3D wire-frame and shaded solids; Meshed toplogies for engineering, Introduction to Building Information Modeling (BIM), Drafting and Design Package, 3D Printing.

List of sheets

1. Curves (ellipse, Parabola, hyperbola, Cycloid, involute)
2. Line, Planes, Solids
3. Application of Section and development of solids
4. Orthgraphic Projection
5. Isometric
6. Auto CAD practice sheet 1
7. Auto CAD practice sheet 2
8. Blueprint sheet

Suggested Text/ Reference Books

1. Bhatt N.D. Panchal V.M. & Ingle P.R., (2014), Engineering drawing, Charotar Publiishing house
2. Jolhe D.A., (2016) Engineering drawing with an Introduction to Auto CAD", Tata McGraw-Hill Publishing Co. Ltd., New Delhi.
3. Shah M.B. & Rana B.C. (2008), Engineering drawing and Computer Graphic, Pearson Education.
4. Agarwal B & Agarwal C.M. (2012), Engineering Graphics, TMH PUBLICATION
5. Narayana K.L & P Kannaiah (2008), Text Book on Engineering Drawing, Scitech Publishers.
6. (Concesponding set of) CAD Software Theory and USER Manuals.

Syllabus for B.E. Semester I Department of Humanities

Course Code : HUT152 | **Course : Constitution of India**

L: 2 Hrs. T: 0 Hrs. P: 0 Hrs. Per week | **Total Credits : 0**

Course outcome

1. Students will understand the role of constitution in democratic India
2. Students will be responsible students by knowing their fundamental rights and duties
3. Students will develop better understanding of democratic functions of the government of India
4. Students will form better understanding of system of governance for effective participation

Course content

1. Meaning of the constitution law and constitutionalism
2. Historical perspective of the Constitution of India
3. Salient features and characteristics of the Constitution of India
4. Scheme of the Fundamental Rights
5. The scheme of the Fundamental Duties and its legal status
6. The Directive Principles of State Policy – Its importance and implementation
7. Federal structure and distribution of legislative and financial powers between the Union and the States
8. Parliamentary Form of Government in India – The constitution powers and status of the President of India
9. Union Executive: structure, functions
10. Judiciary: Structure, role with special reference to PIL, writ petitions, strengthening of democracy & social justice
11. Amendment of the Constitutional Powers and Procedure
12. Emergency Provisions: National Emergency, President Rule, Financial Emergency
13. Local Self Government – Constitutional Scheme in India
14. Provisions of civil services: Characteristics, functions, merits and demerits
15. Democratic principles in industry

Book

Durga Das Basu "An Introduction to Constitution of India" 22nd Edition, LexisNexis

Syllabus for B.E. Semester I Department of Humanities

Course Code : PEP151 **Course : Yoga / Sports**

L: 0 Hrs. T: 0 Hrs. P: 2 Hrs. Per week **Total Credits : 0**

Course outcome

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

1. Understand fundamental skills and basic rules of games offered by the Physical Education Department of RCOEM.
2. Obtained health related physical fitness.
3. Develop body-mind co-ordination through games and yoga.
4. Changed sedentary life styles towards active living.

Brief Objectives of Sports/Yoga Practical Classes

It has long been proven that a healthy body leads to a healthy mind. With a strong belief in this, Physical Education Department at RCOEM will conduct Sports/Yoga Classes with the objective of maintaining health, fitness and wellness of students as well as create awareness about need for good health and physical fitness. The objective would also be to make the all-round development with team spirit, social values as well as to identify and develop leadership qualities in students through various sports activities. Sports activities would also be conducted with the objective to provide better interaction and recreation to the students which is an important neutralizer for stress. Additionally, the objective would be to evaluate the health related fitness of students so as to recommend and conduct specific Yoga and Sports activities. The emphasis is on participation, with healthy competition.

Programme Outline

- **Sports :**

1. Introduction to sports, offered by the department.
2. Health and safety issues related to sports; knowledge, recognition and ability to deal with injuries and illness associated with sports.
3. Practicing the fundamental skills and bringing awareness of basic rules and regulations.
4. Conduction of small recreational games and activities.

- **Yoga :** Includes various sitting, standing and lying Asanas, Suryanamaskars and Pranayamas.
- Physical Efficiency Tests : This includes 6 health related physical fitness tests.

Components	Name of Tests
Speed	50 mts Dash
Agility	Shuttle run
Cardiovascular Endurance	8 mins Run/Walk
Test Flexibility	Sit and Reach Test
Abdominal Strength (M) / shoulder strength (F)	Bent Knee Sit-ups (M)/ Modified Pull-ups (F)
Yogic exercises	Suryanamaskars

Syllabus for B.E. Semester I / II

Course Code : CHT152 **Course : Chemistry**

L: 3 Hrs, T: 1 Hr, P : 0 Hr., Per week **Total Credits : 4**

Course Outcomes

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

- Predict the properties and interactions of chemical substances by understanding their composition at the atomic level. **[CO for Unit – 1]**
- Conversant in applying unique properties of nano-materials to solve challenges in our life. **[CO for Unit – 2]**
- Explain the differences in the mechanical behavior of engineering materials based upon bond type, structure, composition, and processing. **[CO for Unit – 3]**
- Study chemical kinetics using concepts of computational chemistry. **[CO for Unit – 4]**
- Discuss how spectroscopic methods are used for qualitative and quantitative analyses. **[CO for Unit – 5]**
- Analyse impurities present in the water and suggest the methodology for its removal. **[CO for Unit – 6]**

Syllabus

Unit 1: Solid State Chemistry (7 Hours)

Bondings in atoms : Primary bonding: ionic, covalent, metallic. Secondary bonding: dipole-dipole, induced dipole-induced dipole, London dispersion/van der Waals, hydrogen. Shapes of molecules: hybridization, LCAO-MO, VSEPR theory.

Electronic material : Band theory: metals, insulators, and semiconductors. Band gaps, doping. Silicon wafer production.

Unit 2: Nano-material-I (7 Hours)

Basics of Nano chemistry : Definition of Nano, Scientific revolution-Atomic Structure and atomic size, emergence and challenges of nanoscience and nanotechnology, carbon age-new form of carbon (CNT to Graphene), One dimensional, Two dimensional and Three dimensional nanostructured materials, mechanical-physical-chemical properties.

Application of Nanomaterial : Molecular electronics and nano electronics, Nanotechnology for waste reduction and improved energy efficiency, Carbon Nanotubes for energy storage, Hydrogen Storage in Carbon Nanotubes, nanotechnology based water treatment strategies.

Unit 3: Advanced Materials: (7 hours)

Composite materials : Introduction, Classification: Polymer Matrix Composites, Metal Matrix Composites, Ceramic Matrix Composites, Carbon–Carbon Composites, Fiber - Reinforced Composites and Applications.

Reinforcements : Fibres- Glass, Kevlar, Carbon, Silicon Carbide, and Born Carbide Fibres.

Industrial Polymer : Thermoplastics, Thermosetting Plastics, Polymers used in electronic industries, Piezo and pyroelectric polymers, Polymers in optical media data storage devices.

Unit 4: Computational Chemistry [6 Hours]

Rate of the reaction, Order and Molecularity of the reaction, Rate expression for Zero Order, First Order and Second Order Reactions, Effect of the temperature, Use of Mathematica for determining rate of the reaction, etc.

Unit 5: Material Characterization using different Spectroscopic Techniques [7 Hours]

Fundamentals of spectroscopy, Infrared Spectroscopy, Electronic Spectrocopy, Nuclear Magnetic Resonance Spectroscopy.

Fundamentals of X-Ray Diffractions (XRD), X-Ray Fluorescence (XRF) spectroscopy.

Unit 6: Water Technology [8 Hours]

Impurities in natural water, hardness and alkalinity, Disadvantages of hardness i. e. sludge and scale formation, softening of water using lime-soda, zeolite and ion-exchange method, advantages and limitations of these water softening processes, Desalination of water using Reverse Osmosis.

Text Books

1. J. Michael Hollas, Modern Spectroscopy, Fourth Edition, John Wiley and Sons, 2004.
2. William Kemp, Organic Spectroscopy, Third Edition, Palgrave Publication, 1991.
3. Bradley D. Fahlman, Materials Chemistry, Third Edition, Springer Nature, 2018.
4. Brian W. Pfennig, Principles of Inorganic Chemistry, John Wiley and Sons, 2015.
5. Steven S. Zumdahl, Donald J. DeCoste, Chemical Principles, Eighth Edition, Cengage Learning, 2017.
6. Catherine E. Housecroft and Edwin C. Constable, Chemistry: An Introduction to Organic, Inorganic and Physical Chemistry, Third Edition, Pearson Education Limited, 2006.
7. Michael J. Moran and Howard N. Shapiro, Fundamentals of Engineering Thermodynamics, Fifth Edition, John Wiley and Sons, 2006.
8. Donald L. Pavia, Gary M. Lampman, George S. Kriz, and James R. Vyvyan, Introduction to Spectroscopy, Fifth Edition, Cengage Learning, 2009.
9. C. N. R. Rao, A. Muller and A. K. Cheetham, The Chemistry of Nanomaterials: Synthesis, Properties and Applications, Wiley-VCH, 2004.
10. P. C. Jain and Monica Jain, Engineering Chemistry, DhanpatRai Publication.
11. S. S. Dara, A Textbook of Engineering Chemistry, S. Chand Publications.
12. J. D. Lee, Concise Inorganic Chemistry, Fourth Edition, Chapman and Hall Publications.

Syllabus for B.E. Semester I / II

Course Code : CHP152 **Course : Chemistry Lab**

L: 0 Hrs., T: 0 Hrs., P: 3 Hrs., Per week **Total Credits : 1.5**

Course Outcomes

The chemistry laboratory course will consist of experiments illustrating the principles of chemistry relevant to the study of science and engineering.

The students will learn to:

- Estimate the amount of different impurities in water/waste water samples.
- Estimate rate constants of reactions and order of the reaction from concentration of reactants/products as a function of time and to validate adsorption isotherms.
- Measure molecular/system properties such as surface tension, viscosity of aqueous or other industrially important liquids/mixtures etc.
- Synthesize a polymer or drug molecule or nano-material.
- Use principle of spectroscopic techniques for structural determination.

List of Experiments: [Any Eight from the List]

[1] Preparation of different Solutions: Molar solution, Normal solution and percent solution and Determination of concentration.

[2] To find out types of alkalinity and estimation of their extent in the water sample.

[3] Estimation of temporary, permanent and total hardness present in the water sample using complexometric titration method.

[4] Spectroscopic/Colorimetric determine of wavelength of maximum absorption of chemical/biological compound in solution and determination of concentration using Lambert-Beer's Law.

[5] Determination of rate of the reaction of hydrolysis of ethyl acetate at room temperature and analysis of experimental data using Computational Software.

[6] To study chemical kinetics of peroxydisulphate and iodide ions reactions and to find out order of the reaction and analysis of experimental data using Computational Software.

[7] Synthesis of Nano-material/Polymer and its study.

[8] Determination of relative and kinematic viscosities of aqueous solutions of Poly-ethylene glycol (Polymeric Liquid) using Redwood Viscometer (type I or II) at different temperatures.

[9] To study effect of bondings of water molecules with electrolyte (NaCl/KCl) and non-electrolyte solute (Soap) in the solution through Surface Tension Determination.

[10] Study of ion-exchange column for removal of hardness in the water sample.

[11] Demonstrations of organic spectral techniques: IR, NMR.

[12] Demonstration of in-organic spectral techniques: XRD, XRF.

Text Books/Reference Books

(1) S. S. Dara, A Textbook on Experiments and Calculations in Engineering Chemistry, S. Chand Publications.

(2) J. B. Yadav, Advanced Practical Physical Chemistry, Krishna's Prakashan Media (P) Limited.

(3) A. J. Elias, Collection of Interesting General Chemistry Experiments, Universities Press Publications.

(4) V. K. Ahluwalia, S. Dhingra and A. Gulati, College Practical Chemistry, Universities Press Publications.

(5) Ashutosh Kar, Advanced Practical Medicinal Chemistry, New Age International Publisher.

Syllabus of Group 1 - Semester I and Group 2 - Semester II, Bachelor of Engineering

Course Code: CST151 Course : Programming for Problem Solving

L: 4 Hrs.,T: 0 Hrs.,P: 0 Hrs.,Per week Total Credits : 4

Course Outcomes

On successful completion of course student will learn:

1. To formulate simple algorithms for arithmetic and logical problems, translate the algorithms to programs (in C language), test and execute the programs and correct syntax and logical errors.
2. To implement conditional branching, iteration and recursion, to decompose a problem into functions and synthesize a complete program using divide and conquer approach.
3. To use arrays, to solve various matrix operations, searching, sorting and pointers, structures for the formulation of algorithms and programs.
4. To understand basics of file operations and to apply various I/O operations for file handling programming.

UNIT-I: Introduction to Programming

Introduction to components of a computer system (disks, memory, processor, where a program is stored and executed, operating system, compilers etc.)

Idea of Algorithm : Steps to solve logical and numerical problems. Representation of Algorithm: Flowchart / Pseudocode with examples. Arithmetic expressions and precedence

UNIT-II: C Programming Language

Introduction to C language: Keywords, Constant, Variable, Data types, Operators, Types of Statements, Preprocessor Directives, Decision Control Statement-if, if-else, Nested if-else statement, Switch case, Loops and Writing and evaluation of conditionals and consequent branching.

UNIT-III: Arrays and Basic Algorithms

Arrays: 1-D, 2-D, Character arrays and Strings.

Searching, Basic Sorting Algorithms (Bubble, Insertion and Selection), Finding roots of equations, notion of order of complexity through example programs (no formal definition required)

UNIT-IV: Functions and Recursion

User defined and Library Functions, Parameter passing in functions, call by value, Passing arrays to functions: idea of call by reference. Recursion: As a different way of solving problems. Example programs, such as Finding Factorial, Fibonacci series, Ackerman function etc. Quick sort or Merge sort.

UNIT-V: Pointers and Structures

Structures, Defining structures, Array of Structures, Introduction to pointers, Defining pointers, Pointer arithmetic, pointer operators, Use of Pointers in self-referential structures, notion of linked list (no implementation) UNIT-VI: File handling

Streams in C, Types of Files, File Input/ Output Operations: Modes of file opening, Reading and writing the file, Closing the files, using fflush().

Text Books

1. Programming in ANSI C : E. Balguruswami McGraw Hill
2. Mastering C: K. R. Venugopal and S. R. Prasad, Tata McGraw Hill

Reference Books

1. Programming with C: Byron Gottfried, Schaums Outline Series.

Let Us C: Yashwant Kanetkar, BPB Publication

Syllabus of Group 1 - Semester I and Group 2 - Semester II, Bachelor of Engineering

Course Code: CSP151 **Course : Programming for Problem Solving Lab**

L: 0 Hrs.,T:0 Hrs.,P:2 Hrs.,Per week **Total Credits : 1**

Course Outcomes

On successful completion of course student will be able to:

1. Understand the fundamentals of C programming and choose the loops and decision making statements to solve and execute the given problem.
2. Implement different Operations on arrays also design functions to solve the given problem using C programming.
3. Understand pointers, structures, unions and apply them to develop programs.
4. Implement file Operations in C programming for a given application.

Experiment List

1. To Study Unix / Linux Operating System Environment and demonstrate basic data types in C and implementing arithmetic expressions.
2. To Study and Demonstrate Decision Control Structure.
3. To Study and Demonstrate Loop Control Structures.
4. To Study and Demonstrate Multi-way Decision Control Structures (Switch Case).
5. To Study and Demonstrate 1D and 2D Array.
6. To Study and Demonstrate Function and Recursion.
7. To Study and Demonstrate Structure and Pointers.
8. To Study and Demonstrate File Handling in C.

CREATIVITY INNOVATION AND DESIGN THINKING COURSE SYLLABUS

Course Code : IDT151 **Credits : 1**

L:1Hrs., T:0Hrs., P:0Hrs., Per week

Course Outcomes

C1: Be familiar with processes and methods of creative problem solving

C2: Enhance their creative and innovative thinking skills

C3: Practice thinking creatively and innovative design and development

Detailed Topics

UNIT I. Introduction: Making a case for creativity, Creative thinking as a skill, Valuing diversity in thinking: Thinking preferences, Creativity styles, Creativity in problem solving

UNIT 2. Pattern Breaking: Thinking differently , Lateral thinking, Mind stimulation: games, brain-twisters and puzzles, Idea-collection processes, Brainstorming/Brainwriting, The SCAMPER methods, Metaphoric thinking, Outrageous thinking , Mapping thoughts, Other (new approaches)

UNIT 3. Using Math and Science, Systematic logical thinking, Using math concepts, Eight-Dimensional (8D) Approach to Ideation: Uniqueness, Dimensionality, Directionality, Consolidation, Segmentation, Modification, Similarity, Experimentation

UNIT4. Systematic Inventive Thinking: Systematic inventive thinking: The TRIZ methodology, Decision and Evaluation: Focused thinking framework, Six thinking hats , Ethical considerations

UNIT 5. Design for Innovation: Introduction to design for interaction, nine lessons for innovation, difference in creativity and innovation, Building blocks for innovation

UNIT 6.Intellectual Property: Introduction to intellectual property: Patents, Copyrights©, Trademarks ®, Trade Secret, Unfair Competition.

Reference Books and Text Book

1. Creative Problem Solving for Managers - Tony Proctor - Routledge Taylor & Francis Group
2. 101 Activities for Teaching creativity and Problem Solving - By Arthur B Vangundy - Pfeiffer
3. H. S. Fogler and S.E. LeBlanc, Strategies for Creative Problem Solving, Prentice Hall
4. E. Lumsdaine and M. Lumsdaine, Creative Problem Solving, McGraw Hill,
5. J. Goldenberg and D. Mazursky, Creativity in product innovation. Cambridge University Press, 2002.

Course Assignments for internal continuous assessment of 20 Marks (NO T1 and T2)

- Brain teasers (aka Puzzle Busters, to be solved individually)
- Cartoon captions (small teams)
- TRIZ, a systematic ideation method, reading (individual)
- Book readings and discussions (small teams)
- Small teams presentations on innovation: (1) innovative individual, (2) innovative company, (3) innovative movie / game, (4) sustainable innovation, (5) innovation in business, (6) innovation in art, (7) innovation in architecture, (8) innovative nation, (9) innovation in science, and (10) innovation in engineering.
- Large groups hands-on projects
- Eight-dimensional (8D) ideation method examples
- Large teams videos

Syllabus Department of Industrial Engineering

Course Code : INT151 **Course : Workshop / Manufacturing Practices (Theory)**

L:1Hrs., T:0Hrs., P:0Hrs., Per week **Total Credits:1**

Course Outcomes

1. Identify the different manufacturing process commonly employed in Industry along with prevailing safety practices.
2. Identify the various tools and equipments to carry out different manufacturing processes accompanied by the inspection of the work part.

Syllabus

Unit-1 Fundamentals of metal cutting, single point cutting tool, fundamental mechanics of metal cutting, fitting operations, and associated measuring and marking tools

Unit-2 Introduction to pattern making for metal casting, different types of carpentry tools, measuring tools and marking tools, holding devices, different types of carpentry joints.

Unit-3 Smithy and Forging, Forging tools like chisels, hammers, types of furnaces, types of coal, Forming operations, Hot working and Cold working of metals.

Unit-4 Metal joining Process, mechanics of welding, types of welding, soldering and brazing, types of joints

Unit-5 Introduction to foundries, Metal Casting, types of sand, Introduction to Molding tools & casting process.

Unit-6 Introduction to Plastic Injection Molding

Suggested Text Book

1. "Elements of Workshop Technology" Hajra S.K, Choudhury A. K , Roy Nirjhar Vol. I and Vol .II, Media Promoters and Publishers Private Ltd. Mumbai.

Reference Books

1. Kalpakjian S. and Schmid S. “Manufacturing Engineering and Technology”4th Edition, Pearson India Education 2008
2. Roy A. and Lindberg, “Process and Materials of Manufacture”4th Edition, Prentice Hall India 1998.

Syllabus Department of Industrial Engineering

Course Code : INP151 **Course : Workshop/Manufacturing Practices Lab (Practical)**

L:0Hrs.,T:0Hrs.,P:2Hrs.,Per week **Total Credits :1**

Laboratory Outcomes

On the completion of the course the students shall be able to;

1. Recognize the different manufacturing process commonly employed in the Industry
2. Make the components using required manufacturing process, inspection methods while practicing the requisite safety precautions

Contents

1. Fitting Practice
2. Welding and Soldering Practice
3. Pattern Making Practice
4. Metal Casting Practice
5. Smithy and Forging Practice
6. Machining Practice
7. Plastic Molding Process
8. Glass Cutting Process

Suggested Text Book

1. “Elements of Workshop Technology'” Hajra S.K, Choudhury A.K , Roy Nirjhar Vol. I and Vol .II, Media Promoters and Publishers Private Ltd Mumbai.

Reference Books

1. Kalpak Jain S. and Schmid S. “Manufacturing Engineering and Technology”4th Edition, Pearson India Education 2008
2. Roy A. and Lindberg, “Process and Materials of Manufacture", Prentice hall India 1998.

Syllabus for B.E. Semester I / II Dept of Humanities Humanities and Social Sciences

Course Code: HUT151 **Course : English**

L: 2 Hrs. T: 0 Hrs. P: 0 Hrs. Per week **Total Credits : 2**

Course Objectives

The main objective of the subject is to enhance the employability skills of engineering students as well as communication skills at work place. The sub-objectives are:

1. To develop vocabulary of students.
2. To orient students in basic writing skills.
3. To orient students in functional grammar.
4. To orient students in the process of effective writing.
5. To provide practice and improve students' oral communication skills.

Course Outcomes

1. Students will have good word power.
2. Students will acquire basic writing skills.
3. Students will understand functional grammar and its usage.
4. Students will organize and express their thoughts effectively through written communication.
5. Students will learn oral communication skills in order to handle themselves effectively in an interview and group discussion

SYLLABUS

1. Vocabulary Building

1.1. The concept of Word Formation

1.2. Root words from foreign languages and their use in English

1.3. Acquaintance with prefixes and suffixes from foreign languages in English to form derivatives

1.4. Synonyms, Antonyms and standard abbreviations

2. Basic Writing Skills

2.1 Sentence Structures

2.2 Use of phrases and clauses in sentences

2.3 Importance of proper punctuation

2.4 Creating coherence

2.5 Organizing principles of paragraphs in documents

2.6 Techniques for writing precisely

3. Identifying Common Errors in Writing

3.1 Subject-verb agreement

3.2 Noun-pronoun agreement

3.3 Misplaced modifiers

3.4 Articles

3.5 Redundancies

Cliches

1. Nature and Style of sensible Writing

4.1 Describing

4.2 Defining

4.3 Classifying

4.4 Providing examples or evidence

2. Writing Practices

5.1 Comprehension

5.2 Precis Writing

5.3 Essay Writing

5.4 Letter Writing

5.5 Email Writing

3. Oral Communication

(This unit involves interactive practice sessions in Language Lab)

- Listening Comprehension
- Pronunciation, Intonation, Stress and Rhythm
- Common Everyday Situations : Conversations and Dialogues
- Communication at Workplace
- Interviews
- Formal Presentations

Books

1. Communication Skills. Sanjay Kumar and PushpLata. Oxford University Press. 2011.
2. Practical English Usage. Michael Swan. OUP. 1995.
3. Remedial English Grammar. F.T. Wood. Macmillan.2007
4. On Writing Well. William Zinsser. Harper Resource Book. 2001
5. Study Writing. Liz Hamp-Lyons and Ben Heasly. Cambridge University Press. 2006.
6. Exercises in Spoken English. Parts. I-III. CIEFL, Hyderabad. Oxford University Press

Syllabus for B.E. Semester I
Humanities

Course Code: HUP151 **Course : English Lab**

L: 0 Hrs. T: 0 Hrs. P: 2 Hrs. Per week **Total Credits: 1**

Course objective

1. To enhance competency of communication in English among learners.

Course outcomes

1. Students learn presentation and public speaking skills
2. Students learn to practice effective strategies for Personal Interview and Group Discussions
3. Students learn and effectively apply language skills – listening, speaking, reading and writing

Syllabus

1. Common Everyday Situations: Conversations and Dialogues
2. Pronunciation, Intonation , Stress, and Rhythm
3. Formal Presentations: Orientation
4. Formal Presentations : Practice Session
5. Interviews: Orientation
6. Interviews: Practice Session
7. Communication at Workplace: Group Discussion- Orientation
8. Communication at Workplace: Practice Session

III Semester

Department of Computer Science & Engineering

Course Code: CST251 **Course : Fundamentals of Digital Logic and Computer Architecture**

L: 4 Hrs, T: 0 Hr, P: 0 Hr, Per Week **Total Credits : 04**

Course Objectives

The objective of this course is to familiarize the prospective engineers with:

1. Concepts of digital logic for designing digital circuits.
2. Concepts of organization and architecture of a computer, including basic components like CPU, I/O, bus, pipeline, memory, and design of memory systems.

SYLLABUS

UNIT I Fundamental Concepts of Digital Systems

Overview of Boolean algebra, Minimization of combinational circuits using Karnaugh maps up to five variables, Design procedure of combinational circuits, Code Converters, and their use in realizing Boolean functions.

UNIT II Combinational Circuits

Multiplexers, Demultiplexer, Encoders, Decoders, Adders, Subtractor (Half, Full), BCD Adder/ Subtractor , ripple and carry look-ahead addition.

UNIT III Sequential Circuits

Flip-flops and latches: D, T, J-K, S-R, Master Slave J-K flip-flops Conversion of one FF to another FF. Sequential circuit Analysis, Sequential circuit Design: Counters, asynchronous and synchronous circuit design, Registers and Shift registers.

UNIT IV Basic Structure of Computers

Basic organization of computers, Block level description of the functional units as related to the execution of a program, Instructions set architecture of a CPU, Addressing modes, instruction set classification, Execution of a Complete Instruction, RISC versus CISC architectures, Floating point numbers representation.

UNIT V Memory Organization

Memory Technology, static and dynamic memory, Random Access and Serial Access Memories, Cache memory and Memory Hierarchy, Address Mapping, and memory management unit, Memory Allocation strategies.

Memory Design : RAM Design, Secondary storage – Magnetic disk, Optical disk

UNIT VI Input/Output Organization

I/O mapped I/O and memory mapped I/O, Concept of handshaking, Polled and Interrupt-driven I/O, DMA data transfer, Basic concepts Bus Control.

Pipelining: Basic concepts of pipelining, speedup, Pipeline hazards and their resolution.

Course Outcomes

On completion of the course the student will be able to

1. Understand fundamental concepts of Digital logics
2. Understand Combinational and Sequential Circuits.
3. Understand the basic components of a computer, including CPU, memories, and input/output with instruction execution and control unit design.
4. Understand the memory hierarchy and design of memories.

Text Books

1. Morris Mano; Digital Logic Design; Fourth edition, McGraw Hill
2. V.C.Hamacher, Z.G.Vranesic and S.G.Zaky; Computer Organisation; 5th edition; Tata McGraw Hill,2002.
3. W. Stallings; Computer Organization & Architecture; PHI publication; 2001.

Reference Books

1. A Anand Kumar; Fundamental of Digital Electronics; Second Edition, PHI
2. Computer Organization and Design, by David Patterson and John Hennessey," Elsevier. 2008.
3. Computer Architecture and Organization, by Hayes, J.P.1998,McGraw-Hill

III Semester

Department of Computer Science & Engineering

Course Code : CSP251 **Course : Fundamentals of Digital Logic and Computer Architecture Lab**

L: 0 Hrs, T: 0 Hr,P: 2 Hr, Per Week **Total Credits : 01**

Course Objectives

The objective of this course is to familiarize the prospective engineers with:

1. Design of logic gates, combinational and sequential circuits.
2. Understanding of basic components of computer architecture.
3. Understanding of assembly language code.

SYLLABUS

Practical based on CST251 syllabus

Course Outcomes

On Successful completion of course, students will be able to

1. Design and Implement combinational and sequential circuits using VHDL
2. Understand components of computer architecture and simulate assembly language code.

III Semester

Department of Computer Science & Engineering

Course Code : CST252 **Course : Data Structures and Algorithms**

L: 3 Hrs, T: 0 Hr, P: 0 Hr, Per Week **Total Credits : 03**

Course Objectives

1. To impart to students the basic concepts of data structures and algorithms.
2. To familiarize students on different searching and sorting techniques.
3. To prepare students to use linear (stacks, queues, linked lists) and non-linear (trees, graphs) data structures.
4. To enable students to devise algorithms for solving real-world problems.

SYLLABUS

UNIT I Data Structures and Algorithms Basics

Introduction: basic terminologies, elementary data organizations, data structure operations; abstract data types (ADT) and their characteristics.

Algorithms: definition, characteristics, analysis of an algorithm, asymptotic notations, time and space tradeoffs.

Array ADT: definition, operations and representations – row-major and column- major.

UNIT II Stacks and Queues

Stack ADT: allowable operations, algorithms and their complexity analysis, applications of stacks – expression conversion and evaluation (algorithmic analysis), multiple stacks.

Queue ADT: allowable operations, algorithms and their complexity analysis for simple queue and circular queue, introduction to double-ended queues and priority queues.

UNIT III Linked Lists

Singly Linked Lists: representation in memory, algorithms of several operations: traversing, searching, insertion, deletion, reversal, ordering, etc.

Doubly and Circular Linked Lists: operations and algorithmic analysis. Linked representation of stacks and queues, header node linked lists.

UNIT IV Sorting and Searching

Sorting: different approaches to sorting, properties of different sorting algorithms (insertion, Shell, quick, merge, heap, counting), performance analysis and comparison.

Searching: necessity of a robust search mechanism, searching linear lists (linear search, binary search) and complexity analysis of search methods.UNIT V Trees

Trees: basic tree terminologies, binary tree and operations, binary search tree [BST] and operations with time analysis of algorithms, threaded binary trees.

Self-balancing Search Trees: tree rotations, AVL tree and operations, B+-tree: definitions, characteristics, and operations (introductory).

UNIT VI Graphs and Hashing

Graphs: basic terminologies, representation of graphs, traversals (DFS, BFS) with complexity analysis, path finding (Dijkstra's SSSP, Floyd's APSP), and spanning tree (Prim's method) algorithms.

Hashing: hash functions and hash tables, closed and open hashing, randomization methods (division method, mid-square method, folding), collision resolution techniques.

Course Outcomes:

On completion of the course the student will be able to

1. Recognize different ADTs and their operations and specify their complexities.
2. Design and realize linear data structures (stacks, queues, linked lists) and analyze their computation complexity.
3. Devise different sorting (comparison based, divide-and-conquer, distributive, and tree-based) and searching (linear, binary) methods and analyze their time and space requirements.
4. Design traversal and path finding algorithms for Trees and Graphs.

Text Books

1. Ellis Horowitz, Sartaj Sahni & Susan Anderson-Freed, Fundamentals of Data Structures in C, Second Edition, Universities Press, 2008.
2. Mark Allen Weiss; Data Structures and Algorithm Analysis in C; Second Edition; Pearson Education; 2002.
3. G.A.V. Pai; Data Structures and Algorithms: Concepts, Techniques and Application; First Edition; McGraw Hill; 2008.

Reference Books

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein; Introduction to Algorithms; Third Edition; PHI Learning; 2009.
2. Ellis Horowitz, Sartaj Sahni and Sanguthevar Rajasekaran; Fundamentals of Computer Algorithms; Second Edition; Universities Press; 2008.
3. A. K. Sharma; Data Structures using C, Second Edition, Pearson Education, 2013.

III Semester

Department of Computer Science & Engineering

Course Code : CSP252 **Course : Data Structures and Algorithms Lab**

L: 0 Hrs, T: 0 Hr, P: 4 Hr, Per Week **Total Credits : 02**

Course Objectives

1. To enable students to employ different searching and sorting methods.
2. To prepare students to identify and apply linear (stacks, queues, linked lists) and non- linear (trees, graphs) data structures in solving problems.
3. To encourage students to design and execute tree-based algorithms for solving real- world problems.

SYLLABUS

Experiments based on CST252 Syllabus in C | C++.

Course Outcomes

On completion of the course the student will be able to

1. Design and realize different linear data structures.
2. Identify and apply specific methods of searching and sorting to solve a problem.
3. Implement and analyze operations on binary search trees and AVL trees.
4. Implement graph traversal algorithms, find shortest paths and analyze them.

Reference Books

1. K R. Venugopal and Sudeep. R Prasad; Mastering C; Second Edition; McGraw Hill; 2015.
2. Ellis Horowitz, Sartaj Sahni & Susan Anderson-Freed, Fundamentals of Data Structures in C, Second Edition, Universities Press, 2008.
3. Mark Allen Weiss; Data Structures and Algorithm Analysis in C; Second Edition; Pearson Education; 2002.

III Semester

Department of Computer Science & Engineering

Course Code : CSP253 **Course : Systems Lab I**

L: 0 Hrs, T: 0 Hr, P: 4 Hr, Per Week **Total Credits : 02**

Course Objectives

1. Introduce students with basic Python programming concepts
2. Students will learn different data structures supported by python and its applications
3. to develop complex real life python applications.

SYLLABUS

Practicals based on the following syllabus

- Python Execution model and Basic building blocks of Python Programs / Scripts / Modules
- Various keywords, Operators , control and loop constructs used in Python
- User defined Function generation in Python
- Dealing with Python files, Modules and Packages SciPy, an Open Source Python- based library, which is used in mathematics, scientific computing, Engineering, and technical computing.
- Developing small mathematical applications using packages like Numpy, Matplotlib etc.
- Introduction of with Web scrapping and its need
- Application development to scrape the web with the help of standard libraries like Requests and bs4(Beautiful Soup).

Course Outcomes

On completion of the course the student will be able to

1. Design Python programs using different data and control structures.
2. Design and use Python Files ,Modules and Packages to handle complex python programs
3. Develop mathematical and scientific applications in python using numpy, scipy libraries
4. Develop small applications for web scrapping using standard libraries.

Text Books

1. Learning Python: Poweful object oriented programming, Mark Lutz, O'REILLY publications 5th addition
2. Introduction to Computing & Problem Solving with Python Jeeva Jose and P Sojan Lal Ascher
3. Problem Solving with Algorithms and Data Structures using Python by By Brad Miller and David Ranum, 2nd addition

Reference Books

1. Allen Downey ,Jeffrey Elkner ,Chris Meyers,:Learning with Python, Dreamtech Press
2. The Python 3 Standard Library by Example (Developer's Library) by Doug Hellmann, second edition.

III Semester

Department of Computer Science & Engineering

Course Code : MAT252 **Course : Linear Algebra and Statistics**

L: 2 Hrs, T: 1 Hr, P: 0 Hr, Per Week **Total Credits : 03**

Course Objectives

The objective of this course is to familiarize the prospective engineers with techniques in linear algebra, probability and statistics. It aims to equip the students with standard concepts and tools at an intermediate to advanced level that will serve them well towards tackling more advanced level of mathematics and applications that they would find useful in their disciplines.

SYLLABUS

Module 1 (10-Lectures)

Vector Space; Subspaces; Linear Dependence/Independence; Basis; Dimension; Linear transformation; Range Space and Rank; Null Space and Nullity; Rank nullity theorem, Matrix Representation of a linear transformation; Linear Operators on R^n and their representation as square matrices; Invertible linear operators; Inverse of a non-singular matrix.

Module 2(8-Lectures)

Eigenvalues and eigenvectors of a linear operator; Inner Product Spaces, Norm; Orthonormal Sets, Gram Schmidt orthogonalisation process; projections, positive definite matrices, and Singular Value Decomposition.

Module 3(13-Lectures)

Review of Discrete and continuous random variable, joint probability function, Introduction to stochastic process, random walk, stationary and auto regressive process, transition probability Matrix, Discrete time Markov chain, Continuous time Markov chain.

Module 4(6-lectures)

Hypothesis testing for sampling distributions of means, proportions, sum and differences of means and proportions for large and small samples.

Course Outcomes

On completion of the course the student will be able to

1. Computational techniques and algebraic skills essential for the study of systems of linear equations, matrix algebra, vector spaces, eigenvalues and eigenvectors, orthogonality and diagonalization.
2. Visualization, spatial reasoning, as well as geometric properties and strategies to model, solve problems, and view solutions, especially in R^2 and R^3, as well as conceptually extend these results to higher dimensions.

3. To prepare the background of students to pursue statistical theory or methodology and analyze data in any stream of computer science and information technology.

Text Books

1. Hoffman and Kunze : Linear Algebra, Prentice Hall of India, New Delhi
2. Gilbert Strang : Linear Algebra And Its Applications (Paperback) , Nelson Engineering (2007)
3. M R. Spiegal : Theory and Problems of probability and statistics :,2nd ed, Schaum series.

Reference Books

1. Seymour Lipschutz et al: Linear Algebra, 3rd ed: Schaum series.
2. V. Krishnamoorthy et al : An introduction to linear algebra , Affiliated East West Press, New Delhi P.G. Bhattacharya, S.K. Jain and S.R.
3. Nagpaul : First course in Linear Algebra, Wiley Eastern Ltd., New Delhi
4. K.B.Datta : Matrix and Linear Algebra, Prentice Hall of India, New Delhi
5. W. Feller, An Introduction to Probability Theory and its Applications, Vol. 1, 3rd Ed., Wiley, 1968.
6. S. Ross, A First Course in Probability, 6th Ed., Pearson Education India, 2002.

III Semester
Department of Computer Science & Engineering

Course Code : HUT253 **Course : Business Communication**

L: 3 Hrs, T: 0 Hr, P: 0 Hr, Per Week **Total Credits : 03**

SYLLABUS

UNIT I Fundamentals of Business Communication

Definition of communication and business communication, Objectives of Business Communication, Audience recognition, Barriers of Communication, Product Promotion, Usage of Social Media, Negotiation Skills, Persuasive Communication, PAC concept.

UNIT II Technical Writing

Process of Technical Writing, Types of Technical Writing. Letters: Job application, Job Description and CV, enquiry, complaint, order, follow-up, cover/transmittal letters, and e-mails. Writing to Persuade: Proposals and Sales Letters. Other Forms of Technical Writing: Notices, Circulars, Memos, Organizational announcements, Minutes of Meeting.

UNIT III Grammar for Writing

Functional Grammar: Punctuations, Mechanics, Active/ Passive, Transformation of Sentences, Subject-Verb Agreement, Articles, Prepositions.

UNIT IV Business Reports

Basic formats and types (Annual, Progress, Project (Project Charter, Project Timeline), Market Search, Sales, Feasibility/Recommendation), Case Study evaluation.

UNIT V Preparation of Documents

Visual Appeal: Document Design, Graphics, Tables, User Manuals, Brochures, Fliers.

UNIT VI Effective Oral Communication

Non-Verbal Communication, Public speaking, Presentation, Group Discussion.

Course Outcomes

1. Students will understand the fundamentals and objectives of business communication, and role of audience in effective communication.
2. Students will develop technical writing skills and produce effective workplace documents.
3. Students will learn the application of grammar in writing.
4. Students will develop skills to enhance visual appeal of documents.
5. Students will understand strategies for effective oral communication for professional needs.

Text Books

1. Sharon Gerson, Steven Gerson, “Technical Communication: Process and Product”, 2018, Pearson
2. Sanjay Kumar, Pushpa Lata, Communication Skills, 2nd Edition, Oxford Publication, 2018.

Reference Books

1. Vikram Bisen, Business Communication, New Age International Publishers, 2010.
2. P.D. Chaturvedi and Mukesh Chaturvedi, Fundamentals of Business Communication, Pearson Publications, 2012.
3. William Strunk Jr. and The Elements of Style, Allyn & Bacon - a Pearson Education Company.

III semester
Department of Computer Science & Engineering

Course Code : HUT257 **Course : Cyber Laws and Ethics in IT**

L: 2 Hrs, T: 0 Hr, P: 0 Hr, Per Week **Total Credits : 02**

Course Objectives

1. Describe laws governing cyberspace and analyze the role of Internet Governance in framing policies for Internet security
2. Identify intellectual property right issues in the cyberspace and design strategies to protect your intellectual property
3. Understand the importance of freedom of expression, defamation and hate speech in cyber world.
4. Recognize the importance of digital divide, contingent workers and whistle blowing situations.

SYLLABUS

UNIT I

Cyber laws and rights in today's digital age; IT Act, Intellectual Property Issues connected with use and management of Digital Data, Emergence of Cyberspace, Cyber Jurisprudence.

UNIT II

Cyber Crimes against Individuals, Institution and State, Hacking, Digital Forgery, Cyber Stalking/Harassment, Cyber terrorism, Cyber Defamation, Different offences under IT Act, 2000, Cyber Torts.

UNIT III

Ethics in business world, Ethic s in IT, Ethics for IT professionals and IT users, IT professional malpractices, communications eavesdropping, computer break-ins, denial-of-service, destruction and modification of data, distortion and fabrication of information, Types of Exploits and Perpetrators.

UNIT IV

Intellectual Property: Copy rights, Patents, Trade Secret Laws, Key Intellectual property issues, Plagiarism, Competitive Intelligence, Cybersquatting, Information warfare policy and ethical Issues.

UNIT V

Privacy: The right of Privacy, Protection, Key Privacy and Anonymity issues, Identity Theft, Consumer Profiling, Defamation, Freedom of Expression, Anonymity, National, Security Letters, Defamation and Hate Speech.

UNIT VI

Ethics of IT Organization: Contingent Workers H- IB Workers, Whistle- blowing, Protection for Whistle- Blowers, Handling Whistle- blowing situation, Digital divide.Course Outcomes:

On successful completion of the course, students will be able

1. To identify and analyze statutory, regulatory, constitutional, and organizational laws that affects the software professional.
2. To understand carious cyber laws with respect to legal dilemmas in the Information Technology field.
3. To interpret various intellectual property rights, Privacy, Protection issues in software development field.
4. To understand role of ethics in IT organization.

Text Books

1. George Reynolds, "Ethic s in information Technology",5th edition, Cengage Learning
2. Hon C Graff, Cryptography and E-Commerce - A Wiley Tech Brief, Wiley Computer Publisher, 2001.

Reference Books

1. Michael Cross, Norris L Johnson, Tony Piltzecker, Security, Shroff Publishers and Distributors Ltd.
2. Debora Johnson," Computer Ethic s",3/e Pearson Education.
3. Sara Baase, "A Gift of Fire: Social, Legal and Ethical Issues, for Computing and the Internet," PHI Public at ions.
4. Chris Reed & John Angel, Computer Law, OUP, New York, (2007).

IV Semester
Department of Computer Science & Engineering

Course Code : CST254 **Course : Discrete Mathematics and Graph Theory**

L: 3 Hrs, T: 1Hr, P: 0 Hr, Per Week **Total Credits : 04**

Course Objectives

1. To teach students predicates, quantifiers, and logical connectives.
2. To derive the solution using deductive logic and evaluate Boolean functions using the properties of Boolean algebra.
3. To develop the given problem using techniques of graph theory.

SYLLABUS

UNIT I

Relation and Function: Basic concepts of Set theory, Power set, some operations on Sets, Venn diagram, some basic set identities, Cartesian products. Properties of binary relation in a set, Relation matrix and the graph of the relation, Partition and covering of a set. Equivalence relations, Compatibility relations, Compositions of binary relations. Definition and composition of functions, inverse functions and characteristic functions of a set.

UNIT II

Mathematical Logic: Statement and notations, connectives, Negation, conjunction, disjunction, conditional & bi-conditional, statement formulas & truth tables.

Tautologies, equivalence of formulas, Duality law, Tautological implications. Normal Forms – Principal disjunctive and principal conjunctive normal forms. Theory of inference for statement calculus. Theory of inference for predicate calculus.

UNIT III

Algebraic structures: Semi groups, monoids definition and examples, Group definitions and examples, cyclic group, permutation groups, subgroups and homomorphism, co-sets, Lagrange's theorem and Normal subgroup.

UNIT IV

Rings and Field: Ring definition and examples, sub rings, Ring homomorphism, ideals and Quotient rings, polynomial rings. Finite field, Galois field, Integral domain.

UNIT V

Lattice theory and Boolean algebra: Lattices as partially ordered set, Definitions and Examples, some properties of Lattices, Lattices as algebraic system, sub lattices, direct product, homomorphism, some special Lattices. Boolean algebra: Definitions and examples, Application of Boolean Algebra to switching circuits.

UNIT VI

Graphs and Trees : Graphs and their properties, Degree, Connectivity, Path, Cycle, Sub-Graph, Isomorphism, Eulerian and Hamiltonian Walks, Graph Colouring, Colouring maps and Planar Graphs, Colouring Vertices, Colouring Edges, List Colouring, Perfect Graph, definition properties and Example, rooted trees, trees and sorting, weighted trees and prefix codes, Bi-connected component and Articulation Points, Shortest distances.

Course Outcomes

On successful completion of the course, students will be able

1. Analyze given logic sentence in terms of predicates, quantifiers, and logical connectives.
2. Derive the solution using deductive logic and prove it based on logical inference.
3. Classify the algebraic structure for a given mathematical problem
4. Evaluate Boolean functions and simplify expressions using the properties of boolean algebra.
5. Develop the given problem as graph networks and solve with techniques of graph theory.

Text Books

1. J. P.Tremblay and R. Manohar; Discrete Mathematical Structures with Applications to Computer Science; Tata McGraw-hill Publication 1997.
2. Babu Ram; Discrete Mathematics; Pearson Education, 2011.
3. C.L. Liu and D.P. Mohapatra, Combinatorial Mathematics, 3rd edition Tata McGraw Hill..

Reference Books

1. Kenneth H. Rosen, Discrete Mathematics and its Applications, Tata McGraw – Hill
2. Susanna S. Epp, Discrete Mathematics with Applications, 4th edition, Wadsworth Publishing Co. Inc.
3. C L Liu and D P Mohapatra, Elements of Discrete Mathematics A Computer Oriented Approach, 3rd Edition by, Tata McGraw – Hill.

IV Semester

Department of Computer Science & Engineering

Course Code : CST255 **Course : Operating Systems**

L: 3 Hrs, T: 0 Hr, P: 0 Hr, Per Week **Total Credits : 03**

Course Objectives

1. To learn the mechanisms of OS to handle processes and threads and their communication
2. To learn the mechanisms involved in memory management in contemporary OS
3. To gain knowledge on distributed operating system concepts that includes architecture, Mutual exclusion algorithms, deadlock detection algorithms and agreement protocols
4. To know the components and management aspects of concurrency management.

SYLLABUS

UNIT I

Introduction: Concept of Operating Systems, Generations of Operating systems, Types of Operating Systems, OS Services, System Calls, Structure of an OS - Layered, Monolithic, Microkernel Operating Systems, Concept of Virtual Machine. Case study on UNIX and WINDOWS Operating System.

UNIT II

Processes: Definition, Process Relationship, Different states of a Process, Process State transitions, Process Control Block (PCB), Context switching

Thread: Definition, Various states, Benefits of threads, Types of threads, Concept of multithreads

Process Scheduling: Foundation and Scheduling objectives, Types of Schedulers, Scheduling criteria: CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time; Scheduling algorithms: Pre-emptive and Non pre-emptive, FCFS, SJF, RR; Multiprocessor scheduling: Real Time scheduling: RM and EDF..

UNIT III

Inter-process Communication: Critical Section, Race Conditions, Mutual Exclusion, Hardware Solution, Strict Alternation, Peterson's solution, Producer\Consumer Problem, Semaphores, Monitors, Message Passing, Classical IPC Problems: Reader's & Writer Problem, Dinning Philosopher Problem etc.

UNIT IV

Deadlocks: Definition, Necessary and sufficient conditions for Deadlock, Deadlock Prevention, Deadlock Avoidance: Banker's algorithm, Deadlock detection and Recovery.UNIT V

Memory Management: Basic concept, Logical and Physical address map, Memory allocation: Contiguous Memory allocation – Fixed and variable partition–Internal and External fragmentation and Compaction, Paging: Principle of operation – Page allocation – Hardware support for paging, Protection and sharing, Advantages & Disadvantages of paging.

Virtual Memory: Basics of Virtual Memory – Hardware and control structures – Locality of reference, Page fault, Working Set, Dirty page/Dirty bit – Demand paging, Page Replacement algorithms: Optimal, First in First Out (FIFO), Second Chance (SC), Not recently used (NRU) and Least Recently used (LRU).

UNIT VI

I/O Hardware: I/O devices, Device controllers & Device drivers, Secondary-Storage Structure: Disk structure, Disk scheduling algorithms

File Management: Concept of File, Access methods, File types, File operation, Directory structure, File System structure, Allocation methods, Free-space management, directory implementation, efficiency and performance.

Disk Management: Disk structure, Disk scheduling - FCFS, SSTF, SCAN, C-SCAN, LOOK, C-LOOK, Disk reliability, Disk formatting, Boot-block, Bad blocks.

Course Outcomes

On successful completion of the course, students will be able to demonstrate

1. Ability to describe the general architecture of computers, Contrast and compare differing structures for operating systems.
2. Ability to understand and analyze theory and implementation of processes and schedulers.
3. Ability to understand and design resource control (synchronization and deadlock).
4. Ability to apply knowledge of physical and virtual memory, scheduling, I/O and files.

Text Books

1. Operating System Concepts, 8th Edition by A. Silberschatz, P.Galvin, G. Gagne, Wiley India Edition.
2. Modern Operating Systems, 2nd Edition by Andrew Tanenbaum, PHI.

Reference Books

1. Operating Systems: Internals and Design Principles, 5th Edition, William Stallings, Prentice Hall of India.
2. Understanding the Linux Kernel, 3rd Edition, Daniel P. Bovet, Marco Cesati, O'Reilly.

IV Semester

Department of Computer Science & Engineering

Course Code : CSP255 **Course : Operating Systems Lab**

L: 0 Hrs, T: 0 Hr, P: 4 Hr, Per Week **Total Credits : 02**

Course Objectives

Using C language in Linux environment

1. To develop ability of students to design and implement concepts of operating systems such as system calls, CPU scheduling, process/thread management.
2. To develop the components and management aspects of concurrency management, memory management, and File management.

SYLLABUS

Experiments based on CST255 Syllabus.

Course Outcomes

On completion of the course the student will be able to demonstrate

1. Ability to use LINUX system calls and implement system commands.
2. Ability to implement process and process schedulers.
3. Ability to design and implement solution to handle synchronization and deadlock.
4. Ability to implement memory and File management algorithms.

IV Semester

Department of Computer Science & Engineering

Course Code : CST256 **Course : Object Oriented Programming**

L: 3 Hrs, T: 0 Hr, P: 0 Hr, Per Week **Total Credits : 03**

Course Objectives

1. To make students understand Fundamental features of an object oriented language like Java: object classes and interfaces, exceptions and libraries of object collections
2. Introduce students with fundamental concepts like exception handling, generics, multithreading and streams.

SYLLABUS

UNIT I

Features of Object Oriented Programming languages, Abstraction, Encapsulation, Inheritance, polymorphism and late binding.Concept of a class, Access control of members of a class, instantiating a class, constructor and method overloading.

UNIT II

Concept of inheritance, methods of derivation, use of super keyword and final keyword in inheritance, run time polymorphism, abstract classes and methods, Interface, implementation of interface, creating packages, importing packages, static and non-static members, Lambda Expressions Introduction, Block, Passing Lambda expression as Argument.

UNIT III

Exceptions, types of exception, use of try catch block, handling multiple exceptions, using finally, throw and throws clause, user defined exceptions, Introduction to streams, byte streams, character streams, file handling in Java, Serialization.

UNIT IV

Generics, generic class with two type parameter, bounded generics. Collection classes: Arraylist, LinkedList, Hashset, Treeset .

UNIT V

Multithreading: Java Thread models, creating thread using runnable interface and extending Thread, thread priorities, Thread Synchronization, InterThread communications.

Basic SQL commands, DDL and DML commands, Java Database Connectivity, Working with Connection, Statement and Resultset, Data Manipulation using JDBC, Data navigation.UNIT VI

Introduction to Design Patterns, Need of Design Pattern, Classification of Design Patterns, Role of Design Pattern in Software design, Creational Patterns, Structural Design Patterns and Behavioral Patterns.

Course Outcomes

On successful completion of the course, students will be able to demonstrate

1. Understand the principles of object-oriented programming; create classes, instantiate objects and invoke methods.
2. Understand concept of generics and implement collection classes. Use exception handling mechanism.
3. Efficiently work with streams, use multithreading for solving classic synchronization problems. Perform java database connectivity and execute basic SQL commands.
4. Understand characteristics and need of Design Pattern in Software Design Process.

Text Books

1. Herbert Schildt; JAVA The Complete Reference; Ninth Edition, Tata McGraw- Hill Publishing Company Limited.
2. Design Patterns By Erich Gamma, Pearson Education.

Reference Books

1. Cay S. Horstmann and Gary Cornell; Core JAVA Volume-II Advanced Features; Eighth Edition; Prentice Hall, Sun Microsystems Press 2008.
2. Herbert Schildt and Dale Skrien; Java Fundamentals A Comprehensive Introduction; Tata McGraw- Hill Education Private Ltd 2013.

IV Semester

Department of Computer Science & Engineering

Course Code : CSP256 **Course : Object Oriented Programming Lab**

L: 0 Hrs, T: 0 Hr, P: 2 Hr, Per Week **Total Credits : 01**

Course Objectives

1. To develop ability of students to implement basic concepts and techniques of object oriented programming paradigm like encapsulation, inheritance, polymorphism, exception handling.
2. Develop solution to problems using collection classes, generics, streams, multithreading and JDBC.

SYLLABUS

Experiments based on CST256 Syllabus.

Course Outcomes

On completion of the course the student will be able to

1. Design solution to problems using concepts of object oriented programming like classes, objects, inheritance with proper exception handling.
2. Use collection classes, generic classes to design programs and perform database connectivity.

Implement programs based on streams and multithreading.

IV Semester

Department of Computer Science & Engineering

Course Code : CST257 **Course : Formal Language and Automata Theory**

L: 3 Hrs, T: 0 Hr, P: 0 Hr, Per Week **Total Credits : 03**

Course Objectives

1. To provide students an understanding of basic concepts in the theory of computation.
2. To teach formal languages and various models of computation.
3. To exhibit fundamental concepts related with computability theory.

SYLLABUS

UNIT I

Basics of Sets and Relation, Countability and Diagonalisation, Principle of mathematical induction, Pigeon-hole principle. Fundamentals of formal languages and grammars, Chomsky hierarchy of languages.

UNIT II

Finite automata: Deterministic finite automata (DFA), Nondeterministic finite automata (NFA) and equivalence with DFA, Minimization of finite automata, NFA with Epsilon Transitions, Finite Automata with output.

UNIT III

Regular expressions and Regular languages, Regular grammars and equivalence with finite automata, properties of regular languages, pumping lemma for regular languages, Context-free grammars (CFG) and language(CFL), parse trees, ambiguity in CFG, Reduction of CFGs, Chomsky and Greibach normal forms.

UNIT IV

Push Down Automata: Deterministic pushdown automata and Non-Deterministic pushdown automata, Acceptance by two methods: Empty stack and Final State, Equivalence of PDA with CFG, closure properties of CFLs.

UNIT V

Turing machines: The basic model for Turing machines (TM), Turing recognizable (recursively enumerable) and Turing-decidable (recursive) languages, variants of Turing machines, unrestricted grammars and equivalence with Turing machines, TMs as enumerators.

UNIT VI

Undecidability: Church-Turing thesis, Universal Turing machine, Undecidable problems about languages, Recursive Function Theory.Course Outcomes:

On successful completion of the course, students will be able to demonstrate

1. Describe the formal relationships among machines, languages and grammars.
2. Design and Optimize finite automata for given regular language.
3. Design Push Down Automata, Turing Machine for given languages.
4. Demonstrate use of computability, decidability, recursive function theory through problem solving.

Text Books

1. John E. Hopcroft, Rajeev Motwani and Jeffrey D. Ullman, Introduction to Automata Theory, Languages, and Computation, Pearson Education Asia.

Reference Books

1. Harry R. Lewis and Christos H. Papadimitriou, Elements of the Theory of Computation, Pearson Education Asia.
2. Dexter C. Kozen, Automata and Computability, Undergraduate Texts in Computer Science, Springer.
3. Michael Sipser, Introduction to the Theory of Computation, PWS Publishing.
4. John Martin, Introduction to Languages and The Theory of Computation, Tata McGraw Hill.

IV Semester

Department of Computer Science & Engineering

Course Code : CST258 **Course : System Programming and Device Driver**

L: 3 Hrs, T: 0 Hr, P: 0 Hr, Per Week **Total Credits : 03**

Course Objectives

1. To introduce the concepts and principles of system programming.
2. To introduce the procedure for designing and implementing system programs like assembler, macroprocessor, loader etc.
3. To introduce BIOS and DOS interrupts.
4. To develop simple utilities including device drivers and advance input/output.

SYLLABUS

UNIT I

Assembler- Introduction to System Programming & its components, Machine Architecture, Basic Assembler functions, Machine dependent & Machine Independent Assembler Features, Assembler Design, design of single pass and multi pass Assembler.

UNIT II

Macroprocessor - Basic Macro Processor Functions, Machine Independent Macro Processor Features, Design of macro processor.

UNIT III

Linker and Loader - Basic Loader Functions, Concept of static and dynamic relocation, external symbols,Machine dependent & Machine Independent Loader Features, Loader Design Options.

UNIT IV

Video and Keyboard Operation – Introduction to Video and Keyboard Processing, Video Systems, Keyboard Processing.

UNIT V

Advanced Input/Ouput: Facilities for using the Mouse, Disk storage I: organization, Disk storage II: Writing and Reading Files Disk Storage III: INT 21H functions for Support Disks and Files, Disk Storage IV: INT 13H Disk Functions, Facilities for Printing.

UNIT VI

UNIX Device Drivers - Definition, Anatomy and Types, Device Programming, Installation and Incorporation of driver routines, Basic device driver operation, Implementation with Line Printer, Comparative study between device drivers for UNIX & Windows, TSR Programming.

Course Outcomes

On successful completion of the course, students will be able to demonstrate

1. Understand the purpose of system software such as assembler, macro-processor, linker, loader, device driver.
2. Apply general design procedure for designing various system software.
3. Understand the use of video and keyboard Operation.
4. Understand the operation of advanced Input/Ouput devices.

Text Books

1. Leland L. Beck; System Software: An introduction to systems programming; Pearson Education; 3rd edition; 1997.
2. D. M. Dhamdhere; System Programming and Operating systems; Tata McGraw Hill Education; 2nd edition; 1999.
3. George Pajari; UNIX Device Drivers; Pearson Education; 1993.
4. Peter Abel,Assembly Language Programming,5th Edition ,Pearson Education,2003.
5. Rajesh K. Maurya; System Programming; Dreamtech Press,2011.

Reference Books

1. J. J. Donovan; System Programming; Tata McGraw Hill Education; 2011 reprint
2. Sivarama P. Dandmudi, Introduction to Assembly Language Programming,Springer 1st Edition 2003.
3. Keringham and Pike; UNIX programming Environment; PHI, 1984.

IV Semester

Department of Computer Science & Engineering

Course Code : CSP258 | **Course : System Programming and Device Driver Lab**

L: 0 Hrs, T: 0 Hr, P: 2 Hr, Per Week | **Total Credits : 01**

Course Objectives

1. To learn the principles of processing of system software programs.
2. To acquire skills of system programming utility tools.

SYLLABUS

Experiments based on CST258 Syllabus.

Course Outcomes

On completion of the course the student will be able to

1. Design and implementation of system software.
2. Understand the use of various system utilities and tools.
3. Understand the use of various device drivers in UNIX.

Text Books

1. Leland L. Beck; System Software: An introduction to systems programming; Pearson Education; 3rd edition; 1997
2. Peter Abel,Assembly Language Programming,5th Edition ,Pearson Education,2003
3. Sivarama P. Dandmudi, Introduction to Assembly Language Programming, Springer 1st Edition 2003.

Reference Books

1. J. J. Donovan; System Programming; Tata McGraw Hill Education; 2011 reprint
2. Keringham and Pike; UNIX programming Environment; PHI, 1984.

IV Semester

Department of Computer Science & Engineering

Course Code : CSP259 **Course : Systems Lab II**

L: 0 Hrs, T: 0 Hr, P: 4 Hr, Per Week **Total Credits : 02**

Course Objectives

Throughout the course, students will be expected to learn Python Language basics to do the following:

1. Understand UI/UX basics and its use in software industry
2. Understand basic use cases of UI/UX.
3. Develop small utilities using UI/UX tools
4. Develop and integrate UI/UX with basic programs.

SYLLABUS

Programs based on:

1. Illustration tool box
2. Storytelling and typography tools
3. UX writing and AR/VR tools
4. Voice technology tools
5. Motion Design, Animated graphics

Course Outcomes

On completion of the course the student will be able to

1. Design UI/UX use cases using Illustration tool box
2. Design and use storytelling and typography for requirement specification.
3. Use UX writing, AR and VR models to develop interfaces for use cases
4. Develop small applications using voice technology, motion design, animation.

Text Books

1. UI/UX design for designer and developers: by Nathan Clark
2. User Story Mapping software for agile age [Paid subscription on yearly basis]
3. User story mapping by Jeff Patton, O'Reilly Publication.

IV Semester

Department of Computer Science & Engineering

Course Code : CHT252 **Course : Environmental Science**

L: 2 Hrs, T: 0 Hr, P: 0 Hr, Per Week **Total Credits : 00**

SYLLABUS

Principle of contaminant behaviour and recent trends in environmental pollution control.

I- Air pollution and its control techniques: (4 lectures)

Contaminant behaviour in the environment, Air pollution due to SOx, NOx, photochemical smog, Indoor air pollution

Natural pathways for degradation: Carbon cycle, Sulphur cycle, Nitrogen cycle, Oxygen cycle.

Factors responsible for altering the composition of atmosphere (deforestation, burning of fossil fuels, industrial and vehicular emissions, CFCs).

Techniques to control Air pollution, ambient air quality and continuous air quality monitoring, Control measures at source, Kyoto Protocol, Carbon Credits.

II- Noise pollution and its control techniques: (2 lectures)

Introduction to noise pollution and its causes

Noise pollution control: Recent advances in noise pollution control and benefits.

III- Soil pollution and its control techniques: (5 lectures)

Soil pollution: Soil around us, Soil water characteristics, soil pollution.

Solid waste management: Composting, vermiculture, landfills, hazardous waste treatment, bioremediation technologies, conventional techniques (land farming, constructed wetlands), and phytoremediation.

Degradation of xenobiotics in environment: Petroleum hydrocarbons, pesticides, heavy metals

IV- Water pollution and its control techniques: (8 lectures)

Major sources of water pollution: Eutrophication, acid mine drains, pesticides and fertilizers, dyeing and tanning, marine pollution, microplastics

Techniques to control water pollution: Conventional waste water treatment-types of sewage, sewerage system, alternative systems, primary, secondary and tertiary processes including aerobic and anaerobic techniques, safe disposal.

Case studies: Treatment schemes for waste water from dairy, textile, power plants, pharmaceutical industries, and agro based industries such as rice mills

V- E-wastes (2 lectures)

Introduction, types of e-wastes, environmental impact, e-waste recycling, e-waste management rules.

VI- Environmental Sustainability: Role of Green technology (5 lectures)

Concept of green technologies, categories, goals and significance, sustainability Green energy, green chemistry, challenges to green technology, advantage and disadvantages of green processes, Eco mark certification- its importance and implementationVII- Different government initiatives (2 lectures)

National ambient air quality standard 2009, Swacch Bharat Abhiyan, National afforestation program and Act- 2016, National river conservation plan, Formation of National Green Tribunal

Course Outcomes

On successful completion of the course, students

1. Will get sufficient knowledge regarding different types of environmental pollutions, their causes, detrimental effects on environment and effective control measures.
2. Will realize the need to change an individual's outlook, so as to perceive our environmental issues correctly, using practical approach based on observations and self-learning.
3. Will become conversant with recent waste management techniques such as E-wastes, its recycling and management.
4. Will gain knowledge about the modes for sustainable development, importance of green energy and processes.
5. Will be able to identify and analyze environmental problems as well as risks associated with these problems and greener efforts to be adopted, to protect the environment from getting polluted.

Suggested Books

1. Benny Joseph, Environmental Studies, Mc Graw Hill Education (India) Private Limited
2. B. K. Sharma, Environmental Chemistry, Goel Publishing House, Meerut
3. P Aarne Vesilind, J. Jeffrey Peirce and Ruth F. Weiner, Environmental Pollution and Control, Butterworth-Heinemann
4. D. D. Mishra, S. S. Dara, A Textbook of Environmental Chemistry and Pollution Control, S. Chand & Company Ltd. Sultan Chand & Company
5. Shree Nath Singh, Microbial Degradation of Xenobiotics, Springer-Verlag Berlin Heidelberg
6. P.T. Anastas & J.C. Warner, Green Chemistry: Theory & practice, Oxford University Press
7. P. Thangavel & Sridevi, Environemental Sustainability: Role of Green technologies, Springer publications.

IV Semester

Department of Computer Science & Engineering

Course Code : CST299-1 **Course : Java Programming and UI design Concepts**

L: 3 Hrs, T: 0 Hr, P: 0 Hr, Per Week **Total Credits : 03**

Course Objectives

This course provides an introduction to object oriented programming (OOP) using the Java programming language.

1. To Gain knowledge about basic Java language syntax and semantics to construct Java programs and compose the concepts such as variables, conditional and iterative execution methods etc.
2. To learn the fundamentals of object-oriented programming in Java, including describing classes, objects, invoking methods.
3. To familiarize the students with concept of packages and exception handling mechanisms.
4. To teach the basics concepts of streams, threads and User Interface designing.

SYLLABUS

UNIT I

Introduction to java programming, Features of Object Oriented Programming languages like data encapsulation, inheritance, polymorphism and late binding. Fields and Methods, Concept of a class, Access control of members of a class, instantiating a class.

UNIT II

Concept of constructor and method overloading, inheritance, use of super keyword and final keyword in inheritance, run time polymorphism. Abstract classes and methods, interface, implementation of interface, static and non-static members.

UNIT III

Creating packages, importing packages, Exceptions, types of exception, use of try catch block, handling multiple exceptions, multiple catch clauses, Nested try Statements, using finally, throw and throws clause, Built-in Exception, user defined exceptions.

UNIT IV

Collection classes: Arrays, Vectors, Array list, Introduction to streams, byte streams, character streams, file handling in Java, Basics of Thread programming, Sleep, Join.UNIT V

Introduction to Java User Interface, Concepts of Applet Class, Applet architecture, Applet Display methods, Parameter passing to applet. Event Handling in Applet.

UNIT VI

Swing Components and Containers, JLabel, JTextField, JList, JCombobox, Swing Buttons (button, toggle button, checkbox, radio button), JTable, JTabbedPane Event handling mechanism, Event Classes, Event Listener interfaces, delegation event model, adapter classes.

Course Outcomes

On successful completion of the course, students will be able to demonstrate

1. Understand the principles of object-oriented programming, create classes, instantiate objects, invoke methods.
2. Understand concept of generics and collection classes.
3. Understand the use of exception handling mechanism and handle character and byte streams.
4. Design User Interface and implement threads and perform basic operations on Threads.

Text Books

1. Herbert Schildt; JAVA The Complete Reference; Seventh Edition, Tata McGraw- Hill Publishing Company Limited 2007.
2. Cay S. Horstmann and Gary Cornell; Core JAVA Volume-II Advanced Features; Eighth Edition; Prentice Hall, Sun Microsystems Press 2008.

Reference Books

1. Herbert Schildt and Dale Skrien; Java Fundamentals A Comprehensive Introduction; Tata McGraw- Hill Education Private Ltd 2013.

IV Semester

Department of Computer Science & Engineering

Course Code : CST299-2 **Course : Design Thinking for Innovation**

L: 3 Hrs, T: 0 Hr, P: 0 Hr, Per Week **Total Credits : 03**

Course Objectives

This course aspects at how individuals and organizations use creativity and design thinking skills to identify and choose opportunities that enable innovation.

1. To develop creative problem solving skills and enhance these skills through a range of real world activities.
2. To provide an overview of design thinking tools to help students understand design thinking as a problem solving approach.

SYLLABUS

UNIT I

What is Design Thinking? Why is Design Thinking Important? How is Design Thinking Different? Four Questions Animation : Four Questions and What Is, What If, What Wows, What Works? Kingwood Intro: Kingwood and What Is, Kingwood and What If, What Wows, What Works, Dealing with Complex Social Systems, Visualization, Six Abilities That Matter Most.

UNIT II

Introduction to What Is : Kingwood Trust: Ethnography Part, Monash University Medical Centre Intro, Monash Looks at What Is, Journey Maps and Unmet Customer Needs, Four Things to Know About Journey Maps, Journey Map Example: The Whole Aquarium, More Journey Map Examples, Journey Maps: My Favorite Tools, Assessing and Expanding Your Repertoire.

UNIT III

Introduction to What If? : The Challenge of Possibility: Geoffrey and George, Part 1, The Challenge of Possibility: Geoffrey and George, Part 2, Intro to Iveragh, Kerry Part, Iveragh and What If, Kerry Part, The FDA Intro, FDA Design Thinking Description, Stakeholder Mapping, Stakeholder Example, Storytelling.

UNIT IV

Intro to What Wows: Assumption Testing, Whiteriver Part 1: First Assumptions, Whiteriver Part 2: Reassessing Assumptions, Intro to What Wows: Prototyping, MasAgro: Creating a Space for Experimentation, MasAgro's Hub System, MasAgro's Innovation Network, Intro to What Works, Monash Learning Launch, Monash Initiative: Long Patient Stays, Integrating DT Across Monash Organization, Reviewing Our Lessons, Actions for Impact.

UNIT V

Indian Health Service (IHS) Hospital Check-in Redesign, Agile Psychological Medicine Clinic Design Thinking Opportunity: Assignment Overview, Design Thinking Opportunity: Assignment Rubric, Identifying a Design Thinking Opportunity.

Course Outcomes

On successful completion of the course, students will be able to demonstrate

1. Apply creative problem solving skills to solve complex problems.
2. Prepare to see and take action when opportunity arises.
3. Use design thinking processes to generate innovative ideas.

Text Books

3. Herbert Schildt; JAVA- The Complete Reference; Seventh Edition, Tata McGraw- Hill Publishing Company Limited 2007.
4. Cay S. Horstmann and Gary Cornell; Core JAVA Volume-II Advanced Features; Eighth Edition; Prentice Hall, Sun Microsystems Press 2008.

Reference Books

1. https://www.coursera.org/learn/uva-darden-design-thinking-innovation
2. https://www.edx.org/course/design-thinking-and-creativity-for-innovation-2
3. https://www.ideou.com/pages/design-thinking

V Semester

Department of Computer Science & Engineering

Course Code : CST351 **Course : Database Management Systems**

L: 3 Hrs, T: 0 Hr, P: 0 Hr, Per Week **Total Credits : 03**

Course Objectives

1. To understand the role of a database management system in an organization.
2. To construct simple and advanced database queries using a data language.
3. To understand and apply logical database design principles and database normalization.
4. To recognize the need for transaction management and query processing.

SYLLABUS

UNIT I Introduction to Database System Concepts and Architecture

Databases and Database Users, Characteristics of the Database Approach, Advantages of Using the DBMS Approach, When Not to Use a DBMS, Data Models, Schemas, and Instances, Three-Schema Architecture and Data Independence, Database Languages and Interfaces, The Database System Environment. Introduction to NoSQL databases and In-Memory databases.

UNIT II The Relational Data Model and SQL

Relational Model Concepts, Relational Model Constraints and Relational Database Schemas, Update Operations, Transactions, and Dealing with Constraint Violations, SQL Data Definition, Data Types and Constraints, Data Management in SQL, Transforming ER Model into Relational Model.

UNIT III Database Design and Normalization

Functional Dependencies, Inference Rules, Equivalence, and Minimal Cover, Properties of Relational Decomposition, Normal Forms Based on Primary Keys, General Definitions of Second and Third Normal Forms, Boyce-Codd Normal Form, Other Dependencies and Normal Forms.

UNIT IV Indexing and Hashing

Ordered Indices, B+-Tree Index Files and its Extensions, Static Hashing and Dynamic Hashing, Comparison of Ordered Indexing and Hashing, Bitmap Indices, Some General Issues Concerning Indexing.UNIT V Query Processing and Optimization

Measures of Query Cost, Query Operation: Selection, Sorting and Join Operation, Transformation of Relational Expressions, Estimating Statistics of Expression Results, Choice of Evaluation Plans.

UNIT VI Transaction Processing, Concurrency Control and Recovery

Introduction to Transaction Processing, Characterizing Schedules Based on Recoverability, Characterizing Schedules Based on Serializability, Two-Phase Locking Techniques for Concurrency Control, Deadlock Handling and Multiple Granularity, Database Recovery Techniques.

Course Outcomes

On completion of the course the student will be able to

1. Identify the basic concepts and various data model used in database design.
2. Recognize the use of normalization and functional dependency.
3. Understand the purpose of query processing and optimization.
4. Apply and relate the concept of transaction, concurrency control and recovery in database.

Text Books

1. Abraham Silberschatz, Henry F. Korth and S. Sudarshan; "Database System Concepts" Sixth Edition, Tata McGraw Hill, 2011.
2. Ramez Elmasri and Shamkant Navathe; "Fundamentals of Database Systems", Sixth Edition, Addison Wesley 2011.

Reference Books

1. Raghu Ramakrishnan and Johannes Gehrke; "Database Management Systems"; Third Edition; Tata McGraw Hill Publication, 2003.
2. C. J. Date; "Database in Depth – Relational Theory for Practitioners"; O`Reilly Media, 2005.

V Semester

Department of Computer Science & Engineering

Course Code : CSP351 **Course : Database Management Systems Lab**

L: 0 Hrs, T: 0 Hr, P: 4 Hr, Per Week **Total Credits : 02**

Course Objectives

1. To enable students to use DDL, DML and DCL.
2. To prepare students to conceptualize and realize database objects (tables, indexes, views and sequences) and execute SQL queries.
3. To encourage students to design and execute PL/SQL blocks and triggers.

SYLLABUS

Experiments based on CST351 Syllabus in Oracle 11g | MySQL.

[Few experiments to be conducted to demonstrate handling of databases on cloud]

Course Outcomes

On completion of the course the student will be able to

1. Understand the use of database languages such as DDL, DML, and DCL.
2. Construct simple, nested, multiple table, and advanced queries for data retrieval.
3. Construct PL-SQL block structure and Trigger for specific application.
4. Implement various integrity constraints, views, sequences, indices and synonym on database.

Reference Books

1. James Groff, Paul Weinberg and Andy Oppel, SQL - The Complete Reference, 3rd Edition, McGraw Hill, 2017.

V Semester

Department of Computer Science & Engineering

Course Code : CST352 **Course : Design and Analysis of Algorithms**

L: 3 Hrs, T: 1 Hr, P: 0 Hr, Per Week **Total Credits : 04**

Course Objectives

1. Students should learn techniques for effective problem solving in computing.
2. Students should analyze different paradigms of problem solving to solve a given problem in efficient way.

SYLLABUS

UNIT I

Mathematical foundations for arithmetic and geometric series, Recurrence relations and their solutions, Principles of designing algorithms and complexity calculation, Asymptotic notations for analysis of algorithms, worst case and average case analysis, amortized analysis and it's applications.

UNIT II

Divide and Conquer- basic strategy, Binary Search, Quick sort, Merge sort, Strassen's matrix multiplication, Maximum sub-array problem, Closest pair of points problem, Convex hull problem.

UNIT III

Greedy method – basic strategy, fractional knapsack problem, Minimum cost spanning trees, Huffman Coding , activity selection problem ,Find maximum sum possible equal to sum of three stacks, K Centers Problem.

UNIT IV

Dynamic Programming -basic strategy, Bellmen ford algorithm, all pairs shortest path, multistage graphs, optimal binary search trees, traveling salesman problem, String Editing, Longest Common Subsequence problem and its variations.

UNIT V

Basic Traversal and Search Techniques, breadth first search and depth first search, connected components. Backtracking basic strategy, 8-Queen's problem, graph coloring, Hamiltonian cycles, sum of subset problem, Introduction to Approximation algorithm.

UNIT VI

NP-hard and NP-complete problems, basic concepts, non-deterministic algorithms, NP-hard and NP complete, decision and optimization problems, polynomial reduction ,graph based problems on NP Principle , vertex cover problem, clique cover problem

Course Outcomes

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

1. Understand mathematical formulation, complexity analysis and methodologies to solve the recurrence relations for algorithms.
2. Design Greedy and Divide and Conquer algorithms and their usage in real life examples.
3. Design Dynamic programming and Backtracking Paradigms to solve the real life problems.
4. Understand NP class problems and formulate solutions using standard approaches.

Text Books

1. Thomas H. Cormen et.al; "Introduction to Algorithms"; 3 Edition; Prentice Hall, 2009.
2. Horowitz, Sahani and Rajasekaram; "Computer Algorithms", Silicon Press, 2008.
3. Brassard and Bratley; "Fundamentals of Algorithms", 1 Edition; Prentice Hall, 1995. 4. Richard Johnsonbaugh, "Algorithms", Pearson Publication, 2003.

Reference Books

1. Parag Himanshu Dave, Balchandra Dave, "Design and Analysis of Algorithms" Pearson Education, O'relly publication
2. Richard Johnsonbaugh, "Algorithms", Pearson Publication, 2003.

V Semester

Department of Computer Science & Engineering

Course Code : CSP352 **Course : Design & Analysis of Algorithms Lab**

L: 0 Hrs, T: 0 Hr, P: 2 Hr, Per Week **Total Credits : 01**

Course Objectives

1. Analyze the performance of algorithms.
2. Demonstrate a familiarity with major algorithms and data structures.
3. Apply important algorithmic design paradigms and methods of analysis.

SYLLABUS

Experiment based on syllabus of Design and Analysis Algorithms (CST352).

Course Outcomes

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

1. Analyze greedy paradigm and implement greedy algorithms.
2. Analyze divide-and-conquer paradigm and synthesize divide-and-conquer algorithms.
3. Implement algorithms using Dynamic Approach and analyze it to determine its computational complexity.
4. Apply backtracking paradigm to realize real world problems.

Text Books

1. Thomas H. Cormen et.al. “Introduction to Algorithms”, Prentice Hall of India.
2. Horowitz, Sahani, Rajsekharam, “Computer Algorithms”, Galgotia Publications Pvt. Ltd.

Reference Books

1. Brassard, Bratley, “Fundamentals of Algorithms”, Prentice Hall
2. Algorithms -- A Creative Approach, 3RD Edition, UdiManber, Addison-Wesley, Reading, MA.

V Semester

Department of Computer Science & Engineering

Course Code : CST353 **Course : Computer Networks**

L: 3 Hrs, T: 0 Hr, P: 0 Hr, Per Week **Total Credits : 03**

Course Objectives

1. To develop an understanding of modern network architectures from a design and performance perspective.
2. To introduce the student to the major concepts involved in network protocols.
3. To provide an opportunity to do network programming

SYLLABUS

UNIT I

Data communication Components: Representation of data and its flow Networks, Various Connection Topology, Protocols and Standards, OSI model, Transmission Media, LAN: Wired LAN, Wireless LANs, Techniques for Bandwidth utilization: Multiplexing - Frequency division, Time division and Wave division

UNIT II

Data Link Layer: Error Detection and Error Correction - Fundamentals, Block coding, Hamming Distance, CRC; Flow Control and Error control protocols - Stop and Wait, Go back – N ARQ, Selective Repeat ARQ.

UNIT III

Medium Access Sub Layer: Switching, Random Access, Multiple access protocols - Pure ALOHA, Slotted ALOHA, CSMA/CD, CDMA/CA, IEEE 802 standard protocols.

UNIT IV

Network Layer: Internet Protocol (IP) – Logical Addressing: IPV4, IPV6; Address mapping: ARP, RARP, BOOTP and DHCP–Delivery, Forwarding and Unicast Routing protocols.UNIT V

Transport Layer: Elements of Transport protocols: Addressing, Connection establishment, Connection release, Crash recovery, User Datagram Protocol (UDP), Transmission Control Protocol (TCP), TCP Congestion Control; Quality of Service, QoS improving techniques: Leaky Bucket and Token Bucket algorithm.

UNIT VI

Application Layer: Domain Name Space (DNS), DDNS, TELNET, EMAIL, File Transfer Protocol (FTP), WWW, HTTP, SNMP, Bluetooth, Firewalls

Course Outcomes

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

1. Understand basics of computer networks and reference models
2. Identify the Design issues of each layer of OSI model
3. Implement the protocols of OSI model

Text Books

1. Computer Networks: 5th ed by Andrew. S. Tanenbaum. PHI Publication.
2. Data Communications and Networks: 3rd ed by Behrouz A. Forouzan. Tata McGraw Hill Publication.

Reference Books

1. James F. Kurose and Keith W. Ross: Computer Networking: A Top-Down Approach Featuring the Internet, 3rd Edition.
2. William Stallings, “Data and Computer Communications”, PHI 6th Edition

V Semester
Department of Computer Science & Engineering

Course Code : CSP353 **Course : Computer Networks Lab**

L:0 Hrs, T: 0 Hr, P: 2 Hrs, Per Week **Total Credits : 01**

Course Objectives

1. To introduce use of different network simulation software.
2. To analyze performance of different protocols at various layers of a network architecture.
3. To demonstrate the implementation of various networking concepts.

SYLLABUS

Experiments based on CST353 Syllabus.

Course Outcomes

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

1. Simulate and then configure different types of networks.
2. Implement algorithms present in different layers of OSI model
3. Implement networking concepts like server, client and addressing mechanism.

V Semester
Department of Computer Science & Engineering

Course Code : CSP354 **Course : Mobile Application Programming Lab**

L: 0 Hrs, T: 0 Hr, P: 4 Hr, Per Week **Total Credits : 02**

Course Objectives

The objective of this course is to develop the ability of students to design android applications. Use various features of android like broadcast receivers, services, threads, content providers etc. Effectively use files and database to store the data. Use location based services to develop navigation based applications.

SYLLABUS

- UI Widgets and Layout Manager
- Activity, Intent & Fragment
- Android Menu
- Data Storage
- Android Service
- Android Notification, Dialog, SMS and Broadcast Receiver
- SQLite and Content Provider and Location Services
- Introduction to IOS, IOS app development basics, Introduction to XCode.

Course Outcomes

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

1. Design basic android applications using UI resources: Activity, Viewgroups and Intents and enhance user interactivity by using toast, notification, dialogs etc.
2. Effectively use Android's APIs for data storage, retrieval, preferences, files, databases, and content providers.
3. Understand and implement Android's communication APIs for SMS, utilize background services, location based services, broadcast receiver.
4. Develop small IOS applications.

Text Books

1. Beginning Android Programming with Android Studio, 4Ed by J. F. DiMarzio, Wrox publication.
2. Professional Android 4 Application Programming by Reto Meier, Wiley Publication

Reference Book

1. Android Programming for Beginners - Second Edition by John Horton, Packt Publishing Pvt. Ltd.

V Semester
Department of Computer Science & Engineering

Course Code : CST355-1 **Course : Computer Graphics (Elective-I)**

L:3 Hrs, T: 0 Hr, P: 0 Hrs, Per Week **Total Credits : 03**

Course Objectives

The computer graphics course prepares students for activities involving in design, development and testing of modeling, rendering, shading and animation. Students will use the standard OpenGL library in several programming projects illustrating the theory and practice of programming computer graphics applications.

SYLLABUS

UNIT I

Introduction to Graphics: Importance of Computer Graphics, Graphics Hardware, Application of Computer Graphics, Raster and Vector Graphics, Raster scan display system, Raster graphics Algorithm, Video Controller, Input Devices for Interactive Operation.

UNIT II

Windows Graphics Programming (WGP): Introduction to Windows and APIs, WGP Fundamentals, Graphics Device Interface (GDI), Graphics Programming Languages, GDI Coordinate System.

UNIT III

Polygon filling methods: Scan Conversion Algorithms: Simple Ordered edge list, Edge Fill, Fence Fill and Edge Flag Algorithm, Seed Fill Algorithms: Simple and Scan Line Seed Fill Algorithm.

UNIT IV

2D Clipping algorithms for regular and irregular windows: Sutherland Cohen Out code, Sutherland Cohen Subdivision, Mid-Point subdivision, Cyrus Beck Algorithm, Polygon Clipping Algorithms.

UNIT V

2D Transformations, Normalized Device Coordinates, Viewing Transformations, 3D System Basics and 3D Transformations, Parallel and Perspective Projections,

UNIT VI

Hidden line & hidden surface removal algorithms, Back face detection, Rendering, Shading, Ray tracing techniques, Illumination methods and Color Systems.

Course Outcomes

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

1. Understand the core concepts of computer graphics, Graphics devices & Graphics Programming.

2. Understand and implement various Scan conversion techniques.
3. Understand and apply the windowing, clipping and various transformations principles.
4. Understand and apply the concepts of color models, lighting and shading models, hidden surface elimination and various shading and rendering models to enhance the image.

Text Books

1. Rogers; Procedural Elements of Computer Graphics; 3rd Edition; McGraw Hill, 2001.
2. Newman and Sproull; Principles of Interactive Computer Graphics; McGraw Hill, 1989.
3. Hearn and Baker; Computer Graphics; 2nd Edition; PHI, India, 1994.
4. Ivan Harringhton; Computer Graphics - A Programming Approach; McGraw Hill Publications, 1987.
5. Computer Graphics Using OPENGL- 2nd edition , F.S. Hill Jr. Pearson Education , 2003

Reference Books

1. James D. Foley, Andries Van Dam, Feiner Steven K. and Hughes John F. – Computer Graphics: Principles & Practise, Adddison Wesley Publishing House

V Semester
Department of Computer Science & Engineering

Course Code : CST355-2 **Course : Embedded Systems (Elective-I)**

L: 3Hrs, T: 0 Hr, P: 0 Hr, Per Week **Total Credits : 03**

Course Objectives

1. To introduce the concepts and principles of ARM Architecture for embedded system.
2. To introduce the programming model of ARM processor Architecture.
3. To introduce and learn Embedded Operating Systems.

SYLLABUS

UNIT I

ARM Architecture ARM Design Philosophy, Registers, PSR, Pipeline, Interrupts and Vector Table, Architecture Revision, ARM Processor Families.

UNIT II

ARM Programming Model-I Instruction Set: Data Processing Instructions, Branch, Load, Store Instructions, PSR Instructions, Conditional Instructions.

UNIT III

ARM Programming Model-II Thumb Instruction Set: Register Usage, Other Branch Instructions, Data Processing Instructions, Single-Register and Multi Register Load- Store Instructions, Stack, Software Interrupt Instructions

UNIT IV

ARM Programming Simple C Programs using Function Calls, Pointers, Structures, Integer and Floating Point Arithmetic, Assembly Code using Instruction Scheduling, Register Allocation, Conditional Execution and Loops.

UNIT V

Firmware & Boot Loader, examples, Embedded Operating Systems, Fundamental Components, Examples Simple Little Operating system. SLOS Directory Layout, Initialization, Memory Model, Interrupts and Exceptions Handling, Scheduler, Context Switch, Device Driver Framework.UNIT VI

Memory Management Cache Architecture, Polices, Flushing and Caches, MMU, Page Tables, Translation, Access Permissions, Content Switch.

Course Outcomes

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

1. Understand the Architecture of ARM processor.
2. Understand the ARM processor programming.
3. Understand the Embedded Operating System fundamentals.
4. Understand the Memory Management of ARM processor.

Text Books

1. ARM Systems Developer's Guides- Designing & Optimizing System Software – Andrew N. Sloss, Dominic Symes, Chris Wright, 2008,Elsevier.

Reference Books

1. Embedded Microcomputer Systems, Real Time Interfacing – Jonathan W. Valvano – Brookes / Cole, 1999, Thomas Learning.

V Semester

Department of Computer Science & Engineering

Course Code : CST355-3 **Course : Information Theory and Coding (Elective-I)**

L: 3 Hrs, T: 0 Hr, P: 0 Hr, Per Week **Total Credits : 03**

Course Objectives

1. Introduce the principles and applications of information theory.
2. To study how information is measured in terms of probability and entropy, and the relationships among conditional and joint entropies.
3. To study coding schemes, including error correcting codes.
4. Explain how this quantitative measure of information may be used in order to build efficient Solutions to multitudinous engineering problems.

SYLLABUS

UNIT I Introduction

Information source, Symbols, and Entropy, Mutual Information, Information measures for continuous Random Variable

UNIT II Source Coding

The source coding theorem, Kraft inequality, Shannon-Fano codes, Huffman codes, Arithmetic Codes, Lempel-Ziv-Welch algorithm, universal source codes

UNIT III Channel Capacity

Channel capacity; Noisy channel coding theorem for discrete memory-less channels; Channel capacity with feedback; Continuous and Gaussian channels

UNIT IV Error Control Coding

Linear block codes and their properties, hard-decision decoding, convolution codes and the Viterbi decoding algorithm, iterative decoding; turbo codes and low density- parity-check codes

UNIT V Rate Distortion Theory

Rate distortion function, random source codes; joint source-channel coding and the separation theorem

UNIT VI Cyclic and BCH Code

Generator polynomial, Encoding and decoding cyclic codes, RS code, Berlekemp algorithm, Galois Fields, definition & construction of BCH code

Course Outcomes

After successful completion of this course, student will be able to

1. Apply information theory and linear algebra in source coding and channel coding
2. Understand various error control encoding and decoding techniques
3. Analyse the performance of error control codes
4. Apply Polynomial generator and Galois Field Theory in the field of Cryptography.

Text Books

1. Ranjan Bose, “Information Theory, coding and cryptography”,TMH,2011
2. Salvatore Gravano “Introduction to Error control codes”, Oxford, 2001.
3. Wade Trape, Lawrence C Washington “Introduction to Cryptography with Coding Theory”, Pearson, 2011.

Reference Books

1. Reza, "An Introduction to Information Theory", Dover 1994
2. T. M. Cover and J. A. Thomas," Elements of Information Theory", John Wiley & Sons, New York, 1991.
3. R. Hill, "A First Course in Coding Theory", Oxford University Press, 1986

L. Hanzo, T.H. Liew and B.L. Yeap, "Turbo coding, turbo equalization and space- time coding for transmission over fading channels", John Wiley and Sons, 2002

V Semester
Department of Computer Science & Engineering

Course Code : CST355-4 **Course : Design Patterns (Elective-I)**

L: 3 Hrs, T: 0 Hr, P: 0 Hr, Per Week **Total Credits : 03**

Course Objectives

1. To learn the fundamentals of software design by referring a catalog of design patterns:
2. Demonstrate how to use design patterns to address code design and user interface issues.
3. Identify the most suitable design pattern to address a given application design problem.
4. Apply design principles (e.g., open-closed, dependency inversion, etc.).
5. Critique code by identifying and refactoring anti-patterns.

SYLLABUS

UNIT I

Elements of Design Pattern, Describing Design Pattern, Design Pattern Classification, Role of design Patterns in software design, how design patterns solve design problems, selection and use of Design Pattern, Example implementation of design pattern using UML

UNIT II Creational Patterns

Creational Design pattern: Introduction, Role of Creational pattern, instantiation of objects using creational patterns, Types (Factory method, Abstract Factory, Builder, Prototype, Singleton), Structure and comparison of various types of creational patterns, Examples of creational patterns.

UNIT III Structural Design Patterns:

Structural Design Pattern: Introduction, Role of Structural pattern, creating flexible and efficient arrangement of objects and classes using structural patterns, Types (Adapter, Bridge, Composite, Decorator, Façade, Proxy), Structure and comparison of various types of structural patterns, Examples of structural patterns, Comparative study of Creational and Structural Design patterns

UNIT IV Behavioral Patterns-I

Behavioral Design pattern: Introduction, Role of Behavioral pattern, Types: Interpreter Design pattern, Language grammar handling using interpreter design pattern, Template Method, Implement run-time variable on template design pattern, Iterator design pattern, Handling aggregate objects using Iterator design pattern,Chain of Responsibility principle, Methodology of responsibility sharing using request passing approach, Example of functional responsibility of object.

UNIT V Behavioral Patterns-II

Mediator Design Pattern, Analysis of Mutual Behavior of classes, Observer Design Pattern, Effect of single object on set of objects, Reference control between objects, State Design Pattern, State-wise behavior of object, Strategy Design pattern, selecting an algorithm at runtime, Memento Design pattern and its implementation

UNIT VI Case Study: Designing a Document Editor:

Design Problems, Document Structure, Formatting, Embellishing the User Interface, Supporting Multiple Look-and-Feel Standards, Supporting Multiple Window Systems, User Operations, Spelling Checking and Hyphenation, Summary, Complexity computation of Various Design Patterns.

Course Outcomes

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

1. Understand Architecture, Need, Characteristics and Ability of Design Pattern in Software Design Process.
2. Identify role, functionality and various abilities of Creational and Structural Design patterns in software design.
3. Identify Role and functionality and various abilities of Behavioral Design pattern in software design.
4. Demonstrate the designing of software using Design patterns and comment on complexity of design process.

Text Books

1. Design Patterns by Erich Gamma, Pearson Education
2. Design Patterns Explained by Alan Shalloway and James Trott, Addison-Wesley; 2nd edition

Reference Books

1. Pattern's in JAVA Vol-I by Mark Grand , Wiley DreamTech.
2. Pattern's in JAVA Vol-II by Mark Grand , Wiley DreamTech.
3. JAVA Enterprise Design Patterns Vol-III by Mark Grand , Wiley DreamTech.
4. Head First Design Patterns by Eric Freeman, O'Reilly

V Semester

Department of Computer Science & Engineering

Course Code : HUT353 **Course : Indian Traditional Knowledge**

L: 2 Hrs, T: 0 Hr, P: 0 Hr, Per Week **Total Credits : 0**

Course Outcomes

On successful completion of the course, students will have increased ability to understand the importance and application of:

1. Indian Knowledge system and its scientific approach.
2. Traditional knowledge and protection of nature.
3. The legality and its importance for the protection of Indian traditional knowledge.
4. Indian philosophical tradition.
5. Indian artistic tradition

SYLLABUS

1. Basic Structure of Indian Traditional Knowledge: Vedas, Upavedas, Vedang, Upadang, scientific approach
2. Ecology and Indian Traditional Knowledge: Meaning, role, case studies
3. Intellectual Property Rights and Indian traditional Knowledge: Meaning, role in protection of Indian traditional knowledge, cases studies
4. Indian Philosophical traditions: Nyay, Sankaya, Yog, Mimansa, Jainism, Buddhism, Sikhism, and other approaches
5. Indian Artistic Traditions: Chitrakala, Murtikala, Vastukala, Sangeet, Sthpatya, Nritya evam Sahitya, case studies
6. Knowledge of traditional Indian Science and Technology

Reference Material

1. Amit Jha (2009), Traditional Knowledge System in India, Atlantic Publishers and Distributors.
2. RR Gaur, Rajeev Sangal, GP Bagaria, Human Values and Professional Ethics (Excel Books, New Delhi, 2010)
3. V. Sivaramakrishanan (ed.), Cultural Heritage of India – Course material, Bharatiya VidyaBhavan, Mumbai, 5th Edition, 2014
4. Swami Jitatmanand, Modern Physics and Vedant, Bharatiya Vidya Bhavan
5. Swami Jitatmanand, Holistic Science and Vedant, Bharatiya Vidya Bhavan
6. S.C. Chatterjee and D.M. Datta, An introduction to Indian Philosophy, University of Calcutta, 1984
7. Pramod Chandra, Indian Arts, Howard University Press, 1984
8. Krishna Chaitanya, Arts of India, Abhinav Publications, 1987

   https://www.researchgate.net/publication/299625768_Traditional_Knowledge_systems_in_India_for_biodiversity_conservation/link

V Semester

Department of Computer Science & Engineering

Course Code : CST 398-1 **Course : Python and Data Analysis**

L: 3 Hrs, T: 0 Hr, P: 0 Hr, Per Week **Total Credits : 03**

Course Objectives

This course is designed to teach students how to analyze different types of data using Python. Students will learn how to prepare data for analysis, perform simple statistical analysis, create meaningful data visualizations and predict future trends from data.

Course Outcomes

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

1. Understanding basics of python for performing data analysis.
2. Understanding the data, performing preprocessing, processing and data visualization to get insights from data.
3. Use different python packages for mathematical, scientific applications and for web data analysis.
4. Develop the model for data analysis and evaluate the model performance.

Syllabus

UNIT - I : Python Fundamentals for Data Analysis

Python data structures, Control statements, Functions, Object Oriented programming concepts using classes, objects and methods, Exception handling, Implementation of user-defined Modules and Packages, File handling in python.

UNIT - II : Introduction to Data Understanding and Preprocessing

Knowledge domains of Data Analysis, Understanding structured and unstructured data, Data Analysis process, Dataset generation, Importing Dataset: Importing and Exporting Data, Basic Insights from Datasets, Cleaning and Preparing the Data: Identify and Handle Missing Values.

UNIT - III : Data Processing and Visualization

Data Formatting, Exploratory Data Analysis, Filtering and hierarchical indexing using Pandas. Data Visualization: Basic Visualization Tools, Specialized Visualization Tools, Seaborn Creating and Plotting Maps.

UNIT - IV : Mathematical and Scientific applications for Data Analysis

Numpy and Spicy Package, Understanding and creating H-dimensional arrays, Basic indexing and slicing, Boolean indexing, Fancy indexing, Universal functions, Data processing using arrays, File input and output with arrays.

UNIT V : Analysing Web Data

Data wrangling, Web scrapping, Combing and merging data sets, Reshaping and pivoting, Data transformation, String Manipulation, case study for web scrapping.

UNIT VI : Model Development and Evaluation

Introduction to machine learning - Supervised and Unsupervised Learning, Model development using Linear Regression, Model Visualization, Prediction and Decision Making, Model Evaluation: Over-fitting, Under-fitting and Model Selection.

Text Books

1. David Ascher and Mark Lutz, Learning Python, Publisher O' Reilly Media.
2. Reema Thareja, "Python Programming using Problem Solving approach", Oxford University press.
3. Wes Mckinney "Python for Data Analysis", First edition, Publisher O' Reilly Media.

Reference Books

1. Allen Downey, Jeffery Elkne, Chris Meyres; Learining with Python, Dreamtech Press.
2. David Taieb, "Data Analysis with Python: A Modern Approach" 1st Edition, Packt Publishing.

VI Semester

Department of Computer Science & Engineering

Course Code : CST356 **Course : Artificial Intelligence**

L:3 Hrs, T: 0 Hr, P: 0 Hrs, Per Week **Total Credits : 03**

Course Objectives

1. To understand challenges involved in designing intelligent systems.
2. To represent given problem using state space representation and solve it by using different search techniques.
3. To understand knowledge representation methods using logic programming.
4. To understand uncertainty theory in designing AI systems.
5. To understand learning methods in solving AI problems.

SYLLABUS

UNIT I

Introduction: Basics of problem solving, problem representation; Search Techniques: Problem size, complexity; Uninformed search techniques: Depth, Breadth, Uniform Cost, Depth Limited, Iterative deepening DFS.

UNIT II

Informed search techniques: Heuristic Based Search, Greedy Based First Search, A* Search; Local Search algorithms: Hill-climbing, Simulated Annealing, Genetic Algorithms.

UNIT III

Constraint Satisfaction Problems, Adversarial Search: Two player Games, The min- max algorithm, Alpha-Beta pruning.

UNIT IV

Propositional Logic: Inference, Equivalence, Validity and satisfiability, Resolution, Forward and Backward Chaining, First Order Logic: Syntax and Semantics of FOL, Inference in FOL, Unification, Forward Chaining, Backward Chaining, and Resolution.

UNIT V

Uncertainty Knowledge and Reasoning: Probability and Baye's Theorem, Statistical reasoning: Bayesian networks, Naïve bayes algorithm, Fuzzy Logic, Introduction to expert systemUNIT VI

Learning: Types of Learning, k-nearest neighbor, Decision Tree Learning, Artificial Neural Network, Perceptron Learning algorithm

Course Outcomes

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

1. Represent given problem using state space representation and apply uninformed and informed search techniques on it.
2. Solve the fully informed two player games using different AI techniques.
3. Solve the AI problems by using logic programming
4. Apply uncertainty theory based on techniques like probability theory and fuzzy logic.
5. Apply learning methods in solving AI problems.

Text Book

1. Stuart Russel and Peter Norvig; Artificial Intelligence: A Modern Approach; Third Edition; Pearson Education, 2009.

Reference Book

1. E.Rich, K. Knight, S. B. Nair; Artificial Intelligence; 3rd Edition; Tata McGraw Hill, 2014.
2. Denis Rothman; Artificial Intelligence By Example: Develop machine intelligence from scratch using real artificial intelligence use cases; Kindle Edition, Packt Publishing Ltd, 2018

VI Semester
Department of Computer Science & Engineering

Course Code : CSP356 **Course : Artificial Intelligence Lab**

L:0 Hrs, T: 0 Hr, P: 2 Hrs, Per Week **Total Credits : 01**

Course Objectives

1. To understand the concept behind design of AI problems.
2. To demonstrate the use of logic programming for solving AI problems.
3. To learn the use of probability and learning models for solving AI problems.

SYLLABUS

Practicals based on CST356 syllabus.

Practical will be performed in Python with OpenAI/ core AI tool.

Course Outcomes

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

1. Implement different AI toy problems by using search techniques.
2. Design two player games using min-max algorithm with Alpha-Beta pruning.
3. Simulate AI problems using logic programming.
4. Implement probabilistic based methods to solve classification problems.
5. Implement different learning methods for solving AI problems.

VI Semester
Department of Computer Science & Engineering

Course Code : CST357 **Course : Software Engineering**

L:3 Hrs, T: 0 Hr, P: 0 Hr, Per Week **Total Credits : 03**

Course Objectives

1. To make students a successful professionals in the field with solid fundamental knowledge of software engineering.
2. To prepare students with strong communication and interpersonal skills, as well as professional and ethical principles when functioning as members and leaders of multi-disciplinary teams.
3. To teach students how to apply their foundations in software engineering to adapt to readily changing environments using appropriate theory, principles and processes.

SYLLABUS

UNIT I

Introduction to Software Engineering, Software engineering principles, Software Myths, Software Engineering- A Layered Technology, Software Process Framework, Requirements Engineering Tasks, Requirement Engineering Process, Eliciting Requirement: Case Study Software Requirements Specification.

UNIT II

Software Process Models, The Waterfall Model, Incremental Process Models, Evolutionary Process Models, Specialized Process Models, The Unified Process Model, COCOMO Model, Agile Process Models, Agile metrics, Extreme Programming (XP), Scrum, Kanban, Software Deployment, Case Study.

UNIT III

Basic concepts of testing, Testing Life Cycle, Structural Testing, Functional Technique, Static testing, Dynamic testing, Unit Testing, Integration Testing, Validation Testing, System Testing, Debugging. Software Testing fundamentals, Black Box Testing, White Box Testing, Web Testing, Test case design, building, execution, Automated Testing

UNIT IV

Software Project management- Plans, Methods and Methodology, The Business Case, Project Success and Failure, Project Evaluation, Cost-benefit evaluation technique, Project Planning-stepwise project Planning, Software Effort Estimation- Albrecht Function Point Analysis, COSMIC Function Point, Cost Estimation, Project Scheduling.

UNIT V

An overview, Software Quality, A Framework for Product Metrics, Metrics for Analysis & Design Models, Metrics for Source Code, Metrics for Testing & Maintenance. Metrics for process & project - Software measurement, metrics for software quality, metrics for small organization, Managing people in software environment.

UNIT VI

Risk management - Risk strategies, Software risks, Risk identification, Risk refinement, RMMM, Risk Response development & Risk Response Control, Risk Analysis: Agile risk management using Jira, Change Management- Software Configuration Management, SCM Repository, SCM Process, Estimation, Software reengineering, Reverse engineering: A practical approach, Project Technical Writing: User manuals, Software Installation guides etc.

Course Outcomes

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

1. Ability to understand software engineering practices and various models.
2. Ability to understand software development Life Cycle.
3. Ability to understand software testing principles and techniques.
4. Ability to understand various software project management tasks and methods to implement them.

Text books and Reference books

1. Roger Pressman; Software Engineering-A Practitioner's Approach; Sixth Edition, MaGraw Hill, 2010
2. Project Management by Clifford F. Gray, Erik W. Larson, McGraw Hill
3. Ian Somerville; Software Engineering; Seventh Edition; Pearson Education. 2008.
4. Ethics in Information Technology, George W. Reynalds, 4th Edition, Cengage Leraning Publication
5. David Gustafsan, Software Engineering;Schaum's Series,Tata McGraw Hill,2002
6. Sanjay Mohapatra; Software Project Management, First Edition, Cengage Learning, 2011.
7. Rajib Mall, Software Project Management, 5th Edition, McGrawHill

VI Semester

Department of Computer Science & Engineering

Course Code : CSP357 **Course : Software Engineering Lab**

L: 0 Hrs, T: 0 Hr, P: 2 Hr, Per Week **Total Credits : 01**

Course Objectives

1. To teach students UML modeling tool employed in the software development life cycle.
2. To make students familiar with the hundreds of hierarchical and interrelated engineering requirements necessary for large and/or complex systems.
3. To teach students software testing tools employed in the software testing.
4. To teach students prototyping tool employed in the software industry to develop software prototype.

SYLLABUS

Practical based on CST357 syllabus.

Course Outcomes

After successful completion of this course, the student should able to:

1. Design Use case and activity diagram for given problem definition.
2. Design Sequence, class and state diagram for given problem definition.
3. Design Component and deployment diagrams for given problem definition.
4. Test cases using white box testing method.
5. Test cases using black box testing method

VI Semester

Department of Computer Science & Engineering

Course Code : CST358 **Course : Compiler Design**

L:3 Hrs, T: 0 Hr, P: 0 Hrs, Per Week **Total Credits : 03**

Course Objectives

1. To understand the theory and practice of compiler implementation.
2. To explore the principles, algorithms, and data structures involved in the design and construction of compilers.
3. To understand various phases of compiler and their working.

SYLLABUS

UNIT I

Introduction to Compilers- Introduction to Compilers, Phases of compiler design, Relating Compilation Phases with Formal Systems

Lexical Analysis- Lexical analysis, tokens, pattern and lexemes, Design of Lexical analyzer, Regular Expression, transition diagram, recognition of tokens, Lexical Errors.

UNIT II

Syntax Analysis- Specification of syntax of programming languages using CFG, Top- down parser, design of LL(1) parser, bottom up parsing technique, Handle and Viable Prefix, LR parsing, Design of SLR, CLR, LALR parsers, Parser Conflicts, Handling Ambiguous Grammars, Applications of the LR Parser.

UNIT III

Syntax directed translation- Study of syntax directed definitions & syntax directed translation schemes, Type and Type Checking, A Simple Type Checking System, implementation of SDTS, intermediate notations- postfix, syntax tree, TAC, translation of Assignment Statement, expressions, controls structures, Array reference.

UNIT IV

Storage allocation & Error Handling- Run time storage administration stack allocation, Activation of Procedures, Storage Allocation Strategies, Garbage Collection, symbol table management, Error detection and recovery- lexical, syntactic and semantic.UNIT V

Code optimization- Machine-independent Optimisation- Local optimization techniques, loop optimization- control flow analysis, data flow analysis, Loop invariant computation, Induction variable removal, other loop optimization techniques, Elimination of Common sub expression, and Machine-dependent Optimization techniques.

UNIT VI

Code generation – Problems in code generation, Simple code generator, code generation using labelling algorithm, Register allocation by Graph Colouring, Code Generation by Dynamic Programming.

Course Outcomes

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

1. Exhibit role of various phases of compilation, with understanding of types of grammars and design complexity of compiler.
2. Design various types of parses and perform operations like string parsing and error handling.
3. Demonstrate syntax directed translation schemes, their implementation for different programming language constructs.
4. Implement different code optimization and code generation techniques using standard data structures.

Text Books

1. Aho, Sethi, and Ullman; Compilers Principles Techniques and Tools; Second Edition, Pearson education, 2008.
2. Alfred V. Aho and Jeffery D. Ullman; Principles of Compiler Design; Narosa Pub.House, 1977.
3. Manoj B. Chandak, Khushboo P Khurana; Compiler Design; Universities Press, 2018.

Reference Books

1. Vinu V. Das; Compiler Design using Flex and Yacc; PHI Publication, 2008.
2. V. Raghavan; Principles of Compiler Design, McGraw Hill Education (India), 2010.

VI Semester

Department of Computer Science & Engineering

Course Code : CSP358 **Course : Compiler Design Lab**

L: 0 Hr, T: 0 Hr, P: 4 Hrs, Per Week **Total Credits : 02**

Course Objectives

This laboratory course is intended to make the students experiment on the basic techniques of compiler construction and use of various tools for implementation. This will provide deeper insights into the aspects of programming languages and various phases of compiler.

SYLLABUS

Experiments based on syllabus of Complier Design (CST358).

Course Outcomes

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

1. Use Open Source tools to create a lexical analyzer and parser.
2. Implement different types of Parsing techniques.
3. Implement various syntax directed translation schemes to generate intermediate code.
4. Implement various code optimization techniques to improve performance of a program segment and code generation.

Text Books

1. Doug Brown, John Levine, Tony Mason, Lex and Yacc, O'Reilly Media, 2nd Edition, 2012.
2. Des Watson, A Practical Approach to Compiler Construction, Springer, 1st ed. edition, 2017.

VI Semester

Department of Computer Science & Engineering

Course Code : CST359-1 **Course : Advanced Algorithms (Elective-II)**

L: 3 Hrs, T: 0 Hr, P: 0 Hr, Per Week **Total Credits : 03**

Course Objectives

1. To demonstrate a familiarity with major algorithms and data structures.
2. To analyze different algorithms with their practical applications.
3. To create efficient algorithms to address engineering problems.

SYLLABUS

UNIT I

Hashing: Review of Hashing, Hash Function, Collision Resolution Techniques in Hashing, Separate Chaining, Open Addressing, Analysis of Open and Closed Hashing, Rehashing, Hash Tables with Worst-Case O(1) Access: Perfect Hashing, Cuckoo Hashing, Hopscotch Hashing, Extendible Hashing.

UNIT II

Red Black Trees: Height of a Red Black Tree, Red Black Trees Bottom-Up Insertion, Top-Down Red Black Trees, Top-Down Deletion in Red Black Trees, Analysis of Operations.

Splay Trees: Splaying, Search and Update Operations on Splay Trees, Amortized Analysis of Splaying.

UNIT III

B-Trees: Advantage of B- trees over BSTs, Height of B-Tree, Search and Update Operations on B-Trees, Analysis of Operations, Introduction to B + Trees

Garbage Collection: Review, Challenges, Recent Trends, Memory Management Interface, Mark-and-Sweep: Garbage Collection Algorithm, Garbage Collection in Java

UNIT IV

Text Processing: String Operations, Brute-Force Pattern Matching, Boyer-Moore Algorithm, Rabin-Karp Algorithm, String Matching with Finite Automata, the Knuth- Morris-Pratt Algorithm, Multiple Longest Common Subsequence Problem (MLCS).

UNIT V

Computational Geometry: One Dimensional Range Searching, Two Dimensional Range Searching, Constructing a Priority Search Tree, Searching a Priority Search Tree, Priority Range Trees, Quad-trees, k-D Trees, Applications.

UNIT VI

Randomized Algorithms: Need for Randomized Algorithms, Approaches, Approx Weighted Vertex Cover, Randomized Max 3-SAT, Randomized MST, Randomized Median Finding, Probabilistic Max Cut, Randomized Quicksort, Primality Testing, Approximation Algorithm, Sum of Subset Problem.

Course Outcomes

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

1. Understand implementation of symbol table using hashing techniques.
2. Develop and analyze algorithms for red-black trees, B-trees and Splay trees.
3. Develop algorithms for text processing applications.
4. Identity suitable data structures and develop algorithms for computational geometry problems.

Textbooks

1. Mark Allen Weiss, Data Structures and Algorithm Analysis in C++, Fourth Edition, Pearson Education, 2002.
2. Horowitz, Sahni and Rajasekaran, Computer Algorithms, Universities Press, 2000.
3. Cormen, Leiserson, Rivest and Stein, Introduction to Algorithm, Third edition, PHI, 2009.

References

1. Aho, Hopcroft and Ullman, Data Structures and Algorithms, Pearson Education, 2002.
2. M T Goodrich, Roberto Tamassia, Algorithm Design, John Wiley, 2002.
3. Tanenbaum, Langram and Augestien, Data Structures using C and C++, Prentice Hall of India, 2002.

VI Semester

Department of Computer Science & Engineering

Course Code : CST359-2 **Course : Distributed Systems (Elective-II)**

L: 3 Hrs, T: 0 Hr, P: 0 Hr, Per Week **Total Credits : 03**

Course Objectives

This course introduces students to principles, design and implementation of distributed system. The lecture focus primarily on the principles and design of distributed systems and cover communication, distributed storage, naming, synchronization, scheduling, fault tolerance and recovery.

SYLLABUS

UNIT I

Introduction to Distributed systems- Examples of distributed systems, challenges, issues in distributed operating systems, communication primitives, Theoretical Foundations - inherent limitations of a distributed system, Lamports logical clocks, vector clocks, casual ordering of messages, global state, cuts of a distributed computation, termination detection.

UNIT II

Distributed Mutual Exclusion – introduction, the classification of mutual exclusion and associated algorithms (token based and non-token based approach), a comparative performance analysis.

UNIT III

Distributed Deadlock Detection -Introduction , deadlock handling strategies in distributed systems, issues in deadlock detection and resolution ,control organizations for distributed deadlock detection, centralized and distributed deadlock detection algorithms, hierarchical deadlock detection algorithms. Agreement protocols – introduction, the system model, a classification of agreement problems, solutions to the Byzantine agreement problem

UNIT IV

Distributed File system: Introduction to DFS, design issues, File service architecture, Distributed shared memory: design issues, Architecture, algorithms for implementing DSM, memory coherence and protocolsUNIT V

Distributed Scheduling: Introduction, issues in load distributing, components of a load distributing algorithm, load distributing algorithms, performance comparison, selecting a suitable load sharing algorithm, requirements for load distributing, task migration and associated issues.

UNIT VI

Failure Recovery: Introduction, classification of failures, consistent set of check points, synchronous and asynchronous check pointing and recovery. Fault Tolerance: Introduction, Atomic Actions and committing, Commit protocols, Non Blocking Commit Protocols, Voting Protocols, Dynamic Voting Protocols.

Course Outcomes

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

1. Understand and apply knowledge of basic distributed system techniques and concepts.
2. Comprehend issues in mutual exclusion, deadlock detection, and agreement protocols in the context of distributed systems.
3. Realize design issues for distributed file system, distributed shared memory and distributed scheduling.
4. Recognize the importance of fault tolerance and failure recovery in a distributed environment.

Text Books

1. Advanced concepts in Operating Systems – Singhal and Shivratri; McGraw Hill
2. Coulouris, Dollimore, Kindleberg; Distributed Systems Concepts and Design, Fourth Edition, Pearson education, 2009.
3. Distributed Systems An Algorithmic Approach, Second Edition, Sukumar Ghosh, CRC Press.

Reference Books

1. Andrew S. Tanenbaum; Distributed Operating System; Pearson education; 2003.
2. Pradeep K. Sinha, "Distributed Operating System-Concepts and Design", PHI, 2003.

VI Semester

Department of Computer Science & Engineering

Course Code : CST359-3 **Course : Digital Signal Processing (Elective-II)**

L: 3 Hrs, T: 0 Hr, P: 0 Hr, Per Week **Total Credits : 03**

Course Objectives

1. To make students aware about the meaning and implications of the properties of systems and signals.
2. To make students familiar with the most important methods in DSP, including digital filter design, transform-domain processing and importance of Signal Processors.

SYLLABUS

UNIT I

Introduction to digital signal processing: Discrete time signals & systems: Discrete time signals and its classification, Discrete time systems, Classification of discrete time systems, Linear convolution, Cross Correlation.

UNIT II

The Z-transform: Z-transforms, Inverse Z-transform, properties of z-transform, Concepts of zeros and poles of a system, region of convergence (ROC) of z- transform.

UNIT III

Structures for realization of LTI discrete-time systems in z domain: IIR systems: Direct Form-I, Direct Form-II, Cascade form and parallel form. FIR systems: Direct form, cascade form and linear phase realization.

UNIT IV

Frequency domain representation of discrete time signals and systems: Fourier transform of discrete time signals, properties of discrete time Fourier transform, Frequency response analysis of discrete time systems.

UNIT V

Discrete Fourier Transform: Discrete Fourier transform definition, properties of DFT, circular convolution, Decimation in time FFT algorithm, decimation in frequency FFT algorithm, Inverse FFT.

UNIT VI

Filter design Techniques: Design of discrete time IIR filters from continuous time filters, Design of FIR filters by windowing method – Hamming and Kaiser.

Introduction to DSP processor and its applications, Recent Trends/Developments.

Course Outcomes

On Successful completion of course, students will be able to:

1. Understand and Analyze different frequency domain signals
2. Analyze and process signals in the discrete domain
3. Design filters to suit specific requirements for specific applications

Text Books

1. Discrete time signal processing: Alan V. Oppenheim, Ronald W. Schafer & Buch, Pearson Education.
2. Digital Signal Processing: Salivahanan, Tata McGraw Hill.

Reference Books

1. Digital Signal Processing Theory and application: Proakis and Manolakis, PHI Ltd.
2. Digital Signal Processing: Sanjit K. Mitra, Tata McGraw Hill.
3. Digital Signal Processing: Jonathan Stein, Wiley India Ltd.

VI Semester

Department of Computer Science & Engineering

Course Code : CST359-4 **Course : Data Warehousing and Mining (Elective-II)**

L: 3 Hrs, T: 0 Hr, P: 0 Hr, Per Week **Total Credits : 03**

Course Objectives

This course gives an introduction to:

1. Methods and theory for development of data warehouses and data analysis using data mining.
2. Data quality and methods and techniques for preprocessing of data.
3. Modeling and design of data warehouses.
4. Algorithms for classification, clustering and association rule analysis.

SYLLABUS

UNIT I

Introduction to Data Warehouse, Data Warehouse basic Concepts, Architecture of Data Warehouse, Overview of ETL and OLAP OLTP integration – comparison of OLAP with OLTP systems, ROLAP, MOLAP and HOLAP, Multidimensional modeling

UNIT II

Data Cube, Data Cube Computation methods, Advanced SQL support for OLAP, Data Preprocessing-Data Cleaning methods, Descriptive Data Summarization, Data Reduction, Data Discretization and Concept hierarchy generation

UNIT III

Space Management in Data warehouse - Schemas for storing data in warehouse using different storage structures, B-tree index, hash index, clusters, Bitmap index functional index, domain index, Data partitions.

UNIT IV

Introduction: - What is Data mining? Data Mining on what kind of data, Data mining Functionalities, Classification of Data Mining Systems, Major Issues on Data mining, KDD Process, Association Rule mining.

UNIT V

Classification and Prediction:- Classification by decision tree induction, Bayesian Classification, Rule-based Classification, Associative Classification.

UNIT VI

Clustering: Measuring Data Similarity and Dissimilarity Partition based Clustering, Hierarchical based clustering, Density based clustering.

Course Outcomes

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

1. Understand fundamental theories and concepts of data warehousing.
2. Apply multi-dimensional modeling techniques in designing data warehouses.
3. Understand the principles of data mining.
4. Analyze and apply different methods and techniques involved in data mining.

Text books

1. Jaiwei Han and Micheline Kamber; Data Mining Concepts and Techniques; 2 edition; Morgan Kaufmann Publishers, 2006.
2. Tang and MacLennan, Data Mining with SQL Server 2005, Wiley Publishing, 2005
3. Data Warehousing and Fundamentals by Paulraj Ponniah, A Wiley-Interscience Publication

VI Semester

Department of Computer Science & Engineering

Course Code : CSP360 **Course : Project-1**

L: 0 Hrs, T: 0 Hr, P: 6 Hr, Per Week **Total Credits : 03**

Course Outcomes

On completion of this course the student will be able to

1. Identify and analyze problem statements by investigating various domains in the society.
2. Perform requirement analysis and design methodology for the complex problems.
3. Apply advanced programming technique and modern tools for the design and development of solution.
4. Apply ethical principles and working in team for software development and understand its impact in societal and environmental context.

Scope

Students are expected to approach to solving a real-world problem in providing effective and efficient software solution through team effort.

VI Semester

Department of Computer Science & Engineering

Course Code : CSP361 **Course : Comprehensive Viva**

L: 0 Hrs, T: 0 Hr, P: 2 Hr, Per Week **Total Credits : 01**

Course Objectives

1. To assess the overall knowledge of the student in Computer Science and Engineering.
2. To assess preparedness of the student for placements and entrance examinations for higher learning, viz. GATE, GRE, CAT, etc.
3. To facilitate the students in selecting appropriate career track for themselves.

SYLLABUS

The Comprehensive Viva will cover the contents from the courses, both the theory and the lab practice which the student learnt during third thru sixth semester of the undergraduate programme.

Mode of Conduction

- The students will be assessed using process (to be decided by department) for the Internal Assessment Component.
- The comprehensive viva (End Semester Component) shall be conducted by a committee consisting of one external examiner and two internal examiners.

Course Outcomes

On completion of the course the student will be able to

1. Respond to the queries and issues covering computing domain studied during second and third year of study.
2. Exhibit oral presentation skills and inter-personal skills.

VI Semester

Department of Computer Science & Engineering

Course Code : CST 399-1 **Course : Recent Trends in Computing**

L: 3 Hrs, T: 0 Hr, P: 0 Hr, Per Week **Total Credits : 03**

Course Objectives

The objective of this course is to impart necessary and practical knowledge of recent Java based web development frameworks and develop skills required to design real-life web based projects by:

1. Designing Enterprise based applications by encapsulating an application's business logic.
2. Designing and developing multi-tier web applications
3. Designing applications using existing frameworks - Spring, Hibernate, Angular JS 2.0

SYLLABUS

UNIT I

JDBC Java Database Connectivity (JDBC): The Design of JDBC, The Structure Query Language, JDBC Configuration, Executing a basic SQL Statement, using PreparedStatement to prevent SQL Injection Attack.

UNIT II

Servlet Basics, the Servlet Life Cycle, The Servlet That Generate HTML. Handling the Client Request, Form Data: The Role Of Form Data, Reading Form Data From Servlets: Reading Three Parameters. Handling the Client Request, Generating the Server Response.

UNIT III

JSP Java Server Pages (JSP): Overview of JSP Technology- The need of JSP, Benefit of JSP, JSP vs Servlets, Basic Syntax. Invoking JAVA Code with JSP Scripting Elements, JSP directives, Integrated Servlets and JSP.

UNIT IV

Spring: Overview of Spring Framework, Spring Bean Life Cycle, Spring Bean Scope, Basic Bean Wiring, Hibernate : Overview, Hibernate architecture, Hibernate Environment, Hibernate Configuration, Hibernate Sessions, Collections Mappings, Association Mappings, Hibernate Query Language.

UNIT V

Setting up Angular Development Environment, Understanding Angular Framework, Angular Architecture, Angular modularity, Component Structure, Decorators, Selector, Template, Style and Component. Building the first App.

UNIT VI

Developing an application with data binding and directives, Attribute directives, Structural directives, Form inputs and Events, Event handling, Error handling.

Course Outcomes

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

1. Implement JAVA based database application.
2. Demonstrate server and client side programming using servlets & Java server pages.
3. Perform Dependency Injection using Spring, and create mappings in Hibernate using HQL.
4. Create dynamic web application using Angular JS, and understand how data binding and dependency injection play a role in reducing redundancy.

Text Books

1. M. Deitel, P. J. Deitel, S. E. Santry; Advanced Java 2 Platform HOW TO PROGRAM; H– Prentice Hall.
2. Cay Horstman, Gary Cornell; Core JAVA Volume-II Advanced Features; 8th Edition.
3. Craig Walls; Spring In Action; 2nd Edition
4. Marty Hall, Larry Brown; Core Servlets and Java Server Pages Volume-1: Core Technologies; 2nd Edition.
5. Brad Green, Shyam Seshadri; “Angular JS”; O'Reilly; 2013

Reference Books

1. Kathy Sierra; Head First Java; 2nd Edition; O'Reilly Media, 2005.
2. Jim Keogh; “J2EE The Complete Reference”; McGraw Hill; Fifth Edition.
3. Angular JS Documentation https://angular.io/

IV Semester (Minor Scheme)

Department of Computer Science & Engineering

Course Code : CSTM41 **Course : Data Structures and Algorithms**

L: 4 Hrs, T: 0 Hr, P: 0 Hr, Per Week **Total Credits : 04**

Course Objectives

1. To impart to students the basic concepts of data structures and algorithms.
2. To familiarize students on different searching and sorting techniques.
3. To prepare students to use linear (stacks, queues, linked lists) and non-linear (trees, graphs) data structures.
4. To enable students to devise algorithms for solving real-world problems..

SYLLABUS

UNIT I Data Structures and Algorithms Basics

Introduction: basic terminologies, elementary data organizations, data structure operations; abstract data types (ADT) and their characteristics.

Algorithms: definition, characteristics, analysis of an algorithm, asymptotic notations, time and space tradeoffs. Array ADT: definition, operations and representations – row-major and column-major.

UNIT II Stacks and Queues

Stack ADT: allowable operations, algorithms and their complexity analysis, applications of stacks – expression conversion and evaluation (algorithmic analysis), multiple stacks.

Queue ADT: allowable operations, algorithms and their complexity analysis for simple queue and circular queue, introduction to double-ended queues and priority queues.

UNIT III Linked Lists

Singly Linked Lists: representation in memory, algorithms of several operations: traversing, searching, insertion, deletion, reversal, ordering, etc.

Doubly and Circular Linked Lists: operations and algorithmic analysis. Linked representation of stacks and queues, header node linked lists.

UNIT IV Sorting and Searching

Sorting: different approaches to sorting, properties of different sorting algorithms (insertion, Shell, quick, merge, heap, counting), performance analysis and comparison.

Searching: necessity of a robust search mechanism, searching linear lists (linear search, binary search) and complexity analysis of search methods.UNIT V Trees

Trees: basic tree terminologies, binary tree and operations, binary search tree [BST] and operations with time analysis of algorithms, threaded binary trees.

Self-balancing Search Trees: tree rotations, AVL tree and operations, B+-tree: definitions, characteristics, and operations (introductory).

UNIT VI Graphs and Hashing

Graphs: basic terminologies, representation of graphs, traversals (DFS, BFS) with complexity analysis, path finding (Dijkstra's SSSP, Floyd's APSP), and spanning tree (Prim's method) algorithms.

Hashing: hash functions and hash tables, closed and open hashing, randomization methods (division method, mid-square method, folding), collision resolution techniques.

Course Outcomes:

On completion of the course the student will be able to

1. Recognize different ADTs and their operations and specify their complexities.
2. Design and realize linear data structures (stacks, queues, linked lists) and analyze their computation complexity.
3. Devise different sorting (comparison based, divide-and-conquer, distributive, and tree-based) and searching (linear, binary) methods and analyze their time and space requirements.
4. Design traversal and path finding algorithms for Trees and Graphs.

Text Books:

1. Ellis Horowitz, SartajSahni& Susan Anderson-Freed, Fundamentals of Data Structures in C, Second Edition, Universities Press, 2008.
2. Mark Allen Weiss; Data Structures and Algorithm Analysis in C; Second Edition; Pearson Education; 2002.
3. G.A.V. Pai; Data Structures and Algorithms: Concepts, Techniques and Application; First Edition; McGraw Hill; 2008.

Reference Books

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein; Introduction to Algorithms; Third Edition; PHI Learning; 2009.
2. Ellis Horowitz, Sartaj Sahni and Sanguthevar Rajasekaran; Fundamentals of Computer Algorithms; Second Edition; Universities Press; 2008.
3. A. K. Sharma; Data Structures using C, Second Edition, Pearson Education, 2013.

V Semester (Minor Scheme)

Department of Computer Science & Engineering

Course Code : CSTM51 **Course : Software Engineering and Project Management**

L:4 Hrs, T: 0 Hr, P: 0Hr Per Week **Total Credits : 04**

Course Objectives

1. To make students a successful professionals in the field with solid fundamental knowledge of software engineering.
2. To prepare students with strong communication and interpersonal skills when functioning as members and leaders of multi-disciplinary teams.
3. To teach students how to apply software development life cycle concepts in readily changing environments using the appropriate theory, principles and processes.

SYLLABUS

UNIT I

Introduction to Software Engineering, Software Characteristic, Software Myths, Software Engineering-A Layered Technology, Software Process Framework, Software engineering principles: Communication Practices, Planning Practices, Modeling Practices, Construction Practice & Deployment.

UNIT II

Software Process Models, The Waterfall Model, Linear sequential model, IncrementalProcess Models, Evolutionary Process Models, Agile Process Models- Scrum, Software requirements: Functional non-functional requirements, User requirement, System requirements, Software requirements Specification.

UNIT III

An overview, Requirements Analysis, Analysis Modeling Approaches, Data Modeling, Object-Oriented Analysis, Scenario-Based Modeling, Flow-Oriented Modeling, Class-based Modeling, Behavioral Model. Design Engineering Concepts, Design Model, Unified Modeling Language using StarUML.

UNIT IV

An overview, Unit Testing, Integration Testing, Validation Testing, System Testing, Debugging. Software Testing, Fundamentals, Black-Box Testing, White-Box Testing. Automated testing using selenium.

UNIT V

An overview, Software Quality, A Framework for Product Metrics, Software Project management- Plans, Methods and Methodology, Software Effort Estimation- Albrecht Function Point Analysis, Cost Estimation, Project Scheduling using PERT and Gantt charts.

UNIT VI

Risk management - Risk strategies, Software risks, Risk identification, Risk refinement, RMMM, Change management, configuration management, maintenance tools and techniques, Software Configuration Management SCM Repository, SCM Process.

Course Outcomes

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

1. Ability to understand software engineering practices and various models.
2. Ability to understand software development Life Cycle.
3. Ability to understand software testing principles and techniques.
4. Ability to understand various software project management tasks and methods to implement them.

Text books and Reference books

1. Roger Pressman; Software Engineering-A Practitioner's Approach; Sixth Edition, McGraw Hill, 2010
2. Ian Somerville; Software Engineering; Seventh Edition; Pearson Education. 2008.
3. Ethics in Information Technology, George W. Reynolds, 4th Edition, Cengage Learning Publication
4. David Gustafsan, Software Engineering; Schaum's Series,Tata McGraw Hill,2002
5. Sanjay Mohapatra; Software Project Management, First Edition, Cengage Learning, 2011.
6. Rajib Mall, Software Project Management, 5th Edition, McGrawHill

VI Semester (Minor Scheme)

Department of Computer Science & Engineering

Course Code : CSTM61 **Course : AI and Machine Learning**

L: 4 Hrs, T: 0 Hr, P: 0 Hr, Per Week **Total Credits : 04**

Course Objectives

1. To understand challenges involved in designing intelligent systems.
2. To represent given problem using state space representation and solve it by using different search techniques.
3. To understand basic concepts of machine learning.
4. To understand and apply artificial neural network to real world problems.
5. To understand and apply probabilistic machine learning to real time problems.

SYLLABUS

UNIT I

Introduction: Basics of problem solving, problem representation; Search Techniques: Problem size, complexity; Uninformed search techniques- Depth, Breadth, Uniform Cost, Depth Limited, Iterative deepening DFS.

UNIT II

Informed search techniques: Heuristic Based Search, Greedy Based First Search, A* Search; Local Search algorithms: Hill-climbing, Simulated Annealing, Genetic Algorithms.

UNIT III

The concept learning task, General-to-specific ordering of hypotheses. Version spaces. Inductive bias. Decision Tree Learning. Rule Learning: Propositional and First-Order, Over-fitting, Cross-Validation. Experimental Evaluation of Learning Algorithms.

UNIT IV

Instance-Based Learning: k-Nearest neighbor algorithm, Radial basis functions. Case- based learning, K-means and Hierarchical Clustering.UNIT V

Artificial Neural Networks: Linear threshold units, Perceptions, Multilayer networks and back-propagation, recurrent networks.

UNIT VI

Probabilistic Machine Learning: Maximum Likelihood Estimation, MAP, Bayes Classifiers Naive Bayes, Bayes optimal classifiers, Minimum description length principle.

Course Outcomes

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

1. Represent given problem using state space representation and apply uninformed and informed search techniques on it.
2. Understand the basic concept and need of machine learning.
3. Understand learning systems in Artificial Intelligence.
4. Apply uncertainty theory based on techniques like probability theory in AI and Machine Learning system.

Text Books

1. Stuart Russel and Peter Norvig; Artificial Intelligence: A Modern Approach; Third Edition; Pearson Education, 2009
2. Tom Mitchell; Machine Learning – an Artificial Intelligence Approach, Volume-II
3. Ethem Alpaydin; Introduction to machine Learning, Third Edition