RCOEM

Shri Ramdeobaba College of
Engineering and Management, Nagpur

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
2022-23

B. TECH. (COMPUTER SCIENCE & ENGINEERING)

About the Department:

The Department of Computer Science & Engineering was established in 2002, is well-equipped with state-of-the-art infrastructure. The state of art infrastructure includes latest configuration desktops organized in four different laboratories.

The department hosts computers, laptops and lab with internet facility. The 24X7 network managed with Cyberoam UTM firewall, and CISCO router offers intranet and internet connectivity. The computer laboratories 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 high-end computing through Open Source technologies and hosts NVIDIA DGX DL Workstation.

The Department has a distinction of consistently achieving above 95% results in the final year. Students are encouraged to appear in GATE, CAT, GRE and other competitive examinations which have resulted in increasing number of students clearing these exams.

Students teams of CSE have emerged winners at the Grand Finale of 2018, 2019, 2020 and 2022 editions of Smart India Hackthoan and have been excelling at the world renowned prestigious International Collegiate Programming Contest, ACM ICPC Asia West Regional Contents since 2015.

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.

Program Education Objectives:

1. To be able to comprehend, understand and analyze Computer Science Engineering problems related to real life which can be better resolved by artificial intelligence and machine learning.
2. To impart exhaustive knowledge of Computer Science Engineering, AI and Machine Learning to cater the industrial needs and excel in innovation and management fields by prediction analysis.
3. To promote collaborative learning and spirit of team work through multidisciplinary AI based projects and diverse professional ethics.
4. To inculcate a conviction to believe in self, impart professional and ethical attitude and nurture to be an effective team member, infuse leadership qualities, and build proficiency in soft skills and the abilities to relate engineering with the social, political and technical issues as per the current scenario.

Programme 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 modelling 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 B. Tech Computer Science & Engineering

Semester - I

Sr. No.	Course Code	Course Name	Hours/ week			Credits	Maximum marks			ESE Duration (Hrs)
			L	T	P		Continuous Evaluation	End Sem Exam	Total	
1.	CHT154	Chemistry	2	0	0	2	40	60	100	03
2.	CHP154	Chemistry Lab	0	0	2	1	25	25	50	-
3.	MAT153	Mathematics - I	3	0	0	3	40	60	100	03
4.	CST121	Digital Electronics	3	0	0	3	40	60	100	03
5.	CSP121	Digital Electronics Lab	0	0	2	1	25	25	50	
6.	CST151	Programming for Problem Solving	4	0	0	4	40	60	100	03
7.	CSP151	Programming for Problem Solving Lab	0	0	2	1	25	25	50	-
8.	CST122	Computer Workshop-1	1	0	0	1	20	30	50	1.5
9.	CSP122	Computer Workshop-1 Lab	0	0	2	1	25	25	50	-
10.	IDT151	Creativity, Innovation & Design Thinking	1	0	0	1	20	30	50	1.5
		TOTAL	**14**	**0**	**8**	**18**			**700**	

Semester - II

Sr. No.	Course code	Course Name	Hours/ week			Credits	Maximum marks			ESE Duration (Hrs)
			L	T	P		Continuous Evaluation	End Sem Exam	Total	
1.	PHT154	Introduction to Quantum Computing	3	1	0	4	40	60	100	03
2.	PHP154	Introduction to Quantum Computing Lab	0	0	2	1	25	25	50	-
3.	MAT154	Mathematics - II	3	1	0	4	40	60	100	03
4.	MAP151	Computational Mathematics Lab	0	0	2	1	25	25	50	-
5.	CST123	Object Oriented Programming	4	0	0	4	40	60	100	03
6.	CSP123	Object Oriented Programming Lab	0	0	2	1	25	25	50	-
7.	CST124	Computer Workshop-2	1	0	0	1	20	30	50	1.5
8.	CSP124	Computer Workshop-2 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	-
11.	HUT152	Constitution of India	2	0	0	0	-	-	-	-
12.	PEP151	Yoga/Sports	0	0	2	0	-	-	-	-
		TOTAL	**15**	**2**	**12**	**20**			**700**	

Semester - III

Sr. No.	Course code	Course Name	Hours/ week			Credits	Maximum marks			ESE Duration (Hrs)
			L	T	P		Continuous Evaluation	End Sem Exam	Total	
1.	MAT252	Linear Algebra and Statistics	3	0	0	3	40	60	100	03
2.	CST221	Computer Architecture & Organization	3	0	0	3	40	60	100	03
3.	CST222	Data Structures	3	1	0	4	40	60	100	03
4.	CSP222	Data Structures Lab	0	0	2	1	25	25	50	-
5.	CST223	Theory of Computation	3	1	0	4	40	60	100	03
6.	HUT253	Business Communication	3	0	0	3	40	60	100	03
7.	CSP224	Programming Lab-I	0	0	4	2	25	25	50	-
8.	CHT252	Environmental Sciences	2	0	0	0	-	-	-	-
9.	HUT256	Indian Traditional Knowledge	2	0	0	0	-	-	-	-
		TOTAL	**19**	**2**	**6**	**20**			**600**	

Semester - IV

Sr. No.	Course code	Course Name	Hours/ week			Credits	Maximum marks			ESE Duration (Hrs)
			L	T	P		Continuous Evaluation	End Sem Exam	Total	
1.	MAT281	Discrete Mathematics	3	0	0	3	40	60	100	03
2.	CST225	Software Engineering	3	0	0	3	40	60	100	03
3.	CSP225	Software Engineering Lab	0	0	2	1	25	25	50	-
4.	CST226	Operating Systems	3	1	0	4	40	60	100	03
5.	CSP226	Operating Systems Lab	0	0	2	1	25	25	50	-
6.	CST227	Design and Analysis of Algorithms	3	1	0	4	40	60	100	03
7.	CSP227	Design and Analysis of Algorithms Lab	0	0	2	1	25	25	50	-
8.	HUT257	Cyber Laws & Ethics in IT	2	0	0	2	40	60	100	03
9.	CSP228	Programming Lab-II [Cross Platform Application]	0	0	2	1	25	25	50	-
10.	CST299	Open Elective-I/MOOC	3	0	0	3	40	60	100	03
		TOTAL	**17**	**2**	**8**	**23**			**800**	

Semester - V

Sr. No.	Course code	Course Name	Hours/ week			Credits	Maximum marks			ESE Duration (Hrs)
			L	T	P		Continuous Evaluation	End Sem Exam	Total	
1.	CST321	Database Management Systems	3	1	0	4	40	60	100	03
2.	CSP321	Database Management Systems Lab	0	0	2	1	25	25	50	-
3.	CST322	Artificial Intelligence	3	0	0	3	40	60	100	03
4.	CSP322	Artificial Intelligence Lab	0	0	2	1	25	25	50	-
5.	CST323	Computer Networks	3	1	0	4	40	60	100	03
6.	CSP323	Computer Networks Lab	0	0	2	1	25	25	50	-
7.	CST324	Design Pattern	3	0	0	3	40	60	100	03
8.	CST325	Program Elective-I	3	0	0	3	40	60	100	03
9.	CST398	Open Elective-II/MOOC	3	0	0	3	40	60	100	03
10.	CSP326	Mini Project	0	0	4	2	25	25	50	-
		TOTAL	18	2	10	**25**			**800**	

Semester - VI

Sr. No.	Course code	Course Name	Hours/ week			Credits	Maximum marks			ESE Duration (Hrs)
			L	T	P		Continuous Evaluation	End Sem Exam	Total	
1.	CST327	Compiler Design	3	1	0	4	40	60	100	03
2.	CSP327	Compiler Design Lab	0	0	2	1	25	25	50	-
3.	CST328	Machine Learning	3	1	0	4	40	60	100	03
4.	CSP328	Machine Learning Lab	0	0	2	1	25	25	50	-
5.	CST329	Data Warehousing & Mining	3	0	0	3	40	60	100	03
6.	CSP329	Data Warehousing & Mining Lab	0	0	2	1	25	25	50	-
7.	CST330	Program Elective-II	3	0	0	3	40	60	100	03
8.	CST399	Open Elective-III/MOOC	3	0	0	3	40	60	100	03
9.	CSP331	Project-I	0	0	6	3	25	25	50	-
		TOTAL	**15**	**2**	**12**	**23**			**700**	

Semester - VII

Sr. No.	Course code	Course Name	Hours/ week			Credits	Maximum marks			ESE Duration (Hrs)
			L	T	P		Continuous Evaluation	End Sem Exam	Total	
1.	CST421	Cloud Computing	3	0	0	3	40	60	100	03
2.	CSP421	Cloud Computing Lab	0	0	2	1	25	25	50	-
3.	CST422	Program Elective-III	3	0	0	3	40	60	100	03
4.	CSP422	Program Elective-III Lab	0	0	2	1	25	25	50	-
5.	CST423	Program Elective-IV	3	0	0	3	40	60	100	03
6.	CSP423	Program Elective-IV Lab	0	0	2	1	25	25	50	-
7.	CST498	Open Elective-IV/MOOC	3	0	0	3	40	60	100	03
8.	CSP424	Project-II	0	0	8	4	50	50	100	-
9.	CSP425	Industry Internship Evaluation	0	0	2	0	-	-	-	-
		TOTAL	**12**	**0**	**16**	**19**			**650**	

Semester - VIII

Sr. No.	Course code	Course Name	Hours/ week			Credits	Maximum marks			ESE Duration (Hrs)
			L	T	P		Continuous Evaluation	End Sem Exam	Total	
1.	CST426	Program Elective-V	3	0	0	3	40	60	100	03
2.	CST427	Program Elective-VI	3	0	0	3	40	60	100	03
3.	CSP428	Project-III	-	-	-	6	50	50	100	-
		TOTAL	**6**	**0**	**0**	**12**			300	
OR										
							Continuous Evaluation	**Industry Evaluation**	**Total**	
1.	CSP429	Full Semester Industry Internship	-	-	-	12	100	100	200	-
		TOTAL							**200**	

Program Electives [PE]:

Semester	Course Code	Course Name
V [PE-I]	CST325-1	Information Security and Privacy
	CST325-2	IOT Systems
	CST325-3	Computer Graphics
	CST325-4	Advanced Algorithms
VI [PE-II]	CSP330-1	Blockchain Technology
	CSP330-2	Vulnerability Assessment & Penetration Testing
	CSP330-3	System Design
	CSP330-4	Advanced OOPS
VII [PE-III]	CST422-1	Data Analytics and Visualization
	CST422-2	Customer Relationship Management
	CST422-3	Web Intelligence and Big Data
	CST422-4	Business Intelligence
VII [PE-IV]	CST423-1	Natural Language Processing
	CST423-2	Deep Learning
	CST423-3	Digital Image Processing
	CST423-4	Parallel Computing
VIII [PE-V]	CST426-1	Bioinformatics
		Quantum Algorithm
	CST426-2	Bio Inspired Intelligence Techniques
VIII [PE-VI]	CST427-1	Intrusion Detection and Prevention System
	CST427-2	Information Retrieval
	CST427-3	Augmented and Virtual Reality

Syllabus for Semester I, B. Tech. Computer Science & Engineering

Course Code: CHT154 **Course: Chemistry**

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

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 and molecular level.
- Discuss unique properties of nano-materials to solve challenges in our life and applications in computational world.
- Discuss how spectroscopic methods are used for qualitative and quantitative analysis.
- Analyze the utilization of green computing technology for environmental issues

Syllabus:

Module 1: Atomic and Molecular Structure [6 hours]

Equations for atomic and molecular orbitals. Energy level diagrams of diatomic. HOMO, LUMO, Crystal field theory and the energy level diagrams for transition metal ions and their optical and magnetic properties. Band Theory

Module 2: Nanomaterials for Advanced Computing: [6 hours]

Basics of Nanochemistry, classification, synthesis and Industrial applications, quantum dots for quantum computers, Doping of nanocrystals, Nanowires, Nanofibers, Nanotechnological advancements for computing.

Module 3: Characterization Techniques and computational tools: [6 hrs]

Fundamentals of spectroscopy, Electronic Spectroscopy, Nuclear Magnetic Resonance Spectroscopy. Basics of Nuclear magnetic resonance quantum computer Synthesis of drugs, basic soft-wares for bio-chemical assessment of drugs.

Module 4: Green Computing and Chemistry [6 hrs]

Metal extraction from E-wastes: Constraints and opportunities, Chemical exposure (Lead, Mercury, Cadmium, Chromium etc.) and contamination, Principles of Green Chemistry and Green Computing, Role of Green Computing in Environment and Research, Green devices and Green data Server.

Suggested Text Books:

1. Shikha Agrawal , Engineering Chemistry : Fundamentals and Applications, Cambridge University Press.
2. Dr. Rajshree Khare, A Textbook of Engineering Chemistry(AICTE), S.K. Kataria & Sons.
3. S. S. Dara, A Textbook of Engineering Chemistry, S. Chand Publications.
4. A. K. Das and M. Das, An introduction to nanomaterials and nanoscience, CBS Publishers and Distributors
5. M Afshar Alam, Sapna Jain, Hena Parveen, Green Computing Approach Towards Sustainable Development, Wiley Interscience Publications

Suggested Reference Books:

1. C. J. Cramer, Essentials of Computational Chemistry: Theories and Models, Second Edition, Wiley Interscience Publications.
2. Hans-Eckhardt Schaefer, Nanoscience: The Science of the Small in Physics, Engineering, Chemistry, Biology and Medicine, Springer-Verlag Berlin Heidelberg.

Syllabus for Semester I, B. Tech. Computer Science & Engineering

Course Code: CHP154 **Course: Chemistry Lab**

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

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:

1. Apply the fundamental principles of measurement and skills in preparation and handling of hazardous chemicals and interpret the statistical data related to measurements.
2. Estimate the rate constants of reactions and order of the reaction and/or to validate adsorption isotherms.
3. Use of various computational tools for analysis of different spectral properties and bio-activities.

List of Experiments: [Any six from the list]

1. Preparation of different Solutions: Molar solution, Normal solution and percent solution and Determination of concentration.
2. Demonstration of Handling of hazardous chemicals, MSDS (material safety data sheet), waste minimization strategies and chemical waste disposal.
3. Basic statistical analysis of results of neutralization of acid against the base and preparing acceptable graphs using software.
4. Prediction of infrared/NMR spectral and analytical data of organic molecules using Computational Software.
5. Spectroscopic/Colorimetric determine of wavelength of maximum absorption of chemical/biological compound in solution and determination of concentration using Lambert-Beer's Law.
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. Molecular docking of drugs using open computational software.
8. Determination of rate of the reaction at room temperature and analysis of experimental data using Computational Software
9. Use of open access software for the interpretation of various parameters of materials including drugs

Suggested 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.

Syllabus for Semester I, B. Tech. Computer Science & Engineering

Course Code: MAT153 **Course: Mathematics-I**

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

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; Eigenvalues 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. Codington, 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: 2nd ed: 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 for Semester I, B. Tech. Computer Science & Engineering

Course Code: CST121 **Course: Digital Electronics**

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

Course Outcomes

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

1. Understanding of various optimization techniques used to minimize and design digital circuits.
2. Analyze and design various combinational logic circuits.
3. Analyze and design various sequential circuits.
4. Design different microprocessor based components of computer system using combinational and sequential circuits.

Course Contents

UNIT-I

Basics of Digital Electronics

Motivation for digital systems: Logic and Boolean algebra, Number Systems. Logic Gates & Truth Tables, Demorgan's law, Minimization of combinational circuits using Karnaugh maps up to five variable. Map manipulation-essential prime implicants, non-essential prime implicants.

UNIT-II

Combinational Circuit Design

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

UNIT-III

Sequential circuit Design-I

Storage elements, Flip-flops and latches: D, T, J/K, S/R flip-flops. Master Slave Conversion of one of type of F/F to another Sequential circuit. Analysis –Input equations, state table, and analysis with J-K Flip flops. Sequential circuit Design, Design procedure, designing with D & J-K Flip flop.

UNIT-IV

Sequential circuit Design-II

Counters, asynchronous and synchronous design using state and excitation tables. Registers & Shift registers.

UNIT-V

Programmable logic Design

Memory & Programmable logic Devices: RAM, Array of RAM IC's, Read only Memory, PLA, PAL, Flash Memories

UNIT-VI

Fundamental of Microprocessor

Introduction to μp 8085, Addressing modes, Instruction set, Programming of μp 8085.

Text Books

1. Morris Mano; Digital Logic Design; Fourth edition, McGraw Hill
2. R.P.Jain; Modern Digital Electronic; Fourth edition; Tata McGraw-Hill.
3. V.J.Vibhute; 8-Bit Microprocessor & Microcontrollers; fifth edition.

Reference books

1. A. Anand Kumar; Fundamental of Digital Electronics; Second Edition, PHI
2. A.P.Godse; Digital circuit & design; Technical Publications; 2009.
3. Ramesh Gaonkar; 8 bit Microprocessor; CBS Publishers; 2011.

Syllabus for Semester I, B. Tech. Computer Science & Engineering

Course Code: CSP121 **Course: Digital Electronics Lab**

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

Course Outcome

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

1. Use logic gates for designing digital circuits
2. Implement combinational circuits using VHDL
3. Implement sequential circuits using VHDL
4. Apply the knowledge gained for their project work based on the hardware digital circuits

Practicals based on above theory syllabus

Syllabus for Semester I, B. Tech. Computer Science & Engineering

Course Code: CST151 **Course: Programming for Problem Solving**

L: 4 Hrs, T: 0 Hr, P: 0Hr, 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, pointers, structures and I/O operations for the formulation of algorithms and programs.
4. To apply programming to solve matrix addition, multiplication problems and searching & sorting problems.

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 /Pseudo code with examples. Arithmetic expressions and precedence

UNIT-II: C Programming Language

Introduction to C language: Keywords, Constant, Variable, Data types, Operators, Types of Statements,

Pre-processor 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.
2. Let Us C: Yashwant Kanetkar, BPB Publication

Syllabus for Semester I, B. Tech. Computer Science & Engineering

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

L: 0 Hrs, T: 0 Hr, P: 2Hr, 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.

Syllabus for Semester I, B. Tech. Computer Science & Engineering

Course Code : CST122 **Course : Computer Workshop-1**

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

Course Objectives

1. Understand the definition and principles of UI/UX Design in order to design with intention.
2. Achieve a deep understanding of the entire life-cycle of design—the process, purpose, and tools.
3. Learn the basics of HCI (human-computer interaction) and the psychology behind user decision-making.
4. Discover the industry-standard tools and specific project deliverables in UI/UX.
5. Explain why you made design decisions, through presentations of assignments and your personal portfolio.

Unit 1:

UI/UX Overview: Intro to UI/UX, Notion & Figma Setup, Design Thinking.
User Research: How to identify stakeholders, Figma Basics, How to identify user needs.

Unit 2:

User Journeys: Mapping the user journey, Figma Grayscales, Finding solutions & constraint cards, Grayscales & User Testing: UX Principles, Figma Prototype, Understanding user testing.

Unit 3:

UI Principles: UI Principles, Color and Font. Style Guide: Components, Responsive Design.

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

1. Understand basics of UI/UX
2. Find solutions and constraint cards.
3. Design responsive UI.

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

Syllabus for Semester I, B. Tech. Computer Science & Engineering

Course Code: CSP122 **Course : Computer Workshop-1 Lab**

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

Course Objectives

Throughout the course, students will be expected to learn following concept:

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 successful completion of the course, students 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, and animation.

Syllabus for Semester I, B. Tech. Computer Science & Engineering

Course Code: IDT151 **Course: Creativity, Innovation & Design Thinking**

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

Course Outcomes

1. Be familiar with processes and methods of creative problem solving
2. Enhance their creative and innovative thinking skills
3. 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/Brain writing, 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 group's hands-on projects
- Eight-dimensional (8D) ideation method examples Large teams videos

Syllabus for Semester II, B. Tech. Computer Science & Engineering

Course Code: PHT154 **Course: Introduction to Quantum Computing**

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

Course Objectives

1. To introduce the fundamentals of quantum computing to students
2. The problem solving approach using finite dimensional mathematics

Course Outcomes: After successful completion of the course, the students will learn,

1. Basics of complex vector spaces
2. Quantum mechanics as applied in Quantum computing
3. Architecture and algorithms
4. Fundamentals of Quantum computations

Module 1: Complex Vector Spaces

Algebra and Geometry of Complex numbers, Real and Complex Vector Spaces, definitions, properties, basis and dimensions, Generalization to n-dimensional space

Module 2: Linear Algebra

Inner products, Hilbert Spaces, Eigenvalues and Eigenvectors, Hermitian and Unitary Matrices, Tensor Product, Applications of linear algebra in computer graphics, Geometric transforms, positioning the virtual camera

Module 3: Basic Quantum Theory

Introduction to Quantum mechanics, Schrodinger's time dependent equation, Wave nature of Particles, expectation values, variance, standard deviation, probability density, Stationary states, Infinite square well, Uncertainty principle

Module 4: Classical and Quantum Systems

Deterministic and Probabilistic Systems, Quantum Systems, Observations, Quantum measurement principles, Stochastic matrices, Probabilistic double slit experiment with photons, Entangled states, Quantum clocks

Module 5: Architecture

Bits and Qubits, Classical Gates, Reversible Gates, Quantum Gates, Toffoli and Fradkin Gates, Bloch Sphere, Deusch Gate, No-cloning theorem, Applications in Cryptography and Quantum teleportation

Module 6: Quantum algorithms

Deutsch's algorithm, The Deutsch-Jozsa algorithm, Simon's periodicity algorithm, Grover's search

algorithm, Shor's factoring algorithm, Quantum Fourier Transform

Text Book

1. Quantum computing for computer scientists, Noson S. Yanofsky, Mirco A. Mannucci, Cambridge University Press 2008
2. Introduction to Quantum Mechanics, 2nd Edition, David J. Griffiths, Prentice Hall New Jersey 1995

Reference Books

1. Quantum computing explained, David McMahon, Wiley-interscience, John Wiley & Sons, Inc. Publication 2008
2. Quantum computation and quantum information, Michael A. Nielsen and Isaac L. Chuang, Cambridge University Press 2010

Syllabus for Semester II, B. Tech. Computer Science & Engineering

Course Code: PHP154 **Course: Introduction to Quantum Computing lab**

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

Course Outcomes

The physics laboratory will consist of experiments and programming exercises illustrating the principles of physics relevant to the study of computer science and engineering. During the training in the Physics Lab, the students will be able,

1. To develop skills for experimental verification of physics laws
2. To analyze the results using the mathematical tools
3. To learn the computational techniques
4. To write the project reports

The laboratory will consist of general physics experiments and computational physics practicals.

General Physics:

1. Measuring scales and error estimation
2. Verification of Ohm's law and linear least square fitting method
3. Verification of Newton's law of cooling
4. Simple harmonic motion
5. Magnetic flux measurement using the graphical method of integration
6. Measurement, analysis and fitting of non-linear IV characteristics of PN junction diode

Computational Physics

1. Linear least square fit method for data analysis
2. Plotting of Plank's function and verification of Stefan's law
3. Finding inverse, norm and inner products, rank of a matrix
4. Introduction to quantum computing packages (GitHub repository)
5. Implementation of Deutsch-Josza algorithm using Cirq library

Project

Project on the applications of linear algebra, quantum mechanics or quantum computing to solve science and engineering problems.

Reference Books

1. Lab manual prepared by Physics Department, RCOEM, Nagpur

Syllabus for Semester II, B. Tech. Computer Science & Engineering

Course Code: MAT154 **Course: Mathematics-II**

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

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 - I : Differential Calculus: (12hours)

Taylor's and Maclaurin's series expansions; radius of curvature (Cartesian form), evolutes and involutes, Limit and continuity of functions of several variables and their partial derivatives, Eulers Theorem, chain rule, total derivative, Jacobians, Maxima, minima and saddle points; Method of Lagrange multipliers.

Module - II : Integral Calculus: (6 hours)

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

Module - IV : 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 - V : Multiple Integrals (10 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: area, mass and volume by double integration, Center of mass and Gravity (basic concepts).

Module - VI : Vector Calculus (10 hours)

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

Topics for self learning

Rolle's theorem, Mean value theorems, Indeterminate forms , 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. B.S. Grewal, Higher Engineering Mathematics, Khanna Publishers, 35th Edition, 2000.
4. Ramana B.V., Higher Engineering Mathematics, Tata McGraw Hill New Delhi, 11th Reprint, 2010.
5. P. N. Wartikar and J. N. Wartikar, A text book of Applied Mathematics Volume I & II, Pune Vidhyarthi Griha Prakashan, Pune-411030 (India).

Syllabus for Semester I, B. Tech. Computer Science & Engineering

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

L: 0 Hrs, T: 0 Hr, P: 2Hr, 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.
2. A minimum of 8 experiments to be performed based on the above list.

Syllabus for Semester II, B. Tech. Computer Science & Engineering

Course Code: CST123 **Course: Object Oriented Programming**

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

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.

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.

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, Linked List, Hashset, Treeset.

UNIT V

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

UNIT VI

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

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.

Syllabus for Semester II, B. Tech. Computer Science & Engineering

Course Code : CSP123 **Course : Object Oriented Programming Lab**

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

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.

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.
3. Implement programs based on streams and multithreading.

SYLLABUS

Experiments based on above Syllabus.

Syllabus for Semester II, B. TECH Computer Science & Engineering

Course Code : CST124 **Course : Computer Workshop-2**

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

Course Objective

The objective of this course is to familiarize the students with an important web framework for developing user interfaces. It aims for developing high end web applications by the use of ReactJS features.

Course Outcomes

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

1. Implement the fundamentals of React with Java Script and JSX
2. Understand Templating concept along with different types of components in ReactJS
3. Understand different state and Life Cycle Methods
4. Implement Router with react Router.

Course Contents

UNIT-I

Introduction to React

React JS Introduction, Advantages of React JS, Introduction to JSX, Difference between JS and JSX.

UNIT-II

Templating using JSX

Working with React, Create Element, Expressions, logical operators, specifying attributes, children and Fragments

UNIT-III

Basics in React

React Components overview, Types of components, Controlled, Split Up, Composable, Reusable, Component Declarations and Styling Components

UNIT-IV

Props and State

State and it significance, Read state and set state, Passing data to component using props, Validating props using prop Types, Supplying default values to props using default Props

UNIT-V

Lifecycle Methods- EVENT HANDLING

Lifecycle Methods, Fetching Data, ES6 Spread Operators, Conditional Rendering, Client- or Server-side Search, Error Handling.

UNIT-VI

Routing with react router

Introduction to React Router, Routing in single page applications, Browser Router and Hash Router components Configuring route with Route component.

Text Books

- Pure React- a step by step guide - Dave Ceddia
- Road to learn react - Robin Wieruch
- React in Action 1st Edition - Mark Tielens Thomas

Syllabus for Semester II, B. Tech. Computer Science & Engineering

Course Code: CSP124 **Course : Computer Workshop-2 Lab**

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

Course Objective

The objective of this course is to familiarize the students with an important web framework for developing user interfaces. It aims for developing high end web applications by the use of ReactJS features.

- Practical based on CST124

Syllabus for Semester II, B. Tech. Computer Science & Engineering

Course Code: HUT151 **Course: English**

L: 2 Hrs, T: 0 Hr, P: 0Hr, 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

3.6 Cliches

4. Nature and Style of sensible Writing

4.1 Describing

4.2 Defining

4.3 Classifying

4.4 Providing examples or evidence

5. Writing Practices

5.1 Comprehension

5.2 Precise Writing

5.3 Essay Writing

5.4 Letter Writing

5.5 Email Writing

6. 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 Semester II, B. Tech. Computer Science & Engineering

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

L: 0 Hrs, T: 0 Hr, P: 2Hr, 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

List of Practical (2 hours each for each batch) based on unit 6 (oral communication).

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

Syllabus for Semester I, B. Tech. Computer Science & Engineering

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

L: 2 Hrs, T: 0 Hr, P: 0Hr, 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

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

Syllabus for Semester II, B. Tech. Computer Science & Engineering

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

L: 0 Hrs, T: 0 Hr, P: 2Hr, 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.