JECRC™
UNIVERSITY
BUILD YOUR WORLD

Faculty of Engineering

Syllabi and Course Structure

B. Tech. (CSE) with Specialization Data Science
(In Association with Samatrix)

(2020-2024)
Academic Programmes

April 2020

The curriculum and syllabus for B.Tech. Program conforms to outcome based teaching learning process. In general, several outcomes have been identified and the curriculum and syllabus have been planned in such a way that each of the courses meets one or more of these outcomes. Student outcomes illustrate the students are expected to know and be able to do by the time of graduation. These relate to the skills, understanding, and behaviours that students acquire as they progress through the program. Further each course in the program brings out clear instructional objectives which are mapped to the student outcomes.

B.Tech. (CSE) Program Educational Objective (PEO's):
A graduate of the Computer Science and Engineering Program should:

PEO- I

Students will develop themselves as effective professionals by solving real problems through the use of computer science knowledge and with attention to team work, effective communication, critical thinking and problem solving skills.

PEO- II

Students will develop professional skills that prepare them for immediate employment and for life-long learning in advanced areas of computer science and related fields.

PEO- III

Students will demonstrate their ability to adapt to a rapidly changing environment by having learned and applied new skills and new technologies.

PEO- IV

Students will be provided with an educational foundation that prepares them for excellence, leadership roles along diverse career paths with encouragement to professional ethics and active participation needed for a successful career.

Program Outcome (PO's)

A graduate of the Computer Science and Engineering Program will demonstrate:

PO1. Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO2. Problem analysis: Identify, formulate, research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3. 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.

PO4. 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.

PO5. 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.

PO6. 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.

PO7. 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.

PO8. Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO9. Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO10. 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.

PO11. 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.

PO12. Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Program Specific Outcome:

PSO1: The ability to understand, analyze and develop computer programs in the areas related to algorithms, system software, multimedia, web design, big data analytics, and networking for efficient design of computer-based systems of varying complexity.(Professional Skills)

PSO2: The ability to apply standard practices and strategies in software project development using open-ended programming environments to deliver a quality product for business success. (Problem-Solving Skills)

PSO3: The ability to employ modern computer languages, environments, and platforms in creating innovative career paths to be an entrepreneur, and a zest for higher studies.(Successful Career and Entrepreneurship)

Faculty of Engineering
B. Tech. (common to all disciplines) I Year
Course Structure for 2020-2024 Batch

Semester I

Subject Code	Subject	Contact Hours L-T-P	Credits	
BMC120D	English	2-0-0	2	F
BMC121B	Communication Technique Lab	0-2-2	4	
BAS001C	Engineering Mathematics-I *	3-1-0	4	F
BAS010B	Applied Physics	3-0-0	3	F
BES001B	Basic Electronics Engineering	3-0-0	3	F
BES014A	Computer Programming in C *	3-0-0	3	F
Samatrix/spl	Introduction to Data Science and Foundation of Data Analytics	3-0-0	3	SC
Samatrix/spl	Introduction to Data Science and Foundation of Data Analytics lab	2-0-0	2	SC
BAS012A	Applied Physics Lab	0-0-2	2	F
BES002A	Engineering Graphics	0-0-2	2	F
BES012A	Computer Programming C Lab*	0-0-2	2	F
BES004A	Basic Electronics Engineering Lab	0-0-2	2	F
	Environmental science	NC		
	TOTAL	**19-3-10**	**32**	

* **In semester I common to all sections**

NC- Non Credit Course, It is mandatory to clear for completion of degree.

Semester II

Subject Code	Subject	Contact Hours L-T-P	Credits	
BAS002C	Engineering Mathematics-II **	3-1-0	4	F
BES005A	Basic Electrical Engineering	3-0-0	3	F
BES007A	Engineering Mechanics	2-1-0	3	F
BES003A	Engineering Workshop	0-0-2	2	F
Samatrix/spl	Data Analysis using Python, Numpy, Pandas, Matplotlib, and Seaborn	3-0-0	3	SC
Samatrix/spl	Data Analysis using Python, Numpy, Pandas, Matplotlib, and Seaborn lab	0-0-2	2	SC
BES008B	Basic Electrical Engineering Lab	0-0-2	2	F
BES017A	Foundation of Computing	3-0-0	3	F
	Introduction Web Technology Lab	0-0-3	3	F
BES010A	**Engineering Mechanics Lab**	0-0-2	2	F
	Essence of Indian Traditional Knowledge/Indian Constitution		NC	
	TOTAL	**14-2-11**	**27**	

**** In semester II common to all sections**

NC- Non Credit Course, It is mandatory to clear for completion of degree.

B.Tech CSE III Semester

Sr. No.	Course Code	Course Title	L	T	P	Contact Hrs.	Credits	Type
1	BAS 007A	Discrete Mathematics	2	0	0	2	2	F
2	BCO 001B	Software Engineering	3	0	0	3	3	S
3	BCO 002A/ BCO 002B	Data Structures and Algorithms	3	0	0	3	3	C
4	Samatrix/spl	R Programming for Data Science and Data Analysis	2	0	0	2	2	SC
5	Samatrix/spl	Probabilistic Modelling and Reasoning with Python and R	2	0	0	2	2	SC
6	Samatrix/spl	R Programming for Data Science and Data Analysis lab	0	0	2	2	2	SC
7	Samatrix/spl	Probabilistic Modelling and Reasoning with Python and R lab	0	0	2	2	2	SC
8	BEE009A	Digital Systems	4	0	0	4	4	F
9	BCO 005A	Data Structure and Algorithms Lab	0	0	2	2	2	C
10	BEE010A	Digital Systems Lab	0	0	2	2	2	F
11	BCO 080B	Linux Programming Lab	0	0	2	2	3	S
12	BCO 011A	Computer Networks	4	0	0	4	4	C
13	BCO 016A	Seminar	0	0	1	1	1	S
		Total	**20**	**0**	**11**	**31**	**31**	

B.Tech CSE Semester IV

Sr. No.	Course Code	Course Title	L	T	P	Contact Hrs.	Credits	TYPE
1	BCO 007A	Computer Graphics	3	0	0	3	3	C
2	BCO 008B	Operating Systems	3	0	0	3	3	C
3	BCO 009B	Computer Organization and Design	3	1	0	4	4	C
4	BCO 010C	Database Management Systems	4	0	0	4	4	C
5	Samatrix/spl	Machine Learning and Pattern Recognition	3	0	0	3	3	SC
6	BCO 012A	Software Project Management	3	0	0	3	3	S
7	Samatrix/spl	Machine Learning Practical with Python, Scikit-learn, Matplotlib, TensorFlow Lab	0	0	4	4	4	SC
8	BCO 013A	Database Management Systems Lab	0	0	2	2	2	C
9	BCO 014A	Operating Systems (Unix Programming) Lab	0	0	2	2	2	C
10	BCO 015A	Computer Graphics Lab	0	0	2	2	2	C
11	BCO 016A	Seminar	0	0	1	1	1	S
		Total	19	1	11	31	31	

B.Tech. CSE Semester V

Sr. No.	Course Code	Course Title	L	T	P	Contact Hrs.	Credits	Type
1	Samatrix/spl	Scala for Data Science	2	0	0	2	2	SC
2	BCO 017A	Formal Languages & Automation Theory	4	0	0	4	4	S
3	BCO 023A	Design & Analysis of Algorithms	3	0	0	4	4	S
4	BCO 019A	Artificial Intelligence	4	0	0	4	4	S
5	BCO 035A	Programming in Java	4	0	0	4	4	S
6	Samatrix/spl	Scala for Data Science Lab	0	0	2	2	2	SC
7	Samatrix/spl	Big Data Analytics with Spark	2	0	0	2	2	SC
8	Samatrix/spl	Big Data Analytics with Spark lab	0	0	2	2	2	SC
9	BCO 068A	Programming in Java Lab	0	0	2	2	2	S
10	BCO 025A	Design & Analysis of Algorithms Lab	0	0	2	2	2	S
11	BCO 022A	Seminar	0	0	1	1	1	S
		Total	**19**	**0**	**9**	**28**	**28**	

B.Tech CSE Semester VI

Sr. No.	Course Code	Course Title	L	T	P	Contact Hrs.	Credits	TYPE
1	Samatrix/spl	Data Science with Hadoop	2	0	0	2	2	SC
2	BCO 028A	Compiler Construction	3	0	0	3	3	SC
3	BCO 024A	Advanced Computer Architecture	3	0	0	3	3	C
4	BCO 037A	Advance Programming in Java	4	0	0	4	4	S
5	Samatrix/spl	Data Science with Hadoop lab	0	0	2	4	4	SC
6	Samatrix/spl	DevOps for Web Development	0	0	2	2	2	SC
7	Samatrix/spl	DevOps for Web Development Lab	0	0	2	2	2	SC
8	BCO 031A	Compiler Design Lab	0	0	2	2	2	S
9	BCO 069A	Advance Programming in Java Lab	0	0	2	2	2	ID
10	BCO 026A	Seminar	0	0	1	1	1	C
11	BCO 074A	Minor Project	0	0	1	1	1	S
		Total	**12**	**0**	**12**	**24**	**24**	

B.Tech. CSE Semester VII

Sr. No.	Course Code	Course Title	L	T	P	Contact Hrs.	Credits	Type
1	Samatrix/spl	**Data Visualization**	**3**	0	0	3	3	S
2	BCO 030A	Principles of Information System Security	4	0	0	4	4	S
3	BCO 029A	Data Mining & Warehousing	4	0	0	4	4	S
4	Samatrix/spl	Data Visualization Lab	0	0	2	2	2	S
5	BCO 032A	Project	0	0	2	2	2	C
6	BCO 033A	Seminar	0	0	1	1	1	S
		Total	**11**	**0**	**5**	**16**	**16**	

B.Tech. CSE Semester VIII

S. No.	Code	Subject	L	T	P	Contact Hrs.	Credits	Type
1	BCO 034A	Industrial Project/Dissertation	0	0	28	28	28	C
		TOTAL	**0**	**0**	**28**	**28**	**28**	

List of Program Electives-I

Sr. No.	Course Code	Course Title	L	T	P	Contact Hrs.	Credits	Type
1	Samatrix/spl	Principle and Design of IoT Systems	4	0	0	4	4	SC
2	Samatrix/spl	Natural Language Processing	4	0	0	4	4	SC
3	Samatrix/spl	Blockchain and Distributed	4	0	0	4	4	SC
4	Samatrix/spl	Computer Vision	4	0	0	4	4	SC

List of Program Electives-II

Sr. No.	Course Code	Course Title	L	T	P	Contact Hrs.	Credits	Type
1	Samatrix/spl	Self-Driving Cars	4	0	0	4	4	SC
2	Samatrix/spl	DevOps – Build, Test, Deployment Automation	4	0	0	4	4	SC
3	Samatrix/spl	Virtual and Augmented Reality	4	0	0	4	4	SC
4	Samatrix/spl	Bioinformatics	4	0	0	4	4	SC

B. Tech. (common to all disciplines)-I/II Semester
Contact Hours (L-T-P): 2-0-2

BMC120D	English	2-0-0 2

Objectives

1. To enhance Professional competence in reading, writing, listening and speaking.
2. Switch the approach from providing information about the language to use the language.
3. Minimize the Grammar Translation Method of ELT while trying to replace it with Direct Method.
4. Introduce Communicative Method of ELT and focusing the teaching pedagogy on the student-centred learning rather than on the teacher-centred learning.
5. Ability to master three major forms of communications which are vital in academic and professional settings namely professional presentations, interviews and group communications respectively.
6. Providing a deep insight into the techniques for delivering effective presentations, winning job interviews, and actively participating in various forms of group communication.

UNIT 1	**Basic Writing Skills:** Tenses, Voice, Narration,
UNIT 2	**Vocabulary Building:** Word Formation, Affixes, Synonyms, Antonyms, One Word Substitution
UNIT 3	**Composition:** Composing a CV/Resume, Letter Writing, Email Writing, Précis Writing
UNIT 4	**Communication Skills:** What is Communication**,** Process, features of communication, Types**,** Flows of Communication and Barriers to communication.
UNIT 5	**Prose and Poetry:** The Gift of Magi (O' Henry), How Much Land Does a Man Need (Leo Tolstoy), Where the Mind is Without Fear (Rabindra Nath Tagore), If (Rudyard Kipling)

Course Outcomes (CO):

At the end of this course students will have:

CO1: Ability to design a language component or process to meet desired need within realistic, Constraints such as economic, environmental, social, political, ethical, scenario
CO2: Ability to analyze the usage of English words in different contexts.
CO3: An understanding of technical and academic articles' comprehension.
CO4: The ability to present oneself at multinational levels knowing the type of different standards of English

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome							Program Specific Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3
CO1			H		L				L	
CO2			L		M				H	
CO3		M								M
CO4					H			H		

H = Highly Related; M = Medium L = Low

Suggested Readings:

A. Practical English Usage. Michael Swan. OUP. 1995

B. Remedial English Grammar. F.T. Wood. Macmillan. 2007

C. On Writing Well. William Zinsser. Harper Resource Book. 2001

D. Study Writing. Liz Hamp-Lyons and Ben Heasly. Cambridge University Press. 2006.

E. Communication Skills. Sanjay Kumar and Pushp Lata. Oxford University Press. 2011.

F. Exercises in Spoken English. Parts. I-III, Hyderabad. Oxford University Press.

Communication Techniques Lab

BMC121B	Communication Technique Lab	**0-0-2 2**

- Listening Comprehension
- Phonetic Symbols and Transcription
- Stress Patterns, Intonation and Pronunciation
- Job Interviews
- Group Discussion
- Formal Presentation

B. Tech. (common to all disciplines)-I/II Semester **Contact Hours (L-T-P): 3-1-4**

BAS001C	**Engineering Mathematics-I**	**3: 1: 0**	**4**

OBJECTIVE:

The objectives of this course are to make the students:

- To increase the student's appreciation of the basic role played by mathematics in modern technology.
- Incorporate the knowledge of advanced mathematics to support their concurrent and subsequent engineering studies.
- To develop the concepts and tools that will serve as building blocks towards tackling more advanced level of mathematics that they are likely to find useful in their profession when employed in the firm/industry/corporation in public or private sector

UNIT 1	Asymptotes (Cartesian coordinates only), curvature, convexity, concavity, point of inflexion and curve tracing.
UNIT 2	Limit, continuity and partial derivatives, Euler's theorem on homogenous functions, total derivative, approximate calculations; Maxima and minima of two and more independent variables; Method of Lagrange multipliers.
UNIT 3	Beta and Gamma functions and their properties. Surface and volumes of solids of revolutions. Double integrals, change of order of integration in double integrals, Change of variables (Cartesian to polar), Applications: areas and volumes.
UNIT 4	Vectors covering, laws of vector algebra, operations- dot, cross, triple products; Vector function- limits, continuity and derivatives, geometric interpretation; Gradient, divergence and cur- formulae.
UNIT 5	Line integrals, simple connected regions, Line integrals, surface integrals, volume integral, Green's theorem, Stokes theorem and Gauss theorem.

Text Books:

1. B.V.Ramana, Higher Engineering Mathematics, Tata McGraw Hill, 2011.

Reference Books:

1. Erwin Kreyszig , Advanced Engineering Mathematics, Wiley 9th Edition, 2008
2. Maurice D. Weir and Joel Hass, Thomas Calculus, Pearson, 11th Edition, 2005.
3. Higher Engineering Mathematics- B. S. Grewal, Khanna Publications.

Course Outcomes

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

CO1 Understand the concepts of Asymptotes, curvature and curve tracing.

CO2 Understand the functions of more than one independent variable and calculate partial derivatives along with their applications .Also obtain an idea for finding the extreme values of functions of more the one variable.

CO3 Will able to integrate a continuous function of two or three variables over a bounded region and able to trace the curves.

CO4 Understand the representation of vector and its properties.

CO5 Understand line integral, surface integrals, volume integral, Green's theorem, Stokes theorem and Gauss theorem

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1	H	H			M		M					M	H	L	
CO2		M		L	M		H				L	M	M		
CO3	H	H		M	M		H			L		M	M	M	
CO4	H	M		M	L		M					M		M	
CO5	H	H			M		H					M	H	M	

H = Highly Related; M = Medium L = Low

B. Tech. (common to all disciplines)-I/II Semester
Contact Hours (L-T-P): 3-1-2

BAS 010C	Applied Physics	3:1:0	4

OBJECTIVE:
The Objectives of Applied Physics are:

1. An ability to apply profound understanding of Quantum Mechanics and its applications.
2. An understanding of free electron gas model
3. An ability to design a Laser system and its component, or process to meet desired needs within realistic constraints such as health and safety, manufacturability
4. The broad education necessary to understand special theory of relativity.
5. A knowledge of upcoming technologies like photonics

UNIT 1	**Quantum Mechanics:** Compton Effect and quantum nature of light. Compton Profile: Applications in material Science. Schrödinger's Equation: Time dependent and time independent cases. Physical interpretation of wave function and its properties, boundary conditions. Particle in one-dimensional box. **Applications of Quantum Mechanics:** Schrödinger's Equation and its Solution for Particle in three-dimensional boxes. Degeneracy. Barrier penetration and tunnel effect. Tunnelling probability. Overview of **Alpha Decay, Scanning and Tunnelling Microscopes**.
UNIT 2	**Sommerfeld's Free Electron Gas Model and its Applications:** Density of energy states, Fermi energy levels. Determination of Specific Heats of solids. Band Theory of solids: Understanding Semiconductors. Band Gap in solids. Conductivity and Mobility due to electrons and Holes. Solar Cells.
UNIT 3	**Quantum Optics: Coherence**: Spatial and temporal coherence, Coherence length, Coherence time. Q- factor for LASER. Visibility as a Measure of Coherence. Spatial Coherence and Size of the Source. Temporal Coherence and Spectral Purity. **LASER:** Theory of LASER action: Einstein's coefficients, Threshold conditions for LASER Action. Method and Mechanism of production of He-Ne LASER. Semiconductor LASER. Elementary ideas of Q-switching and Mode Locking.
UNIT 4	**Special Theory of Relativity (STR):** Idea of Relativity and Frames of References. Postulates of STR. Lorentz transformations. Relativity of length and time. **Relativity and GPS**. Velocity transformations. Variation in mass with speed. **Mass-Energy equivalence principle**. Relativistic Energy and momentum. Sagnag's formula and Optical gyroscope.
UNIT 5	**Applications of Optical Technologies:** Determination of thickness of thin films using interference technique. Elementary idea of anti-reflection coating. Optical filters. Applications of Diffraction: Bragg's law of X-Ray Diffraction. Polaroids and their industrial applications. **Holography**: Holography versus photography. Basic theory of Holography.

	Applications of Holography in Microscopy and Interferometry. **Optical Communication**: Optical fiber as optical wave-guide. Numerical Aperture and Maximum Angle of Acceptance. **Overview of Upcoming Technologies** * Photonics * Spintronics * Quantum Computers * Nanotechnology and Nano-materials. Carbon Nano-tubes (CNTs).

Course Outcomes

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

CO-1 Students would be able to describe the Quantum Mechanics and its applications.
CO-2 Students would be able to write down the band theory of Solids.
CO-3 to enable student to learn and to apply concepts learnt in Quantum optics in Industry and in real life.
CO-4 to enable students to learn the idea of **Global Positioning System (GPS)** and to explore its further applications and importance in advancement of technologies
CO-5 To identify the applications of electrodynamics using Maxwell equations

Suggested Books

1. Arthur Beiser, **Perspectives in Modern Physics**, McGraw Hill International.
2. H. S. Mani and G. K. Mehta, **Modern Physics**, East-West Press.
3. A. K. Ghatak, **Optics**, TATA McGraw Hill.
4 D. K. Bhattacharya and A. Bhaskaran: **Engineering Physics**, Oxford University Press.
5. A. K. Ghatak and Thyagrajan, Fiber Optics, Oxford University Press.
6. S. O. Pillai, **Solid State Physics**,Wiley Eastern

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	**Program Outcome**												**Program Specific Outcome**		
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
CO1			H		L		H		L					L	
CO2			L		M		L		M	H		L		H	
CO3		M								L		M			M
CO4					H								H		

CO5			H				M						H		

H = Highly Related; M = Medium L = Low

B. Tech. (common to all disciplines)-I/II Semester
Contact Hours (L-T-P): 3-0-2

BES001B	Basic Electronics Engineering	3-0-0	3

Objective

- To understand basic concepts required in understanding electronic circuits
- To understand the concept of Semiconductor Diode and their applications.
- To understand the concept of Opto-Electronic Devices.
- To understand the concept of BJT and their configurations. As well as the concept of Field Effect Transistor with their various configuration.
- The student will be able to understand fundamental circuit analysis techniques and basic electronics backgrounds, including PN Diode, BJT and MOSFET.
- The student will be able to understand the concept of Various Binary Number Systems and conversions.
- To understand Logic Gates and Logic Circuit focussing on basic and universal gates.

UNIT 1	Comparison of Insulator, conductor and semiconductor with energy band diagrams. Semiconductor materials-Intrinsic and Extrinsic semiconductor (P-type and N-type SC), Crystal structures of p-type and Ntype materials, resistivity, conductivity, mobility.
UNIT 2	Semiconductor Diode, PN diode-construction, working and V-I plot, Diode as a Rectifier, Half Wave and Full Wave Rectifiers with and without Filters with calculation of ripple factor and efficiency, Breakdown Mechanisms, Zener Diode – construction, Operation, characteristics; Opto-Electronic Devices – LEDs, Photo Diode, SCR.
UNIT 3	Bipolar Junction Transistor (BJT) – Construction, Operation, Amplifying Action, Common Base, Common Emitter and Common Collector Configurations-(construction, Properties, Input and output graphs), Operating Point, Biasing configurations: Fixed Bias, Emitter bias and Voltage Divider Bias Configuration;
UNIT 4	Field Effect Transistor (FET) – Construction, Characteristics of Junction FET, Depletion and Enhancement type Metal Oxide Semiconductor (MOS) FETs (Construction, Input characteristics and transfer characteristics).

UNIT 5	Number Systems: Binary system, Hexadecimal System, Octal system, Decimal system, Code conversions, Basic Logic Gates(AND, OR , NOT), Universal Gates(NAND and NOR) and other gates(EX-OR,EX-NOR),Truth Tables, Boolean Algebra, De Morgan's Theorems, Realization of other gates using NAND and NOR.

Course Outcome (CO):

At the end of this course students will have:

CO1-Ability to understand the physical properties of different types of semiconductors used in fabricating devices.

CO2- Ability to understand the functioning of PN junction diode and explains its main application as rectifiers and opto-electronic devices.

CO3-Ability to understand the surprising action of BJT and explains its working and biasing in three configurations

CO4-Ability to understand the working of JFET and MOSFET.

CO5-Ability to understand the concept of Various Binary Number Systems and Codes, Logic Gates and Logic Circuit.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specific Outcome		
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1	M	H								M				L	
CO2	M	H						H					L	H	L
CO3			H	M	L				L						M
CO4				H	H						L		H		
CO5						H	H					H			M

H = Highly Related; M = Medium L = Low

Text Books:

R. L. Boylestad& Louis Nashlesky (2007), Electronic Devices &Circuit Theory, Pearson Education

Reference Books

SantiramKal (2002), Basic Electronics- Devices, Circuits and IT Fundamentals, Prentice Hall, India

David A. Bell (2008), Electronic Devices and Circuits, Oxford University Press

Thomas L. Floyd and R. P. Jain (2009), Digital Fundamentals, Pearson Education

R. S. Sedha (2010), A Text Book of Electronic Devices and Circuits, S.Chand& Co.

R. T. Paynter (2009), Introductory Electronic Devices & Circuits – Conventional Flow Version, Pearson Education

B. Tech. (common to all disciplines)-I/II Semester
Contact Hours (L-T-P): 3-0-0

BES014A	Computer Programming in C *	**3: 0: 0**	**3**

Objectives

1. To introduce the parts of the computer system and number system.
2.To describe the concepts of Boolean function and languages.
3. To explain the approach of problem solving and design an algorithmic solution for a given problem with introduction of C language.
4. To prepare the students for write a maintainable C program for a given algorithm using control statements.
5. To prepare C program for simple applications of real life using functions.

UNIT 1	Computer Fundamentals: Flow chart, pseudo code. Binary, octal and hexadecimal number system.
UNIT 2	ASCII, EBCDIC and UNICODE. Boolean operations, primary and secondary memory. Difference among low-level & high-level languages.
UNIT 3	C Programming: Structure of a 'C' program, Data types, enumerated, assignment statements, input output statements,
UNIT 4	If statement, for loops, while loops, do-while loops, switch statement, break statement, continue statement. Data type conversion.
UNIT 5	Functions & program structure (function call and return), scope of variables, parameter passing methods, recursion v/s iteration.

Outcomes:

Upon completion of the subject, students will be able to:

CO1. Identify the parts of the computer system and functioning of various computer components.
CO2. Adequately describe the concepts of Boolean function and languages.
CO3. Explain approach of problem solving and design an algorithmic solution for a given problem with introduction of C language.
CO4. Write a maintainable C program for a given algorithm using control statements.
CO5. Write a C program for simple applications of real life using function.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H												H		
CO2		H													
CO3					H										
CO4				H										H	
CO5						M								M	

Reference Books:

1. Fundamental of Computers By R. Thareja, Oxford University Press.

2. Programming in ANSI C by E Balagurusamy, Tata McGraw-Hill Education.

3. The C Programming Language by Brian W. Kernighan and Dennis M. Ritchie, PHI.

4. C:The Complete Reference by Herbert Schildt, McGraw-Hill Education.

5. Let us C by Yashavant P. Kanetkar, bpb publications.

Samatrix/spl	**INTRODUCTION TO DATA SCIENCE AND FOUNDATION OF DATA ANALYTICS**	**3: 0: 0**	**3**

Software Req: MS Office 2013/2016 Version Hours: 4 per Week
Objectives: *The objective of this course is to teach students the vital data science concepts and tasks that occupy the data scientist*

UNIT – I
Introduction to Data Science: Defining Data Science and Big Data, Benefits and Uses of Data Science and Big Data, Facets of Data, Structured Data, Unstructured Data, Natural Language, Machine generated Data, Graph based or Network Data, Audio, Image, Video, Streaming data, Data Science Process, Big data ecosystem and data science, Distributed file systems, Distributed programming framework, data integration framework, machine learning framework, No SQL Databases, scheduling tools, benchmarking tools, system deployments

UNIT – II
Data Science Processes: Six steps of data science processes, define research goals, data retrieval, cleansing data, correct errors as early as possible, integrating – combine data from different sources, transforming data, exploratory data analysis, Data modelling, model and variable selection, model execution, model diagnostic and model comparison, presentation and automation.

UNIT – III
Introduction to Machine Learning: What is Machine Learning, Learning from Data, History of Machine Learning, Big Data for Machine Learning, Leveraging Machine Learning, Descriptive vs Predictive Analytics, Machine Learning and Statistics, Artificial Intelligence and Machine Learning, Types of Machine Learning – Supervised, Unsupervised, Semi-supervised, Reinforcement Learning, Types of Machine Learning Algorithms, Classification vs Regression Problem, Bayesian, Clustering, Decision Tree, Dimensionality Reduction, Neural Network and Deep Learning, Training machine learning systems

UNIT – IV
Introduction to AI: What is AI, Turing test, cognitive modelling approach, law of thoughts, the relational agent approach, the underlying assumptions about intelligence, techniques required to solve AI problems, level of details required to model human intelligence, successfully building an intelligent problem, history of AI

UNIT – V
Introduction to Data Analytics: Working with Formula and Functions, Introduction to Charts, Logical functions using Excel, Analyzing Data with Excel.

B. Tech. (common to all disciplines)-I/II Semester
Contact Hours (L-T-P): 3-0-2

BAS012A	**APPLIED PHYSICS LAB**	**0: 0: 2 2**

List of Experiments

Students are required to perform any ten experiments out of the following list of experiments.

1. To convert a Galvanometer into an Ammeter of given range and calibrate it.
2. To convert a Galvanometer into a Voltmeter of given range and calibrate it.
3. To study the variation in resistance of a Semiconductor with temperature and to determine its **energy Band-Gap**.
4. To determine specific Resistance of a wire by Carrey-Foster's Bridge.
5. To determine the height of an unknown object using Sextant.
6. To determine Resolving power of Telescope.
7. To determine Dispersive Power of a Prism using Mercury light source and Spectrometer.
8. To determine the wavelength of prominent lines of Mercury by using plane Diffraction Grating and Spectrometer.
9. To measure Numerical Aperture of an Optical Fiber.
10. To determine the profile of He-Ne LASER beam.

By graduation, students in the Engineering Physics Lab program must fulfill the following student outcomes:

CO-1 Engineering Physics graduates must have demonstrated a working knowledge of fundamental physics and basic electrical and/or mechanical engineering principles to include advanced knowledge in one or more engineering disciplines;
CO-2 the ability to formulate, conduct, analyze and interpret experiments in engineering physics; and
CO-3the ability to use modern engineering physics techniques and tools, including laboratory instrumentation.
CO-4 communicate their ideas effectively, both orally and in writing; and function effectively in multidisciplinary teams.

. **MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

Course Outcome	Program Outcome												Program Specific Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1			H		L		H		L					L	
CO2			L		M		L		M	H		L		H	
CO3		M								L		M			M
CO4					H								H		

H = Highly Related; M = Medium L = Low

B. Tech. (common to all disciplines)-I/II Semester
Contact Hours (L-T-P): 0-0-2

BES002A	**Engineering Graphics**	**0: 0: 2**	**2**

Course Objective:

- Increase ability to communicate with people
- Learn to sketch and take field dimensions.

Exercise 1: Draw sheet of Lettering, Scale: Plain Scale, Diagonal Scale,

Exercise 2: Draw sheet of Conic Curves: parabola, hyperbola & ellipse.

Exercise 3: Draw sheet of Engineering Curves: Cycloid, Epicycloid, Hypocycloid and Involute.

Exercise 4: Draw sheet of Projection of points & projection of lines.

Exercise 5: Draw sheet of Projection of planes

Exercise 6: Draw sheet of projection of solid-I

Exercise 7: Draw sheet of projection of solid-II

Exercise 8: Draw sheet of sections and section views.

Exercise 9: Draw sheet of Orthographic projections: first angle of projection.

Exercise 10: Draw sheet of Orthographic projections: Third angle of projection.

Exercise 11: Draw sheet of Isometric projections and view.

Exercise 12: Draw sheet of development of surfaces.

Course Outcomes: After learning the course the students should be able to:-

CO1. Students will be able to draw orthographic projections and sections.
CO2. Student's ability to use architectural and engineering scales will increase.
CO3. Student will be able to read drawing of given object
CO4. Student will differentiate first angle and third angle projection

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specific Outcome	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1				H	L									L
CO2			L		M									H
CO3		M												
CO4					H								H	

H = Highly Related; M = Medium L = Low

Text Books:

1. Bhat, N.D.& M. Panchal (2008), Engineering Drawing, Charotar Publishing House
2. Shah, M.B. & B.C. Rana (2008), Engineering Drawing and Computer Graphics, Pearson Education

Reference Books:

1 Dhawan, R.K. (2007), A Text Book of Engineering Drawing, S. Chand Publications

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

B. Tech. (common to all disciplines)-I/II Semester
Contact Hours (L-T-P): 0-0-2

BES012A	**COMPUTER PROGRAMMING In C LAB**	**0-0-2 2**

List of Experiments

1. Eight programs using input output statements, if statement, for loops, while loops, do-while loops, switch statement, break statement, continue statement, data type conversion etc.

2. Check a number- palindrome, prime, etc.

3. Eight programs using functions.

4. Two programs using recursion and Iteration.

B. Tech. (common to all disciplines)-I/II Semester
Contact Hours (L-T-P): 0-0-2

BES004A	BASIC ELECTRONICS ENGINEERING LAB	0-0-2

Objective

- Understand the nature and scope of modern electronics.
- Describe physical models of basic components.
- To provide students engineering skills by way of breadboard circuit design with electronic devices and components.
- To design and analyze various Electronic circuits such as PN diode,clipping ,applications of PN Diode, digital circuits etc. so that students are able to understand the practical aspects of basic electronics theory.
- Design and construct simple electronic circuits to accomplish a specific function, e.g., designing filters, clippers,clamper.
- Understand student's capabilities and limitations and make decisions regarding their best utilization in a specific situation.
- To make students understand how these small circuits are used in their day to day life.

Experiment 01	Familiarization of electronics component and equipments like C.R.O, Function generator and power supplies etc. Generate a sine wave using a function generator and measure its amplitude and frequency using C.R.O.
Experiment 02	Study of Digital Multimeter
Experiment 03	Study the following passive components (a) Study of resister (b) Study of capacitor (c) study of Inductors(d) Study of Bread Board
Experiment 04	Study of Analog & Digital ICs
Experiment 05	Determine static resistance and dynamic resistance of p-n junction diode and plot theV-I characteristics.
Experiment 06	Design and test, diode clipping circuits for peak clipping and peak detection.
Experiment 07	Design and test, positive and negative clamping circuit for a given reference voltage.
Experiment 08	Observe output waveform of half wave rectifier with and without filter capacitor and measure DC voltage, DC current, ripple factor with and without filter capacitor.

Experiment 09	Observe output waveform of full wave rectifier with and without filter capacitor and measure DC voltage, DC current, ripple factor with and without filter capacitor.
Experiment 10	Observe waveform at the output of Bridge rectifier with and without filter capacitor and measure DC voltage, DC current, ripple factor with and without filter capacitor.
Experiment 11	Design a half wave rectifier using discrete components on a breadboard and measure DC voltage, DC current, ripple factor, with and without filter capacitor
Experiment 12	Design afull wave rectifier using discrete components on a breadboard and measure DC voltage, DC current, ripple factor with and without filter capacitor.
Experiment 13	Verification of Truth table of basic & universal Gates using ICs.

Course Outcome (CO):

CO1- Ability to understand the working of diodes

CO2-Ability to understand the use of CRO and Function Generator

CO3- Ability to understand the operation of PN diode and rectifiers.

CO4- Ability to understand the circuitry which converts an AC to digital DC.

CO5- Ability to understand the designing of different types of filters and logic gates.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome							Program Specific Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3
CO1		L	L	H	H				L	
CO2		H	M	M				L	H	L
CO3			H	H			H			M
CO4			L	H	M			H		
CO5						H	H			M

H = Highly Related; M = Medium L = Low

Text Books:

R. L. Boylestad& Louis Nashlesky (2007), Electronic Devices &Circuit Theory, Pearson Education

Reference Books

SantiramKal (2002), Basic Electronics- Devices, Circuits and IT Fundamentals, Prentice Hall, India
David A. Bell (2008), Electronic Devices and Circuits, Oxford University Press
Thomas L. Floyd and R. P. Jain (2009), Digital Fundamentals, Pearson Education
R. S. Sedha (2010), A Text Book of Electronic Devices and Circuits, S.Chand& Co.
R. T. Paynter (2009), Introductory Electronic Devices & Circuits – Conventional Flow Version, Pearson Education

B. Tech. (common to all disciplines)-I/II Semester
Contact Hours (L-T-P): 0-0-2

BAS 002C	Engineering Mathematics-II	3:1:0 [4]

Objective: At the end of the course, the student should be able to:

- To provide a brief, hands-on overview of ordinary differential equations and Higher order linear differential equation with constant coefficients.
- To understand the second order linear differential equations with variable coefficients.
- To make utilization of Linear Partial differential equations – some important equations Heat, wave and Laplace equation.
- To understand the Laplace transform, Inverse Laplace transform and their applications
- To familiarize and Analyze numerical solution of a differential equation by Euler's, Modified Euler's, Predictor Corrector and Runge Kutta fourth order Methods.

UNIT1	Rank of a Matrix, Inverse of a matrix, System of linear equations (Homogenous and Non-homogeneous); Eigen values and eigen vectors, Cayley's Hamilton theorem.
UNIT2	Convergence of sequence and series, tests for convergence, power series, Taylor's series. Series for exponential, trigonometric and logarithmic functions.
UNIT3	Ordinary differential equation (first order first degree), Homogenous differential Equation, Linear differential equation, Exact differential equation, Higher order linear differential equation with constant coefficients.
UNIT4	Linear equations with variable coefficients: Homogenous form, Exact form, Change of dependent variable, Normal form, Change of independent variable and method of variation of parameters.
UNIT 5	Series solutions of second order linear differential equations with variable coefficients (Complementary functions only). First order partial differential equations, solutions of first order linear and non-linear PDEs.

Text Books: 1. B.V.Ramana, Higher Engineering Mathematics, Tata McGraw Hill, 2011.**Reference Books:**

Recommended Books:1. Erwin Kreyszig , Advanced Engineering Mathematics, Wiley 9th Edition, 2008
2. Thomas and Finney, Calculus and Analytical Geometry, Narosa Publishing House. New Delhi, 2002.
3. M.Ray and Chaturvedi, A Text Book of Differential Equations, Students Friends & Co. Publisher, Agra, 1998.
4. Maurice D. Weir and Joel Hass, Thomas Calculus, Pearson, 11th Edition, 2005.

Outcomes:
At the end of this course, students will be able to:

CO1: Use matrices, determinants and techniques for solving systems of linear equations in the different areas of Linear Algebra. Understand the definitions of Vector Space and its linear Independence.Solve Eigen value problems and apply Cayley Hamilton Theorem.

CO2: Understanding convergence of sequence and series.

CO3: Identify, analyze and subsequently solve physical situations whose behavior can be described by First order and first degree ordinary differential equations and Higher order linear differential equation with constant coefficients.

CO4: Determine solutions to second order linear differential equations with variable coefficients.

CO5: Understanding the series solutions of second order linear differential equations with variable coefficients

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1	H	M	L	L	M				L			L	M		
CO2	H	M	M	M	M				L			L	H		
CO3	H	M	M	M	M		M		L			L		H	M
CO4	H	H	M	M	M			L	L			L			M
CO5	H	H	M	M	M	L			L			L	H		

H = Highly Related; M = Medium L=Low

B. Tech. (common to all disciplines)-I/II Semester
Contact Hours (L-T-P): 3-1-0

B. Tech. (Common to all) – Semester I/II
Contact Hrs per week (L-T-P): 3-0-0

BES005C	**Basic Electrical Engineering**	**3: 0: 0**	**3**

OBJECTIVE:

The objective of this course is to provide the students with an introductory treatment of the field of Electrical Engineering.

Unit 1	**DC Circuit & Theorems –** Ohm's law, KCL & KVL, Voltage & Current Sources, Star-Delta and Delta-Star transformations, Nodal & Mesh Analysis, Superposition Theorem, Thevenin's Theorem, Norton's Theorem, Maximum Power Transfer Theorem
Unit 2	**Single Phase Circuits -** Definition of average value, root mean square value, form factor and peak factor of sinusoidal voltage and current and phasor representation of alternating quantities; Analysis with phasor diagrams of R, L, C, RL, RC and RLC circuits; Real power, reactive power, apparent power and power factor.
Unit 3	**Three Phase AC Circuits**: Necessity and Advantages of three phase systems, Generation of three phase power, definition of Phase sequence, balanced supply and balanced load; Relationship between line and phase values of balanced star and delta connections.
Unit 4	**Transformers -** Principle of operation and construction of single phase transformers (core and shell types). EMF equation, losses, efficiency and voltage regulation
Unit 5	**Rotating Electrical Machines** – Construction & Working principle of DC machine as a generator and a motor; EMF equation of DC generator; torque equation of DC motor. Back EMF of DC Motor. Applications of dc machines and single phase motors.

COURSE OUTCOMES:

- To understand, analyze and solve DC electrical circuits.
- To understand, analyze and solve single phase electrical circuits for different loads and configurations.
- To understand, analyze and solve three phase electrical circuits for different loads and configurations.
- To understand working and applications of transformers.
- To understand working and applications of different AC and DC rotating machines.

Text Books:

1. Nagsarkar and Sukhija, Basic Electrical Engineering, Oxford Uni. Press.

Reference Book:

1. Nagrath I.J. and D. P. Kothari, Basic Electrical Engineering, TMH
2. Kulshreshtha DC, Basic Electrical Engineering, Tata McGraw Hill
3. Rajendra Prasad, Fundamentals of Electrical Engineering, Prentice Hall, India
4. Hughes, E., Electrical Technology. Pearson
5. BL Theraja, A textbook of electrical technology, Vol- II, S.Chand & Co. LTD.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specific Outcome		
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CO1	H	M	M	L	L					L			M		
CO2	H	M	M	L	L					L			M	M	
CO3	H	M	M	M	M	M	M		L	L	L	L	M	M	
CO4	H	M	M	M	M	M	M		L	L	L	L	M	M	M
CO5	H	H	H	M	H	M	H		M	M	M	L	H	M	M

H = Highly Related; M = Medium L=Low

BES 007A	**Engineering Mechanics**	3-0-0	3

Course Objective: The student will able to

- Ability to identify, formulate, and solve engineering problems.
- an ability to apply knowledge of basic mathematics, science and engineering

Unit I	Statics –Basics Concepts, Fundamental principles & concepts: Vector algebra, Newton's laws, gravitation, force (external and internal, transmissibility), couple, moment (about point and about axis), Varignon's theorem, resultant of concurrent and non-concurrent coplanar forces, static equilibrium, free body diagram, reactions. Problem formulation concept; 2-D statics, two and three force members, alternate equilibrium equations, constraints and static determinacy; 3-D statics.
Unit II	Analysis of Structures- Trusses: Types of support reactions, Assumptions, rigid and non-rigid trusses; Simple truss (plane and space), analysis by method of joints. Analysis of simple truss by method of sections; Compound truss (statically determinate, rigid, and completely constrained).Analysis of frames and machines.
Unit III	Friction: Types of friction, Limiting friction, Laws of Friction, Static and Dynamic Friction; Motion of Bodies, screw jack. , Principle of Lifting Machines, Mechanical Advantage.
Unit IV	Moment of Inertia- First moment of mass and center of mass, centroids of lines, areas, volumes, composite bodies. Area moments- and products- of inertia, radius of gyration, transfer of axes, composite areas. Rotation of axes, principal area-moments-of-inertia,
Unit V	Simple stress and strain, Factor of Safety, Types of Beam and loading, Shear force and Bending Moment diagram for simple supported and cantilever Beam.

Text Books:

1. Nelson A., "Engineering Mechanics", McGraw -Hill Publication

Reference Books:

1. Timoshenko P. S. and Young D. H., "Engineering Mechanics", McGraw-Hill Publication.
2. Bhattacharyya Basudeb, "Engineeing Mechanics", Oxford University Press.
3. Engineering Mechanics by RS Khurmi.

Course Outcomes: After learning the course the students should be able to:-

CO1. Identify, formulate, and solve engineering problems
CO2. apply knowledge of basic mathematics, science and engineering.
CO3. Visualize the concept of moment of inertia for different shapes
CO4. Recognition of type of motion and forces.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specific Outcome	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	M	H		M					L					M
CO2	H	M		H			M						M	
CO3	H	H		M									H	
CO4	H	M		H								M		H

H = Highly Related; M = Medium L = Low

BES 003A	**Engineering Workshop**	0-0-2	2

Course Objective: The student will able to

- Read and interpret job drawing.
- Identify, select and use various marking, measuring, holding, striking and cutting tools & equipments.

Unit I	**Machine Shop -** Study of lathe machine, drilling machine and shaper, their parts and demonstration of operations performed on them. 1. Prepare a job on lathe machine by performing turning, facing and chamfering as per given drawing.
Unit II	**Fitting Shop -** Study of fitting tools, their uses and demonstration of operations by using different tools. 3. Prepare a job including finishing of all four sides by filing and make a square notch. 4. Prepare a job by finishing its two sides and perform drilling and taping on it.
Unit III	: **Carpentry Shop -** Study of wood and wood working, tools used in carpentry shop and their applications. 5. Prepare a T-lap/Cross lap joint. 6. Prepare a bridle joint.
Unit IV	**Welding Shop -** Definition of welding and brazing process and their applications. Study of tools used in arc and gas welding shop. 7. Prepare a /butt joint in arc welding shop. 8. Demonstration of different types of flames in gas welding shop. 9. Study of common welding defects
Unit V	: **Foundry Shop -** Study of moulding and casting process, moulding sand, foundry tools and patterns used for moulding. 10 Prepare a mould by using a given pattern. 11 Making and baking of dry sand cores for placing in horizontal, vertical and hanging positions in the mould cavity.

Text Book:

1. Hajra Choudhury Workshop Technology Vol. 1 & 2, Media Promoters & Publishers P. Ltd,Bombay.

Reference Book:

1. Chapman W. A. J., Workshop Technology Parts 1 & 2, Viva Books P. Ltd., New Delhi.

Course Outcomes: After learning the course the students should be able to:-

CO1. Understand applications of hand tools and power tools.
CO2. Understand the operations of machine tools.
CO3. Student will be able to working at shop floor.
CO4. Student will visualize casting process.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specific Outcome	
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2
CO1			M		H				M	M		L		L
CO2					H				H	M			H	L
CO3			L		H	M		M						
CO4					M		H				H			L

H = Highly Related; M = Medium L = Low

Samatrix/spl	**DATA ANALYSIS USING PYTHON**	**3:0:0**

Software: Python, NumPy, Pandas, Matplotlib,Seaborn, SciPy

Objectives: *The objective of this course is to teach students the concepts of Python Programming Language with Libraries*

UNIT – I

Python programming Basic: Python interpreter, IPython Basics, Tab completion, Introspection, %run command, magic commands, matplotlib integration, python programming, language semantics,scalar types. Control flow

Data Structure, functions, files: tuple, list, built-in sequence function, dict, set, functions, namescape, scope, local function, returning multiple values, functions are objects, lambda functions,error and exception handling, file and operation systems

UNIT – II

NumPy: Array and vectorized computation: Multidimensional array object. Creating ndarrays, arithmetic with numpy array, basic indexing and slicing, Boolean indexing, transposing array and swapping axes, universal functions, array-oriented programming with arrays, conditional logic as arrays operations, file input and output with array

Pandas: Pandas data structure, series, DataFrame, Index Object, Reindexing, dropping entities from an axis, indexing, selection and filtering, integer indexes, arithmetic and data alignment, function application and mapping, soring and ranking, correlation and covariance, unique values, values controls and membership, reading and writing data in text format

UNIT -III

Visualization with Matplotlib: Figures and subplots, colors, markers, line style, ticks, labels, legends,annotation and drawing on sublots, matplotlib configuration

Plotting with pandas and seaborn: line plots, bar plots, histogram, density plots, scatter and point plots, facet grids and categorical data

B. Tech. (Common to all) – Semester I/II
Contact Hrs per week (L-T-P): 0-0-2

BES008C	**Basic Electrical Engineering Lab**	**0:0:2**

OBJECTIVE: To provide exposure to the students with hands on experience on various basic engineering practices in Electrical Engineering.

List of experiments (Perform any 10).

1. Familiarization with (a) Electrical Symbols (b) Electrical Abbreviations (c) Electrical Tools
2. Familiarization with (a) Various Electrical Components (b) Electrical Measuring Instruments
3. To study the various Electrical Lamps Viz. Halogen Lamps, Fluorescent Tube & CFL, Sodium Vapour lamp, Neon Lamps, Incandescent Lamps and LED bulbs & Tubes
4. To make house wiring for a lamp operated from two different positions (or two way switching).
5. To make house wiring including Earthling for 1- Phase Energy meter, MCB, Ceiling Fan, Tube light, 3 Pin Plug & Socket.
6. To verify transformation ratio by measuring primary and secondary side voltages of single phase transformer
7. To measure the voltages on primary and secondary sides of three phase transformer for different configurations.
8. To study the construction & working of Ceiling fan
9. To run the single phase induction motor at varying speeds by using autotransformer.
10. To run the 3-pahse phase induction motor at varying speeds by using 3-phase auto transformer.
11. To measure Power in 3-phase load by one-wattmeter method.
12. To measure Power in 3-phase load by three-wattmeter method.
13. To measure Power and Power factor in 3-phase load by two-wattmeter method.

Old Syllabus

List of experiments (Perform all).

1. Familiarization with (a) Electrical Symbols (b) Electrical Abbreviations (c) Electrical Tools
2. Familiarization with (a) Various Electrical Components (b) Electrical Measuring Instruments
3. To study the various Electrical Lamps Viz. Halogen Lamps, Fluorescent Tube & CFL, Sodium Vapour lamp, Neon Lamps, Incandescent Lamps and LED bulbs & Tubes

4. To make house wiring for a lamp operated from two different positions (or two way switching).
5. To make house wiring including Earthling for 1- Phase Energy meter, MCB, Ceiling Fan, Tube light, 3 Pin Plug & Socket.
6. To verify transformation ratio by measuring primary and secondary side voltages of single phase transformer
7. To measure current and voltage of primary and secondary side of 3-phase transformer for different tapings.
8. To study the construction & working of Ceiling fan
9. To run the single phase induction motor at varying speeds by using autotransformer.
10. To run the 3-pahse phase induction motor at varying speeds by using 3-phase auto transformer.
11. To measure Power in 3-phase load by one-wattmeter method.
12. To measure Power in 3-phase load by three-wattmeter method.
13. To measure Power and Power factor in 3-phase load by two-wattmeter method.

OUTCOMES:

CO1. Students can now become familiar with various electrical symbols, abbreviations, tools and measuring instruments and practically connect the electrical circuits.

CO2. The ability to conduct testing and experimental procedures on different types of electrical machines.

CO3. The ability to select a suitable measuring instrument for measuring electrical and non electrical quantities for a given application. They will now be in position to learn different testing procedure of transformers and induction motors.

BES 017A	Foundation of Computing	3- 0-0	2

Unit I	**Linear Data Structure** Introduction of Data Structure; Need and Applications; Abstract Data Type; Dynamic Memory Allocation; Array; Linked List; Stack and Queues; Priority Queues - Implementation and Applications.
Unit II	**Sorting, Searching;** Sorting techniques- need; Types of Sorting, selection sort, Quick Sort; Searching techniques: need; Linear Search, Binary Search; Implementation and Applications of all.
Unit III	Introduction to Logic: Propositional Logic, Predicate Logic.
Unit IV	**Computer Networks and Internet Basics** Computer Networks and Internet, Categories of Networks: Wired, Wireless, Sensor, LAN, WAN, PAN; network topologies need, and type, network switching: Packet and Circuit Switching, Protocol Layers and their functions (example protocol on each layer), Application Layer Protocols, E-mail, FTP, WWW and HTTP. Introduction to Cyber Security and Forensics with needs.
Unit V	Introduction to Machine Learning, Artificial Intelligence and Fuzzy logic,Internet of Things, Natural Language Processing, Big Data, Mobile Computing, Cloud Computing.

Books:

1. Data structures in C by H. Sahani
2. Computer Networking: A Top-Down Approach Featuring Internet by J. F. Kurose and K. W. Ross, 3/e, Pearson Education, 2005.
3. Machine Learning by Tom Mitchel, TMH

Reference:

4. Data Structures by Tanenbum
5. Data Communications and Networking by Forouzan

BES 010A	**Engineering Mechanics lab**	0 0 2	2

Course Objective : The student will able to:

- the ability to identify and formulate elementary level engineering problems related to particle mechanics, in conceptual form as well as in terms of mathematical and physical models;
- the ability to apply the basic principles of classical particles mechanics to the analysis of particles subjected to forces;

(Minimum 12 experiments from following)

- To verify Law of Parallelogram of Forces.
- To verify Polygon law of forces.
- To determine Support Reactions of a Simply Supported Beam.
- To measure coefficient of Static Friction.
- To determine Efficiency of a Compound Lever.
- To determine Efficiency of Bell Crank Lever.
- To determine Efficiency of Worm and Worm Wheel.
- To determine efficiency of a Screw Jack.
- To determine efficiency of Double Purchase Crab Winch.
- To determine efficiency of Differential Wheel & Axle.
- To study pulley systems.
- To Verify lami's Theorem.
- To determine moment of inertia of a flywheel about its own axis of rotation.
- To determine the force in the member of simple roof truss.

- To determine the coefficient of friction between ladder and floor and between ladder and wall.

Course Outcomes: After learning the course the students should be able to:-

CO1. Visualize the concept of component of forces and force balancing.

CO2. Understanding of how machine is creating mechanical advantages and concept of efficiency.

CO3. Visualize the concept of Bending moment and shear force.

CO4. Demonstration of law of friction.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specific Outcome	
	PO1	PO2	PO3	PO4	PO5	PO6	PO 7	PO8	PO9	PO1 0	PO1 1	PO1 2	PS O1	PS O2
CO1	M	H		M					L					M
CO2	H	M		H			M						M	
CO3	H	H		M									H	
CO4	H	M		H								M		H

H = Highly Related; M = Medium L = Low

Non Credit Course

BMC 061A	**Environmental Sciences**	**0-0-0 0**

The objectives of Environment science are to-

1.Creat an awareness about environmental problems among students
2. Impart basic knowledge about the environment and its allied problems.
3. Develop an attitude of concern for the environment.
4. Motivate public through students to participate in environment protection and environment improvement.
5. Acquiring skills to help the concerned individuals in identifying and solving environmental problems.

UNIT 1	**The Multidisciplinary Nature of Environmental Studies:** The Multidisciplinary Nature of Environmental Studies Definition, scope and importance need for public awareness.
UNIT 2	**Natural Resources Renewable and Non-renewable Resources**: •Natural resources and associated problems. (a) Forest resources: Use and over-exploitation, deforestation, case studies. Timber extraction, mining, dams and their effects on forests and tribal people. (b) Water resources: Use and over-utilization of surface and ground water, floods, drought, conflicts over water, dams-benefits and problems. (c) Mineral resources: Use and exploitation, environmental effects of extracting and using mineral resources, case studies. (d) Food resources: World food problems, changes caused by agriculture and overgrazing, effects of modern agriculture, fertilizer-pesticide problems, water logging, salinity, Case studies. (e) Energy resources: Growing energy needs, renewable and non-renewable energy sources, use of alternate energy sources. Case studies. (f) Land resources: Land as a resource, land degradation, man induced landslides, soil erosion and desertification. • Role of an individual in conservation of natural resources. Equitable use of resources for sustainable lifestyles.
UNIT 3	**Ecosystems, Biodiversity and Its Conservation:** •Concept of an ecosystem.

	•Structure and function of an ecosystem. •Producers, consumers and decomposers. • Energy flow in the ecosystem. Ecological succession. •Food chains, food webs and ecological pyramids. •Introduction, types, characteristic features, structure and function of the following ecosystem: (a) Forest ecosystem (b) Grassland ecosystem (c) Desert ecosystem (d) Aquatic ecosystems (ponds, streams, lakes, rivers, oceans, estuaries) **Biodiversity and Its Conservation** •Introduction, definition: genetic, species and ecosystem diversity. •Biogeographical classification of India. • Value of biodiversity: consumptive use, productive use, social, ethical, aesthetic and option values. •Biodiversity at global, National and local levels. •India as a mega-diversity nation. Hot-spots of biodiversity. •Threats to biodiversity: habitat loss, poaching of wildlife, man-wildlife conflicts. •Endangered and endemic species of India. • Conservation of biodiversity: in-situ and ex-situ conservation of biodiversity.
UNIT 4	**Environmental Pollution:** •Definition • Causes, effects and control measures of (a) Air pollution (b) Water pollution (c) Soil pollution (d) Marine pollution (e) Noise pollution (f) Thermal pollution (g) Nuclear hazards • Solid waste management: Causes, effects and control measures of urban and industrial wastes. • Role of an individual in prevention of pollution. •Pollution case studies. •Disaster management: Foods, earthquake, cyclone and landslides.
UNIT 5	**Social Issues and the Environment, Human Population and the**

	Environment, Field Work: • From unsustainable to sustainable development. • Urban problems related to energy. •Water conservation, rain water harvesting, watershed management. • Resettlement and rehabilitation of people; its problems and concerns. Case studies. •Environmental ethics: Issues and possible solutions. •Climate change, global warming, acid rain, ozone layer depletion, nuclear accidents and holocaust. Case studies. •Wasteland reclamation. • Consumerism and waste products. • Environment Protection Act. • Air (Prevention and Control of Pollution) Act. • Water (Prevention and Control of Pollution) Act. • Wildlife Protection Act. •Forest Conservation Act. •Issues involved in enforcement of environmental legislation. •Public awareness. **Human Population and the Environment** •Population growth, variation among nations. •Population explosion—Family Welfare Programme. • Environment and human health. • Human rights. •Value education. HIV/AIDS. • Women and Child Welfare. •Role of Information Technology in environment and human health.

	Field Work • Visit to a local area to document environmental assets—river/forest/grassland/hill/ mountain. • Visit to a local polluted site—Urban/Rural/Industrial/Agricultural. • Study of common plants, insects, birds. •Study of simple ecosystems—pond, river, hill slopes, etc. (Field work equal to **5 lecture hours**) • Case Studies.

Course Outcome (CO)

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

CO-1: Recognize the history, structure, function, interactions and trends of key socio-environmental systems on personal, organizational and intellectual level regarding our surroundings through different media.

CO-2: Examine the generation of scientific knowledge and how that knowledge is presented, evaluated, framed and applied for environmental protection by conservation of Natural resources.

CO-3: Articulate a coherent philosophy of the environment and consider ethical bases for responding to environmental questions.

CO-4: Understand the role of conservation of resources and public awareness in prevention of pollution and ultimately for the sustainable development of society.

CO-5: Understand the social responsibility towards protection of environment and society

CO/PO Mapping

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7
CO-1	H	M	H	H	H	H	M
CO-2	M	H	H	M	M	H	M
CO-3	M	H	H	L	H	H	H
CO-4	M	M	H	M	H	H	H
CO-5	H	H	H	H	H	H	H

DEPARTMENT OF LAW ; JECRC UNIVERSITY

RECOMMENDED SYLLABUS FOR B TECH FIRST YEAR

BMC161A- Indian Constitution

Ser No	Recommended Subject	Number of Proposed Lecture
1.	Salient Features of the Indian Constitution	01
2.	Preamble of the Constitution	01
3.	Nature of the Constitution	01
4.	**Fundamental Rights**	
	(a) Articles 12 & 13	01
	(b) Articles 14 to 18	01
	(c) Articles 19	02
	(d) Articles 21	02
	(e) Articles 32 and Writs	01
5.	Directive Principles and Fundamental Duties	02

B.Tech CSE Semester III

BAS 007A	DISCRETE MATHEMATICS	2-0-0 [2]

Objective:

- To introduce a number of Discrete Mathematical Structures (DMS) found to be serving as tools even today in the development of theoretical computer science.
- Tosolve problems occurred in the development of programming languages.
- Tofamiliarize students with concepts and techniques of graph theory, and sets apart from languages of logic and proof methods.

UNIT 1	**Sets:** Definition and types, Set operations, Partition of set, Cardinality (Inclusion-Exclusion & Addition Principles), Recursive definition of set. Functions: Concept, Some Special Functions (Polynomial, Exponential & Logarithmic, Absolute Value, Floor & Ceiling, Mod &Div Functions), Properties of Functions, Cardinality of Infinite Set, Countable & Uncountable Sets,
UNIT 2	**Graph Theory**: Graphs – Directed, Undirected, Simple,. Adjacency & Incidence, Degree of Vertex, Subgraph, Complete graph, Cycle & Wheel Graph, Bipartite & Complete Bipartite Graph, Weighed Graph, Union of Simple Graphs. Complete Graphs. Isomorphic Graphs, Path, Cycles & Circuits Euclerian& Hamiltonian Graphs. Planar Graph: Kuratowski's Two Graphs, Euler's Formula, Kuratowski's Theorem. Trees: Spanning trees- Kruskal'sAlgo, Finding Spanning Tree using Depth First Search, Breadth First Search, Complexity of Graph, Minimal Spanning Tree.
UNIT 3	**Semigroups, Groups and Coding**: Binary Operations, Semigroups, Products and Quotients of Semigroups, Groups, Product and Quotients of Groups, Coding of Binary Information and Error Correction, Decoding and Error Correction. **Language of Logic**: Proposition, Compound Proposition, Conjunction, Disjunction, Implication, Converse, Inverse &Contrapositive, Biconditional Statements, tautology, Contradiction & Contingency, Logical Equivalences, Quantifiers, Arguments.
UNIT 4	**Proof Methods**: Vacuous, Trivial, Direct, Indirect by Contrapositive and Contradiction, Constructive & Non-constructive proof, Counterexample. The Division Algorithm, Divisibility Properties (Prime Numbers & Composite Numbers), Principle of Mathematical Induction, The Second Principle of Mathematical Induction, Fundamental Theorem of Arithmetic. Algorithm Correctness: Partial Correctness, Loop Invariant. Testing the partial correctness of linear & binary search, bubble & selection sorting.
UNIT 5	Relations: Boolean Matrices, Binary Relation, Adjacency Matrix of Relation, Properties of Relations, Operations on Relations, The Connectivity Relations, Transitive Closure-Warshall's Algorithm, Equivalence relations- Congruence Relations, Equivalence Class, Number of Partitions of a Finite Set, Partial & Total Orderings.

Course Outcome (CO):

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

CO1: Demonstrate complete knowledge on various discrete structures available in literature.
CO2: Realization of some satisfaction of having learnt that discrete structures are indeed useful in computer science and engineering and thereby concluding that no mistake has been done in studying this course.
CO3: Gaining of some confidence on how to deal with problems which may arrive in computer science and engineering in near future.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	L		H		M						L				
CO2		H		H								M	M		
CO3								H	M	L				H	

-
- H = Highly Related; M = Medium L = Low

Text Books

1. B.Kolman et.al- Discrete mathematical Structures, 5th Edn, Pearson Education, New Delhi - 2004.

Reference Books

1. K.H. Rosen – Discrete Mathematics and Its Applications – 4th Edn, Tata McGraw Hill, New Delhi – 2001
2. J.P. Tremblay et.al – Discrete Mathematical Structures with Applications to Computer Science, TMH, New Delhi – 2004.

3. Mott. J.L., Kandel A. and Baker, T.P. "Discrete mathematics", for computer scientists and Mathematicians", Second Edition, Prentice Hall 1986.

4. Tremblay J.P. and Manohar, R. "Discrete Mathematical Structures with Applications to Computer Science", McGraw Hill, 1975.

BCO 001B	SOFTWARE ENGINEERING	3-0-0 [3]

Objective

- To learn about generic models of software development process.
- To understand fundamental concepts of requirements engineering and Analysis Modeling.
- To understand the different design techniques and their implementation.
- To learn various testing and maintenance measures

UNIT 1	Introduction- Introduction to Software Engineering, Software Components, Software Characteristics, Software Crisis, Engineering aspects of Software production – necessity of automation .Job responsibilities of Programmers and Software Engineers as Software developers.Software Development Life Cycle (SDLC)
UNIT 2	Process Models and Program Design Techniques- Software Development Process Models – Code & Fix model, Waterfall model, Incremental model, Rapid Prototyping model, Spiral (Evolutionary) model.Software Requirement Specifications (SRS), Management of User Needs, Data Flow Diagrams, Entity Relationship Diagrams, Decision Tables, SRS Document, Design Techniques – Structured Programming, Coupling and Cohesion, Abstraction and Information Hiding, SoftwareModeling Tools –Data flow Diagrams, UML and XML.
UNIT 3	Software Testing: Testing Objectives, Unit Testing, Integration Testing, Acceptance Testing, Regression Testing, Verification and Validation: Testing of Software Products – Black-Box Testing and White-Box Testing, Static Analysis, Symbolic Execution and Control Flow Graphs –Cyclomatic Complexity.
UNIT 4	Software Project Management: Management Functions and Processes, Project Planning and Control, Organization and Intra-team Communication, Risk Management. Software Cost Estimation – underlying factors of critical concern. Metrics for estimating costs of software products – Function Points. Techniques for software cost estimation –Expert judgment, Work break-down structure and Process breakdown structure, COCOMO and COCOMO-II.
UNIT 5	Software Maintenance,Need for Maintenance, Categories of Maintenance, An Overview of CASE Tools.

Course Outcome (CO):

At the end of this course students will have:

CO1: An ability to design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability.
CO2: An ability to identify, formulates, and solve engineering problems.
CO3: An understanding of professional and ethical responsibility.
CO4: The broad education necessary to understand the impact of engineering solutions in a global, economic, environmental, and societal context.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	L		H		M						L				
CO2		H		H								M	M		
CO3								H	M	L				H	
CO4	L		L			L	M								L

H = Highly Related; M = Medium L = Low

Text Books:

1. Fundamentals of Software Engineering – Carlo Ghezziet. Et.al.
2. Software Engineering – Design, Reliability Management – Pressman.

Reference Books:

1. Software Engineering – Ian Sommerville.
2. Software Engineering - Shoeman.
3. Software Engineering with Abstraction – Berzins and Luqi
4. Pankaj Jalote, Software Engineering, Wiley

BCO 002A	**DATA STRUCTURES AND ALGORITHMS**	**3-0-0 [3]**

OBJECTIVE:

- To study various data structure concepts like Stacks, Queues, Linked List, Trees and Files
- To overview the applications of data structures.
- To be familiar with utilization of data structure techniques in problem solving.
- To have a comprehensive knowledge of data structures and algorithm.
- To carry out asymptotic analysis of algorithm.

UNIT 1	Introduction: Notions of data type, abstract data type and data structures. Importance of algorithms and data structures in programming. Notion of Complexity covering time complexity, space complexity, Worst case complexity & Average case complexity. BigOh Notation, Omega notation, Theta notation. Examples of simple algorithms and illustration of their complexity. Sorting- Bubble sort, selection sort, insertion sort, Quick sort; Heap sort; Merge sort; Analysis of the sorting methods. Selecting the top k elements. Lower bound on sorting.
UNIT 2	Stack ADT, Infix Notation, Prefix Notation and Postfix Notation. Evaluation of Postfix Expression, conversion of Infix to Prefix and Postfix Iteration and Recursion- Problem solving using iteration and recursion with examples such as binary search, Fibonacci numbers, and Hanoi towers. Tradeoffs between iteration and recursion.
UNIT 3	List ADT. Implementation of lists using arrays and pointers. Stack ADT. Queue ADT. Implementation of stacks and queues. Dictionaries, Hash tables: open tables and closed tables. Searching technique- Binary search and linear search, link list- single link list, double link list, Insertion and deletion in link list.
UNIT 4	Binary Trees- Definition and traversals: preorder, post order, in order. Common types and properties of binary trees. Binary search trees: insertion and deletion in binary search tree worst case analysis and average case analysis. AVL trees. Priority Queues -Binary heaps: insert and delete min operations and analysis.
UNIT 5	Graph: Basic definitions, Directed Graphs- Data structures for graph representation. Shortest path algorithms: Dijkstra (greedy algorithm) and Operations on graph, Worshall's algorithm , Depth first search and Breadth-first search. Directed acyclic graphs. Undirected Graphs, Minimal spanning trees and algorithms (Prims and Kruskal) and implementation. Application to the travelling salesman problem.

Course OUTCOME (CO):

CO1: Show the understanding of various data structure concepts like Stacks, Queues, Linked List, Trees and Files
CO2: Understand the applications of data structures.
CO3: Understand with utilization of data structure techniques in problem solving.
CO4: Use comprehensive knowledge of data structures and algorithm.
CO5: Use asymptotic analysis of algorithm.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H												M		
CO2			H		M								M		
CO3		H							M			L		H	
CO4	H	M											L		L
CO5		M		H											L

H = Highly Related; M = Medium; L = Low

Text Books:

1. Data Structures and Algorithms by Alfred V. Aho, Jeffrey D. Ullman and John E. Hopcroft , Addison-Wesley Series (1983)

Reference Books:

1. T.H. Cormen, C.E. Leiserson, and R.L. Rivest. Introduction to Algorithms.The MIT Press and
2. McGraw-Hill Book Company, Cambridge, Massacusetts, 1990 (Available in Indian Edition).
3. Steven S. Skiena. The Algorithm Design Manual.Springer, Second Edition, 2008.
4. Data Structures and Algorithm Analysis in Java (3rd Edition) by Mark Allen Weiss, Addison Wesley(2011).

Samatrix/spl	**R PROGRAMMING FOR DATA SCIENCE AND DATA ANALYSIS**	**2-0-0 [2]**

Software Req: MS Office 2013/2016 Version

Objectives: *The objective of this course is to teach students R Programming Language, basic functions in R programming language and critical techniques*

UNIT – I

Getting Started with R and R Workspace: Introducing R, R as a programming Language, the need of R, Installing R, RStudio, RStudio's user interface, console, editor, environment pane, history pane, file pane, plots pane, package pane, help and viewer pane R Workspace, R's working directory, R Project in R Studio, absolute and relative path, Inspecting an Environment, Inspect existing Symbols, View the structure of object, Removing symbols, Modifying Global Options, Modifying warning level, Library of Packages, Getting to know a package, Installing a Package from CRAN, Updating Package from CRAN, Installing package from online repository, Package Function, Masking and name conflicts

UNIT – II

Basic Objects and Basic Expressions: Vectors, Numeric Vectors, Logical Vectors, Character Vectors, subset vectors, Named Vectors, extracting element, converting vector, Arithmetic operators, create Matrix, Naming row and columns, subsetting matrix, matrix operators, creating and subsetting an Array, Creating a List, extracting element from list, subsetting a list, setting value, creating a value of data frame, subsetting a data frame, setting values, factors, useful functions of a data frame, loading and writing data on disk, creating a function, calling a function, dynamic typing, generalizing a function. Assignment Operators, Conditional Expression, using if as expression and statement, using if with vectors, vectorized if: ifelse, using switch, using for loop, nested for loop, while loop

UNIT – III

Working with Basic Objects and Strings: Working with object function, getting data dimensions, reshaping data structures, iterating over one dimension, logical operators, logical functions, dealing with missing values, logical coercion, math function, number rounding functions, trigonometric functions, hyperbolic functions, extreme functions, finding roots, derivatives and integration, Statistical function, sampling from a vector, Working with random distributions, computing summary statistics, covariance and correlation matrix, printing string, concatenating string, transforming text, Formatting text, formatting date and time, formatting date and time to string, finding string pattern, using group to extract data, reading data

UNIT – IV

Working with Data – Visualize and Analyze Data: Reading and Writing Data, importing data using built-in-function, READR package, export a data frame to file, reading and writing Excel worksheets, reading and writing native data files, loading built-in data sets, create scatter plot, bar chart, pie chart, histogram and density plots, box plot, fitting linear model and regression tree

Samatrix/spl	**PROBABILISTIC MODELLING AND REASONING WITH PYTHON AND R**	**2-0-0 [2]**

Software: Python, NumPy, Pandas, Matplotlib, Seaborn, SciPy
Objectives: The objective of this course is to teach students the concepts of Statistics, probability, probability distribution, and other statistical methods to solve various engineering problems

UNIT – I
Introduction to Statistics: Introduction to Statistics. Role of statistics in scientific methods, current applications of statistics.
Scientific data gathering: Sampling techniques, scientific studies, observational studies, data management.
Data description: Displaying data on a single variable (graphical methods, measure of central tendency, measure of spread), displaying relationship between two or more variables, measure of association between two or more variables.

UNIT – II
Probability Theory: Sample space and events, probability, axioms of probability, independent events, conditional probability, Bayes' theorem.
Random Variables: Discrete and continuous random variables. Probability distribution of discrete random variables, binomial distribution, poisson distribution. Probability distribution of continuous random variables, The uniform distribution, normal (gaussian) distribution, exponential distribution, gamma distribution, beta distribution, t-distribution, χ" distribution. Expectations, variance and covariance. Probability Inequalities. Bivariate distributions

UNIT -III
Point Estimations: Methods of finding estimators, method of moments, maximum likelihood estimators, bayes estimators. Methods of evaluating estimators, mean squared error, best unbiased estimator, sufficiency and unbiasedness
Interval Estimations: Confidence interval of means and proportions, Distribution free confidence interval of percentiles

UNIT - IV
Test of Statistical Hypothesis and p-values: Tests about one mean, tests of equality of two means, test about proportions, p-values, likelihood ratio test, Bayesian tests
Bayesian Statistics: Bayesian inference of discrete random variable, Bayesian inference of binomial proportion, comparing Bayesian and frequentist inferences of proportion, comparing Bayesian and frequentist inferences of mean

Univariate Statistics using Python: Mean, Mode. Median, Variance, Standard Deviation, Normal Distribution, t-distribution, interval estimation, Hypothesis Testing, Pearson correlation test, ANOVA F-test

BEE009A	DIGITAL SYSTEMS	4-0-0 [4]

OBJECTIVE:

1. To provide a comprehensive introduction to digital logic design leading to the ability to understand number system representations, binary codes, binary arithmetic and Boolean algebra, its axioms and theorems, and its relevance to digital logic design.
2. To provide introduction to combinational circuits(such as Karnaugh maps),synchronous sequential logic and Asynchrnous sequential logic.

UNIT1	IC Digital Logic Families - Characteristics of digital IC's, Transistor – TransistorLogic family, Standard TTL characteristics, Other TTL series, Open collector TTL, WiredOR/AND connection, Tristate TTL, Emitter-Coupled Logic family, ECL NOR/OR gate
UNIT2	Simplification of Boolean Functions - Using Karnaugh map and Quine-Mccluskey methods, SOP, POS simplification, NAND and NOR implementations,other two-level implementation (AND-OR-INVERT).
UNIT 3	Combinational Logic Design- Design procedure, Adder : Half adder, Full adder,Serial adder, Parallel adder & Carry look-ahead adder, Subtractors : Half subtractor&Fullsubtractor, BCD to Excess-3 code convertor, BCD to 7-segment decoder, Parity generator and checker .
UNIT 4	Combinational Logic Design using MSI Circuits - Application of typical IC"s like4-bit parallel adder (ex : 7483), Encoders (ex :74148), Multiplexers (ex: 74151, 74153,74157) and their use in realizing boolean functions, Multiplexer trees, Demultiplexer /Decoders (e.g.: 74138, 74154) and their use in realizing a boolean function and demultiplexertrees, 4- it magnitude comparator (ex:7485).
UNIT 5	Synchronous Sequential Logic- Analysis of clocked sequential logic, Statereduction and assignment, Flip-flop excitation tables, Design procedure, Design ofsequentialciruits ex : 3-bit up/down counter (mod < 8), 3-bit up/down gray code counter,Serial adder.

Text Books:

1. M Morris Mano, Digital Design, 3rd Edition, 2006, PHI
2. R. P Jain, Modern Digital Electronics, Second Edition, TMH

Reference Books:

1. Tocci : Digital Systems PHI , 6e, 2001
2. Bignell&Donovan Digital Electronics, 4th Edition, 2007, Thomson Learning.

BCO 082A	PROGRAMMING WITH PYTHON LAB	0-0-2 [2]

List of Experiments

1. Write a Python program to print the documents (syntax, description etc.) of Python built-in function(s).
2. Write a Python program which accepts the radius of a circle from the user and compute the area.
3. Write a Python program to accept a filename from the user print the extension of that.
4. Write a Python program to print the calendar of a given month and year.
5. Write a Python program to calculate number of days between two dates.
6. Write a Python program to calculate the length of a string.
7. Write a Python program to multiplies all the items in a list.
8. Write a Python script to sort (ascending and descending) a dictionary by value.
9. Write a Python program to create a tuple with different data types.
10. Write a Python program to find those numbers which are divisible by 7 and multiple of 5, between 1500 and 2700 (both included).
11. Write a Python program to guess a number between 1 to 9.(User is prompted to enter a guess. If the user guesses wrong then the prompt appears again until the guess is correct, on successful guess, user will get a "Well guessed!" message, and the program will exit.)
12. Write a Python program to count the number of even and odd numbers from a series of numbers.
13. Write a Python function to find the Max of three numbers.
14. Write a Python function to sum all the numbers in a list.
15. Write a Python function that takes a list and returns a new list with unique elements of the first list.
16. Write a Python class to find validity of a string of parentheses, '(', ')', '{', '}', '[' and ']. These brackets must be close in the correct order, for example "()" and "()[]{}" are valid but "[)", "({[)]" and "{{{" are invalid.
17. Write a Python class to find a pair of elements (indices of the two numbers) from a given array whose sum equals a specific target number.
18. Write a Python class to implement pow(x, n).

Course Outcome:

CO1: Various core programming basics—including data types, control structures, algorithm development,
CO2: Overview the applications of Python.
CO3: Show the program design with functions—via the Python programming language.
CO4: Students will solve problems, explore real-world software development challenges, and create practical and contemporary applications

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H		M										H		
CO2		L			M									L	L
CO3			H		M						L		M		L
CO4		M		L					L	L				M	

H = Highly Related; M = Medium L = Low

BCO 005A	DATA STRUCTURE AND ALGORITHMS LAB	0-0-2 [2]

List of Experiments

1.Write a program to implement following searching algorithms using array data structure
- 1.1 Matrix Addition and Subtraction
- 1.2 Matrix Multiplication and Transpose

2.Write a program to implement following searching algorithms using array data structure
- 2.1. Linear Search
- 2.2. Binary Search

3. Write a program to implement following searching algorithms using array data structure
- 3.1. Insertion Sort
- 3.2 Bubble Sort

4. Write a program to implement following searching algorithms using array data structure
- 4.1. Selection Sort
- 4.2 Quick Sort

5. Write a program to implement following operations on stack using array data structure.
- 5.1 Traversing
- 5.2 Push
- 5.3 POP

6. Write a program to implement following examples of recursion
- 6.1 Fibonacci Series
- 6.2 Factorial Function
- 6.3 Tower of Hanoi

7. Write a program to implement Merge Sort.

8. Write a program to implement following operations on Queue using array data structure.
- 8.1 Insertion8.2 Deletion8.3 Traversing

9. Write a program to implement Postfix evaluation.

10. Write a program to implement Infix to Postfix Notation.

11. Write a program to implement following operations on Link List data structure.
- 11.1 Insertion at beginning
- 11.2 Insertion at last
- 11.3 Insertion at any location

12. Write a program to implement following operations on Link List data structure.
- 12.1 Deletion at beginning
- 12.2 Deletion at last
- 12.3 Deletion at any location

13. Write a program to implement Doubly Link List
- 13.1 Insertion13.2 Traversing

14. Write a program to implement Breadth First Search Algorithm.

15. Write a program to implement Depth First Search Algorithm.

Course Outcomes:

Having successfully completed this course, the student will be able to:

CO1: Apply knowledge of computing and mathematics to choose the data structures that effectively model the information in a problem.

CO2: Solve problems by using iterative and recursive methods

CO3: Write various operations like searching, sorting, insertion, deletion, traversing etc. on different data structure.

CO4: Apply programming concepts to solve different problems based on data structures.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO
CO1	H												M		
CO2			H		M								M		
CO3		H							M			L		H	
CO4	H	M											L		L
CO5		M		H											L

H = Highly Related; M = Medium; L = Low

BEE010A	DIGITAL SYSTEMS LAB	0-0-2(2)

List of Experiments

1. Truth Table verification – NAND gate, NOR gate, OR gate, AND gate, NOT gate.
2. Verifying if NAND gate is a universal gate.
3. Verifying if NOR gate is a universal gate.
4. Realizing given truth table using SOP form.
5. Realizing given truth table using POS form.
6. Design and Implementation of Adder and Subtractor.
7. Design and Implementation of Multiplexer and Demultiplexer.
8. Design and Implementation of Binary to gray code converters and vice-versa.
9. Design and Implementation of BCD Adder.
10. Design and Implementation of encoder and decoder.
11. Design and Implementation of parity generator and detector.
12. Design and Implementation of Magnitude Comparator.
13. Design and Implementation of flip flops – RS, JK, D and T flip flops.
14. Design and Implementation of 3-bit synchronous up/down counter.
15. Design and Implementation of SISO, SIPO, PISO and PIPO shift registers using Flip-flops

BCO 080B	LINUX PROGRAMMING LAB	0-0-2 [2]

Lab Experiments:

1.a) Study of Unix/Linux general purpose utility command list: man,who,cat, cd, cp, ps, ls, mv, rm, mkdir, rmdir, echo, more, date, time, kill, history, chmod, chown, finger, pwd, cal, logout, shutdown.

b) Study of vi editor.

c) Study of Bash shell, Bourne shell and C shell in Unix/Linux operating system.

d) Study of Unix/Linux file system (tree structure).

e) Study of .bashrc, /etc/bashrc and Environment variables.

2. Write a C program that makes a copy of a file using standard I/O, and system calls

3. Write a C program to emulate the UNIX ls –l command.

4. Write a C program that illustrates how to execute two commands concurrently with a command pipe. Ex: - ls –l | sort

5. Write a C program that illustrates two processes communicating using shared memory

6. Write a C program to simulate producer and consumer problem using semaphores

7. Write C program to create a thread using pthreads library and let it run its function.

8. Write a C program to illustrate concurrent execution of threads using pthreads library.

Extra Programs

1. Write a shell script that accepts a file name, starting and ending numbers as arguments and displays all the lines between the given line numbers.

2. Write a shell script that displays a list of all files in the current directory to which the user has read, write and execute permissions.

Course Outcome (CO):

At the ends of this course studentswill have:

CO1: The practical knowledge of UNIX/Linux Operating System commands.
CO2: Be able to work confidently in Unix/Linux environment
CO3: Be able to write shell scripts to automate various tasks.
CO4: Be able to learn the important Linux/UNIX library functions and system calls.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H		L										L		
CO2								M					M	M	
CO3				M	M									L	
CO4							M					H			L

H = Highly Related; M = Medium L = Low

BCO 011A	COMPUTER NETWORKS	4-0-0 [4]

OBJECTIVES:

- To build an understanding of the fundamental concepts of computer networking.
- To familiarize the student with the basic taxonomy and terminology of the computer networking area.
- To introduce the student to advanced networking concepts, preparing the student for entry Advanced courses in computer networking.
- To allow the student to gain expertise in some specific areas of networking such as the design and maintenance of individual networks.

UNIT 1	Introduction -Hardware and software, Data communication, Networking, Protocols and standards. Data transmission concepts. Analog and digital transmission. Transmission impairments. Layered Architecture of Computer Networks, OSI and TCP/IP architectures Physical Layer- Guided transmission media and wireless transmission, Data encoding - Digital and analog data. Data communication interface - asynchronous and synchronous transmission, Data link layer - Flow control. Error detection and error control. HDLC and other data link protocols. Multiplexing – Frequency-division, synchronous time-division, and statistical time-division multiplexing
UNIT 2	Link Layer :Medium Access Control: CDMA, ALOHA, and Ethernet; Link Layer Addressing and Forwarding; Spanning Trees; The Channel Allocation Problem, Multiple Access Protocols, Ethernet, Wireless LANs, Broadband Wireless, Bluetooth, Data Link Layer Switching, Switched networks. Circuit-switched networks, switching concepts, Routing incircuit-switched networks. Control signaling. Packet switching principles. Routing and congestion control
UNIT 3	Network Layer: Network layer design issues. Routing algorithms , Flooding, Shortest path routing, Link Sate routing, Hierarchical routing, Broadcast and multicast routings, Routing in the Internet, Path Vector routing, OSPF routing. The network layer in the Internet: IP protocol: ARP and RARP, BOOTP, ICMP, DHCP, Network Address Translation(NAT) Internetworking
UNIT 4	Transport Layer:TCP introduction, Reliable/Un- Reliable Transport, TCP, UDP, Congestion Control, Intra-Domain Routing: Distance-Vector, Intra-Domain Routing: Link- State, Wireless Networks: 802.11 MAC, Efficiency considerations
UNIT 5	Application Layer: DNS-The Domain Name System, Electronic Mail, HTTP, FTP, Simple network management protocol (SNMP), The World Wide Web

***Course Outcome (CO) of* Computer Network**

At the end of this course studentswill have:

CO1: To provide an in-depth understanding of the terminology of network and concepts of OSI reference model and TCP/IP model.
CO2: To equip our students with technical concept of protocols, network interfaces, and design/performance issues in networks.
CO3: To be familiar with contemporary issues in networking technologies.

CO4: To be familiar with network tools and to enhance analytical skills to develop innovative solutions.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H			L									H		
CO2	M		H									L		L	
CO3		M							M				M		L
CO4					H										L

H = Highly Related; M = Medium L = Low

Text Books:

1. Computer Networks, by Andrew S Tanenbaum, PHI. (2010)

Reference Books:

- Data Communications, Computer networking on OSI , by Fred Halsall, Addison Wesley Publishing Co.1998
- Computer Networking -A Top-Down Approach Featuring the Internet ,James F. Kurose and Keith W. Ross ,Addison Wesley Publishing Co. 2004
- Computer Networks: Protocols standards and interfaces , by Uyless Black, Prentice Hall.2002
- Data communication & Networks , by Behrou A. Forouzan, Tata McGraw Hill. 2002
- Data and Computer Communications, by Walliam Stallings, PHI. (2002)

B.Tech CSE Semester IV

BCO 007A	COMPUTER GRAPHICS	3-0-0 [3]

OBJECTIVE:

- To provide students with a foundation in graphical applications programming
- To introduce students with fundamental concepts and theory of computer graphics
- To give basics of application programming interface (API) implementation based on graphics pipeline approach

UNIT 1	**Introduction to Computer Graphics:**Overview of Computer Graphics, Computer Graphics Application and Software, Description of some graphics devices, Input Devices for Operator Interaction, Active and Passive Graphics Devices,Storage Tube Graphics Displays, Calligraphic Refresh Graphics Displays, Raster Refresh (Raster-Scan) Graphics Displays, Cathode Ray Tube Basics, Color CRT Raster Scan Basics, Video Basics, The Video Controller, Random-Scan Display Processor, LCD displays.
UNIT 2	**Scan conversion – lines, circles and Ellipses; Filling polygons and clipping algorithms:**Scan Converting Lines, Mid-point criteria, Problems of Aliasing, end-point ordering and clipping lines, Scan Converting Circles, Scan Converting Ellipses, Filling Polygons, edge data structure, Clipping Lines algorithms Cohen-Sutherland and Liang-Barsky, Clipping Polygons, problem with multiple components.
UNIT 3	.**Two-Dimensional Transformations:**Transformations and Matrices, Transformation Conventions, 2D Transformations, Homogeneous Coordinates and Matrix Representation of 2D Transformations, Translations and Homogeneous Coordinates, Rotation, Reflection, Scaling, Combined Transformation, Transformation of Points, Transformation of The Unit Square, Solid Body Transformations, Rotation About an Arbitrary Point, Reflection through an Arbitrary Line, A Geometric Interpretation of Homogeneous Coordinates, The Window-to-Viewport Transformations.
UNIT 4	**Three-Dimensional Transformations:**Introduction, Three-Dimensional Scaling, Three-Dimensional Shearing, Three-Dimensional Rotation, Three-Dimensional Reflection, Three-Dimensional Translation, Multiple Transformation, Rotation about an Arbitrary Axis in Space, Reflection through an Arbitrary Plane, Matrix Representation of 3D Transformations, Composition of 3D Transformations, Affine and Perspective Geometry, Perspective Transformations, Techniques for Generating Perspective Views, the Perspective Geometry and camera models, Orthographic Projections, Axonometric Projections, Oblique Projections, View volumes for projections
UNIT 5	**Visible-Surface Determination :**Techniques for efficient Visible-Surface Algorithm Categories of algorithms, Back face removal, The z-Buffer Algorithm, Scan-line metho Painter's algorithms (depth sorting), Area sub-division method, BSP trees, Visibl Surface Ray Tracing, comparison of the methods. **Illumination and Shading** Illumination and Shading Models for Polygons, Reflectan properties of surfaces, Ambient, Specular and Diffuse reflections, Atmosphe

	attenuation, Phong's model, Gouraud shading, some examples.

Course Outcome (CO):

At the ends of this course studentswill have:

CO1: Understand the structure of modern computer graphics system
CO2: Understand the basic principles of implementing computer graphics primitives.
CO3: Familiarity with key algorithms for modeling and rendering graphical data
CO4: Develop design and problem solving skills with application to computer graphics

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H			M	L								M		
CO2			M	L					L		L			L	
CO3			L		L						M	L	L		M
CO4			H							L					L

H = Highly Related; M = Medium L = Low

Text Books:

1. Donald Hearn and Pauline Baker, Computer Graphics with OpenGL (third edition), Prentice Hall, 2003

Reference Books:

1.F. S. Hill Jr. and S. M. Kelley, Computer Graphics using OpenGL (third edition), Prentice Hall, 2006
2. Peter Shirley and Steve Marschner, Computer Graphics(first edition), A. K. Peters, 2010
3. Edward Angel, Interactive Computer Graphics. A Top-Down Approach Using OpenGL (fifth Edition), PearsonEducation, 2008

BCO 008B	OPERATING SYSTEMS(UNIX PROGRAMMING)	3-0-0 [3]

OJECTIVE:

- To understand the structure and functions of OS
- To learn about Processes, Threads and Scheduling algorithms
- To understand the principles of concurrency and Deadlocks
- To learn various memory management schemes
- To study I/O management and File systems

UNIT 1	Introduction : Operating system and functions, Classification of Operating systems- Batch, Interactive, Time sharing, Real Time System, Multiprocessor Systems, Multiuser Systems, Multiprocess Systems, Multithreaded Systems, Operating System Structure- Layered structure, System Components, Operating System services, Monolithic and Microkernel Systems.
UNIT 2	Process Management-Process & Threads – Process States - Process Control Block – Process Scheduling – Operations on Processes, Threads, CPU Scheduler – Preemptive and Non- Preemptive; Dispatcher, Scheduling Criteria, Scheduling Algorithms – Process Management in UNIX
UNIT 3	Process Synchronization & Inter process Communication-Concurrent Processes, Co-operating Processes, Precedence Graph, Hierarchy of Processes, Critical Section Problem – Two process solution, Synchronization Hardware, Semaphores – Deadlock- detection, handling, prevention, avoidance, recovery, Starvation, Critical Regions, Monitors, Inter process communication
UNIT 4	Memory Management-Objectives and functions, Simple Resident Monitor Program (No design), Overlays – Swapping; Schemes – Paging – Simple, Multi-level Paging; Internal and External Fragmentation; Virtual Memory Concept, Demand Paging – Page Interrupt Fault, Page Replacement Algorithms; Segmentation – Simple, Multi-level, Segmentation with Paging, Memory Management in UNIX.
UNIT 5	I/O Management and Disk Scheduling: I/O devices, and I/O subsystems, I/O buffering, Disk storage and disk scheduling. File System: File concept, File organization and access mechanism, File directories, and File sharing, File system implementation issues, File system protection and security.

Course Outcome (CO):

At the ends of this course studentswill have:

CO1: Classify Unix Kernel mode with user mode & contrast between Kernel structures.
CO2: Identify and estimate process management & thread management strategies along with their different operations
CO3:Implement different system calls for various file handling operations.
CO4:determine paging and Caching techniques related to Virtual Memory.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H			M				L			L		H		L
CO2		M	L						M					M	
CO3			M		M									M	M
CO4	M		L									L			

H = Highly Related; M = Medium L = Low

Text Books:

1. Operating Systems Concepts – Silberschatz, Galvin, Wiley Publications (2008)
2. Modern Operating Systems - Andrew S. Tanenbaum, Pearson Education Asia / PHI(2005)

Reference Books:

1. Operating Systems – William Stallings, Pearson Education Asia (2002)
2. UNIX System Programming Using C++, by Terrence Chan: Prentice Hall India, 1999.
3. Advanced Programming in UNIX Environment, by W. Richard Stevens: 2nd Ed, Pearson Education, 2005

BCO 009B	COMPUTER ORGANIZATION AND DESIGN	3-1-0 [4]

OJECTIVE:

- To understand the basic structure and operation of digital computer
- To study the design of arithmetic and logic unit and implementation of fixedpointand floating-point arithmetic operations
- To study the two types of control unit techniques and the concept ofpipelining
- To study the hierarchical memory system including cache memories andvirtual memory
- To study the different ways of communicating with I/O devices and standardI/O interfaces

UNIT 1	Basic organization of computers, Register transfer language (RTL), Bus and memory transfer, Arithmetic, logic, shift -micro operations, Types of registers and machine instructions, Fetch, decode and execute cycle.
UNIT 2	Assembly language programming, Instruction format, addressing modes , RISC vs CISC architectures.
UNIT 3	Information representation, Floating point representation (IEEE 754), computer arithmetic and their implementation; Fixed-Point, Signed and 2's complement Arithmetic: Addition, Subtraction, Multiplication and Division, Hardwired and Micro programmed Control.
UNIT 4	Memory Technology, static and dynamic memory, Memory address mapping and cache memory mapping techniques, Memory Hierarchy, Virtual memory and memory management unit
UNIT 5	I/O subsystems: Input-Output devices such as Disk, CD-ROM, Printer etc. Interfacing with IO devices, keyboard and display interfaces; Basic concepts Bus Control, Read Write operations, Programmed IO, Concept of handshaking, Polled and Interrupt-driven I/O, DMA data transfer.

Course Outcome (CO):

At the ends of this course studentswill have:

CO1: Awareness of computer organization.
CO2: Design and architecture of machine.
CO3:Implement different system calls for various units.
CO4: Logical representation of storage, representation and management.
CO5: Analysis of I/O subsystem.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H											M	H		
CO2			H		M									M	
CO3				M					M				L		
CO4				H						M			M		L
CO5				H						M					L

H = Highly Related; M = Medium L = Low

Text Book:

1. Computer Organization by V. Carl Hamacher, Safwat G. Zaky and Zvonko G. Vranesic , McGraw-Hill series(2002)

Reference Books:

1. Computer Organization and Design, by David Patterson and John Hennessey," Elsevier. 2008.

2. Computer System Architecture by Mano, M.M.,, Prentice Hall of India, New Delhi, 1992

3. Computer Systems Design and Architecture (2nd Edition) by Vincent P. Heuring and Harry F. Jordan (Dec 6, 2003)

4. Computer Architecture and Organization, by Hayes, J.P.1998, McGraw-Hill

.BCO 010C	DATABASE MANAGEMENT SYSTEMS	4-0-0 [4]

OJECTIVE:

- To provide knowledge of relational model
- To learn about ER diagrams.
- To learn about Query Processing and Transaction Processing

UNIT 1	Introduction - Database Systems versus File Systems, View of Data, Data Models, database languages, Database Users and Administrators. Transaction Management, Components of a Database management System. Entity-Relationship Model – Basic Concepts, Constraints, Keys, Design Issues, E-R Diagrams.
UNIT 2	Relational Model- Structures of relational databases, Integrity Constraints, Logical database Design, Tables, Views, Data Dictionary. Relational Algebra, Relational Calculus. SQL – Basic Structures, Query Handling,Triggers,Nested SQL Query, Embedded SQL,
UNIT 3	Relational Database Design- Functional Dependencies, Multi-valued Dependencies, Normal Forms, Decomposition into Normalized Relations.
UNIT 4	Fundamental Concepts of Transaction Management, ACID property. Serializability and testing for serializability, concurrency control schemes, lock-based protocols, two-phase locking protocols, graph-based protocols, time stamp-based protocols, deadlocks.
UNIT 5	File System: File organization- Heap File, Sequential File, Hash File, Clustered file, file operations, indexing, B-tree, B+ tree, Introduction to Data Mining, Data Farming, Data Warehousing

Course Outcome (CO):

At the ends of this course students will have:

CO1: Awareness of database management basics and different models that we use for database.
CO2: Design and architecture of relational model, relational algebra and SQL queries.
CO3: Implement different form of normalization.
CO4: Logical representation of internet database.
CO5: Analysis and concepts of transaction, concurrency and recovery systems.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H												H	M	
CO2			H		M				M						L
CO3				H		M							M		
CO4				M								M		L	L
CO5	M	L		H					L				M		

H = Highly Related; M = Medium L = Low

Text Books:

1. Database Systems Concepts – Korthe, TMH
2. An Introduction to Database Design – Date

Reference Books:

1. Fundamentals of Database Systems – Elmasri and Navathe
2. Database Management and Design – Hansen and Hansen .
3. Object-Oriented Database Design – Harrington

Samatrix/spl	**MACHINE LEARNING AND PATTERN RECOGNITION**	**3-0-0 [3]**

Software: Python, NumPy, Pandas, Matplotlib,Seaborn, SciPy, Scikit-Learn
Objectives: *The objective of this course is to teach students the basic concepts of machine learning,supervised learning, unsupervised learning, and reinforcement learning*

UNIT – I
Introduction: Learning systems, real world applications of machine learning, why machine learning, variable types and terminology, function approximation
Types of machine learning: Supervised learning, unsupervised learning, reinforcement learning
Important concepts of machine learning: Parametric vs non-parametric models, the trade-off between prediction accuracy and model interpretability, the curse of dimensionality, measuring the quality of fit, bias-variance trade off, overfitting, model selection, no free lunch theorem

UNIT – II
Linear Regression: Linear regression, estimating the coefficients, accessing the accuracy of coefficient estimates, accessing the accuracy of the model, multiple linear regression, qualitative Predictors Classification: Logistic regression, estimating regression coefficients, making predictions, multiple logistic regressions, linear discriminant analysis, bayes' theorem of classification, LDA for p=1, LDA for p>1, quadratic discriminant analysis

UNIT – III
Resampling Methods, Model Selection and Regularization: Cross-validation, leave-one-out crossvalidation,k-fold cross-validation, the bootstrap, subset selection, shrinkage methods, ridge and lasso regression, dimension reduction methods, principal components regression, partial least square

Tree Based Methods: Advantages and disadvantages of trees, regression Trees, classification trees, bagging, random forest, boosting

UNIT – IV
Support Vector Machine: Maximum margin classifier, classification using a separating hyperplane,the maximal margin classifier, support vector classifier, support vector machines, classification with non-linear decision boundaries, support vector machine, one-versus-one classification, one-versusmany classification
Unsupervised Learning: Principle component analysis, what are principal components, clustering methods, k-means clustering, hierarchical clustering, Independent component analysis, latent semantic indexing, Markov Models, Hidden Markov Models

BCO 012A	SOFTWARE PROJECT MANAGEMENT	3-0-0 [3]

OBJECTIVE:

- To understand Software Project Management Concepts.
- To understand Risk Analysis.
- To Study about Software Quality Management.
- To understand Project Evaluation.

UNIT 1	**Project Management:** Definition of the Project, Project Specification and parameters, Principles of Project Management, Project Management Life Cycle.
UNIT 2	**Software Project Planning**, Project Sequencing and Scheduling Activities, Scheduling resources, Network Planning, Work Breakdown Structure, Activity Resource Requirements, Project Management Plan , Critical path analysis PERT & CPM.
UNIT 3	**Project Scheduling and Tracking Techniques**: Why projects are delayed? Effort Estimation Techniques, Task Network and Scheduling Methods, Monitoring and Control Progress, Graphical Reporting Tools. **Monitoring & Control** : Change Control, Software Configuration Management (SCM)
UNIT 4	**Risk Analysis and Management**: Risk Mitigation and Management, Software Metrics and Project Management
UNIT 5	**Quality Management and People Management**- Introduction, Understanding Behaviour, Organizational Behaviour, Selecting The Right Person For The Job, Motivation, The Old man – Hackman Job Characteristics Model , Working in Groups, Organization and team structures, Decision Making, Leadership, Organizational Structures, Stress, Health And Safety.

Course Outcome:

CO1: Understand and practice the process of project management and its application in delivering successful IT projects;

CO2: Evaluate a project to develop the scope of work, provide accurate cost estimates and to plan the various activities

CO3: Understand and use risk management analysis techniques that identify the factors that put a project at risk and to quantify the likely effect of risk on project timescales

CO4: Identify the resources required for a project and to produce a work plan and resource Schedule

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H										M		M		
CO2		M			M	L								L	M
CO3							M		M				L		
CO4			L					M	L		L		L		M

H = Highly Related; M = Medium L = Low

Text Books:

1.Bob Hughes, Mike Cotterell, "Software Project Management", Tata McGraw Hill. (2009)

Reference Books:

1. Royce, "Software Project Management", Pearson Education. (2005).
2. Robert K. Wysocki, "Effective Software Project Management", Wiley.(2006)

BCO 013A	DATABASE MANAGEMENT SYSTEMS LAB	0-0-2 [2]

List of Experiments

1	Installation of MySQL
2	Analyze the problem and come with the entities in it. Identify what Data has to be persisted in the databases.
3	Represent all entities in a tabular fashion. Represent all relationships in a tabular fashion.
4	Creating of Tables on given problem
5	Applying Not Null, Check, Unique Constraints on database Tables.
6	Applying Primary Key, References, Foreign Key Constraints on database Tables.
7	Applying Insert, Select, Distinct Clause, Where Clause on database Tables.
8	Applying Update, Delete, Drop, on database Tables.
9	Applying table creation with select, Insert data using select, Renaming on database Tables.
10	Practice Queries using MINUS, UNION, INTERSECT, % operator.
11	Practice Queries using Group Functions.
12	Practice Queries using Group By, Having, Order By Functions.
13	Practice Queries using Arithmetic Operators, Comparison Operator.
14	Practice Queries using Logical Operator.
15	Practice Queries using any four String Functions.
16	Practice Queries using any four String Functions.
17	Practice Queries using Numeric Functions.
18	Practice Queries using Date Functions.

Course Outcome (CO):

At the ends of this course studentswill have:

CO1: Awareness of database management basics and different models that we use for database.
CO2: Design and architecture of relational model, relational algebra and SQL queries.
CO3: Implement different form of normalization.
CO4: Logical representation of internet database.
CO5: Analysis and concepts of transaction, concurrency and recovery systems.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H												H	M	
CO2			H		M				M						L
CO3				H		M							M		
CO4				M								M		L	L
CO5	M	L		H					L				M		

H = Highly Related; M = Medium L = Low

BCO 014A	OPERATING SYSTEMS LAB	0-0-2 [2]

List of Experiments

Experiment No	Aim
1	Write a C program to implement the various process scheduling mechanisms such as FCFS scheduling.
2	Write a C program to implement the various process scheduling mechanisms such as SJF Scheduling.
3	Write a C program to implement the various process scheduling mechanisms such as Round Robin Scheduling.
4	Write a C program to implement the various process scheduling mechanisms such as Priority Scheduling.
5	To implement deadlock avoidance & Prevention by using Banker's Algorithm.
6	To implement page replacement algorithms FIFO (First In First Out).
7	To implement page replacement algorithm LRU (Least Recently Used).
8	To implement page replacement algorithms Optimal (The page which is not used for longest time)
9	To implement the memory management policy- Paging.
10	To implement the memory management policy-segmentation.
11	Write a C Program to implement Sequential File Allocation method.
12	Write a C Program to implement Indexed File Allocation method.
13	Write a C Program to implement Linked File Allocation method.
14	Write a program to implement multi program variable task (MVT).
15	Write a program to implement multi program fixed task (MFT).

Course Outcome (CO):

At the ends of this course studentswill have:

CO1: Classify Unix Kernel mode with user mode & contrast between Kernel structures.
CO2: Identify and estimate process management & thread management strategies along with their different operations
CO3:Implement different system calls for various file handling operations.

CO4:determine paging and Caching techniques related to Virtual Memory.
CO5: construct shell scripts.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H			M				L			L		H		L
CO2		M	L						M					M	
CO3			M		M									M	M
CO4	M		L									L			

H = Highly Related; M = Medium L = Low

BCO 015A	COMPUTERGRAPHICS LAB	0-0-2 [2]

List of Experiments

1 Write a Program to Show basic Transformation with OpenGL
2 Write a Menu Driven Program with OpenGL
3 Write a Program to draw a line using Bresenham's Algorithm with OpenGL
4 Write a Program to implement midpoint algorithm to draw circle
5 Write a Program to implement midpoint algorithm to draw ellipse
6 Program to implement 2d scaling about an arbitrary axis.
7 Write a program to implement DDA line Algorithm
8 Program to implement 2d rotation about an arbitrary axis.
9 Program to implement translation of a line and triangle.
10 Program to implement Cohen Sutherland line clipping.
11 Program to implement Sutherland Hodgeman polygon clipping.
12 Program to draw Bezier curve.
13 Program to draw b-spline curve.
14 Program to implement a line using slope intercept formula.
15 Write a program to implement Bresenham 's Algorithm

Course Outcome (CO):

At the ends of this course studentswill have:

CO1: Understand the structure of modern computer graphics system
CO2: Understand the basic principles of implementing computer graphics primitives.
CO3: Familiarity with key algorithms for modeling and rendering graphical data
CO4: Develop design and problem solving skills with application to computer graphics

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H			M	L								M		
CO2			M	L					L		L			L	
CO3			L		L						M	L	L		M
CO4			H							L					L

H = Highly Related; M = Medium L = Low

B.Tech CSE Semester V

Samatrix/spl	**SCALA FOR DATA SCIENCE**	**2-0-0 [2]**

Software: Scala

Objectives: The objective of this course is to teach students Scala language, basic functions Scala and critical techniques

UNIT – I

Scala Language: Getting to know Scala programming language, Scala and Java, Statically typed language, Apache Spark and Scala, Scala Performance Benefits, Installing Scala, Using Scala REPL/Shell, getting help from Scala shell, Hello World, Paste mode, retrieving history, auto-complete feature, exiting from Scala REPL

UNIT – II

Variables, Data Types, Conditional Statements: Immutability of variables, define mutable and immutable variables, mutability and type safety, Specifying types for variables, Scala Identifier rules, naming conventions, Scala data types, Boolean types, string type, multiline strings, string operations, string concatenation, string interpolation, length of string, splitting string, extracting part of string, index of character of strings, the ANY type, type casting, Boolean expressions, conditional statement in Scala, nested IF/ELSE statement, pattern matching,

UNIT – III

Code Blocks, Functions, Collections: Code Blocks in Scala, Why use functions in Scala, understanding functions in Scala, define and invoke a function, functions with multiple parameters, positional parameters, functions with no argument, single-line function, passing function as argument, anonymous function, Collections in Scala, Understanding List, list size, convert list to string, iterating over list, map function and collection, foreach, reduce operation, list equality, create set, indexing map, manipulating maps, understanding tuples, indexing tuples, mutable collections, nested collections

UNIT – IV

Loops, Packages, Classes and Exceptional Handling: For loop, While loop, Breaking Loop iteration, classes and objects in Scala, Create classes and objects, singleton objects, case classes, equality checks, classes and packages, avoid name space collusion, importing package, fundamental of exception handling, type inferences and exception handling, try, catch, finally, Scala built tool (SBT),Compile Scala applications,

BCO 017A	FORMAL LANGUAGES & AUTOMATION THEORY	3-1-0 [4]

Objective:

- To understand various Computing models like Finite State Machine, Pushdown Automata, and Turing Machine.
- To understand Decidability and Undesirability of various problems
- To construct pushdown automata and the equivalent context free grammars.
- To prove the equivalence of languages described by pushdown automata and context free grammars.
- To construct Turing machines and Post machines and prove the equivalence of languages described by Turing machines and Post machines.

UNIT 1	Basics of Strings and Alphabets, Finite Automata – DFA, transition graphs, regular languages, non-deterministic FA, equivalence of DFA and NDFA, Mealy and Moore Machine, minimization of Finite Automata,
UNIT 2	Regular grammars, regular expressions, equivalence between regular languages, properties of regular languages, pumping lemma. Relationship between DFA and Regular expression.
UNIT 3	Context Free Languages – Leftmost and rightmost derivation, parsing and ambiguity, ambiguity in grammar and languages, simplification of CFG, Normal forms
UNIT 4	Pushdown Automata – NDPDA, DPDA, context free languages and PDA,comparison of deterministic and non-deterministic versions, closure properties, pumping lemma for CFL,
UNIT 5	Turing Machines, variations, halting problem, PCP, Chomsky Hierarchy, Recursive and Recursive enumerable language, Rice Theorem.

Course Outcomes: At the end of the course, the student should be able to:

CO1:Understand and construct finite state machines and the equivalent regular expressions.
CO2:Prove the equivalence of languages described by finite state machines and regular expressions.
CO3:Construct pushdown automata and the equivalent context free grammars.
CO4:Prove the equivalence of languages described by pushdown automata and context free grammars.
CO5:Construct Turing machines and Post machines and prove the equivalence of languages described by Turing machines and Post machines

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H		L										H		
CO2		H											L		
CO3	H		H												M
CO4		H		M											
CO5	H											H		L	

H = Highly Related; M = Medium L = Low

Text Books:

1. Hopcroft J.E., Motwani R. and Ullman J.D, "Introduction to Automata Theory,Languages and Computations", Second Edition, Pearson Education, 2008.

Reference Book:

1. Mishra K L P and Chandrasekaran N, "Theory of Computer Science – Automata, Languages and Computation", Third Edition, Prentice Hall of India, 2004.
2. Harry R Lewis and Christos H Papadimitriou, "Elements of the Theory of Computation", Second Edition, Prentice Hall of India, Pearson Education, New Delhi, 2003.
3. Peter Linz, "An Introduction to Formal Language and Automata", Third Edition, Narosa Publishers, New Delhi, 2002.
4. Kamala Krithivasan and Rama. R, "Introduction to Formal Languages, Automata Theory and Computation", Pearson Education 2009.
5. John C Martin, "Introduction to Languages and the Theory of Computation", Third Edition, Tata McGraw Hill Publishing Company, New Delhi, 2007.

BCO 023A	DESIGN AND ANALYSIS OF ALGORITHMS	3-0-0 [3]

OBJECTIVES:

At the end of the course, the student should be able to:

- Design effective, efficient, elegant, and readable algorithms for various classes of computing problems
- Determine space and time complexity of algorithms by the use various algorithm design techniques like (divide and conquer, backtracking, greedy, etc.)

UNIT 1	Introduction, algorithms specification, time and space complexity, performance analysis, recurrence relations. Divide and Conquer – finding max min.
UNIT 2	Dynamic Programming and Greedy Methods – Huffman tree construction, Knapsack problem, 0/1 Knapsack problem,least common subsequence**,** matrix chain multiplication. Backtrack: 4-queen problem, Branch and Bound: assignment problem
UNIT 3	Graph algorithms–flow problems, String Matching Algorithms: Naive algorithm, automata and KMP matcher algorithms, Boyer-Moore algorithm
UNIT 4	Number Theory Problems – CRT, GCD algorithms, modular arithmetic, Lower Bound Theory; Approximate Algorithms – Set cover, vertex cover, .Randomized Algorithms – Las Vegas and Monte Carlo methods
UNIT 5	NP Completeness: Definitions of P, NP-Hard and NP-Complete Problems. Decision Problems**..**

OUTCOMES: After study of this subject student will be able to know

CO1: Various methods of calculating complexity

CO 2: Finding out thebest method for different algorithms

CO3: About computational geometry, like Lower bound theory, modular arithmetic and CRT

CO4: Various Decision Problems like NP Complete, NP hard

CO5: Knowledge of Graph and its algorithm

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	H	M		M			M				L	H	M	
CO2	H	H			H							L	H	M	M
CO3	M	H	L		M			M				M			M

CO4	H	L	M		M							L	M	H	
CO5	H	M	M		M			L				L	M	M	L

Textbooks:

1. Cormen, Leizerson&Rivest, Introduction to algorithms, Prentice-Hall. 2002
2. Horowitz &Sahni, Fundamentals of Computer Algorithms, Galgotia Publication. 1999

Reference Books:

1. Aho, HopCroft, Ullman, The Design and Analysis of Computer Algorithms, Addison-Wesley. 2001.
2. Introduction to Design and Analysis of Algorithms, Anny Levitin, Person Education Press. 2007.
3. Gilles Brassard & Paul Bratley, Fundamental Algorithms, Prentice-Hall. 1998

BCO 019A	ARTIFICIAL INTELLIGENCE	4-0-0 [4]

Objective:

- To explain the basic principles of artificial intelligence;
- To apply logic and structured concepts in knowledge representation;
- To discuss the applications of artificial intelligence;
- To implement heuristic search algorithms;
- To design a natural language processor and implement a simple expert system.

UNIT 1	Introduction- What is intelligence? Foundations of artificial intelligence (AI), Task of artificial intelligence, Techniques of artificial intelligence, Problem Solving- Formulating problems, problem types, states and operators, state space, Expert system and its components.
UNIT 2	*Uni*nformed Search Strategies- Breath First Search, Depth First Search, Depth Limited Search, Informed Search Strategies- Heuristic Functions, Best First Search, Hill Climbing Algorithm, Problems and solutions of Hill Climbing, Iterative Deepening (IDA), A* algorithm, AO* Algorithm.
UNIT 3	Game playing- Introduction, Types of games, Minimax game algorithm, Alpha Beta cut-off procedure. Knowledge Representation- Role of Knowledge, Declarative Knowledge, Procedural Knowledge, Knowledge representation.
UNIT 4	Logics- propositional logics, First Order Predicate Logics (FOPL), Syntax of First Order Predicate Logics, Properties of Wff, Clausal Forms, Conversion to clausal forms.
UNIT 5	Planning- Introduction, Basic representation of plans, partial order planning, planning in the blocks world, Goal Stack Planning, Non-linear planning using constraint posting (TWEAK method).

Outcomes:

Upon the end of this course, student will be able to:

CO1 : Familiar basic principles of artificial intelligence;
CO2 : Able to use logic and structured concepts in knowledge representation;
CO3 : To discuss the applications of artificial intelligence;
CO4 : To implement heuristic search algorithms;
CO5 :To design a natural language processor and implement a simple expert system.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specific Outcome		
	PO 1	PO2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
CO1	H	L		H	H	L	M	M	M	M	L	H	L	M	
CO2	H	H		H	H	L	M	M	M	M	M	H		H	
CO3	H	M	M	H	H	L	M	H	M	M	L	H		H	
CO4	H	H	H	H	H	L	M	M	M	M	M	H	H	M	H
CO5	H	H	H	H	H	L	M	H	M	M	L	H	H	M	M

H = Highly Related; M = Medium L = Low

Text Books:

1.Stuart Russell and Peter Norvig. *Artificial Intelligence – A Modern Approach,* Pearson Education Press, 2001.

2.Kevin Knight, Elaine Rich, B. Nair, *Artificial Intelligence, McGraw Hill, 2008.*

Reference Books:

1.George F. Luger, *Artificial Intelligence,* Pearson Education, 2001.

2.Nils J. Nilsson, *Artificial Intelligence: A New Synthesis,* Morgan Kauffman, 2002.

BCO 035A	Programming in Java	4:0:0 [4]

Objective

- Cover issues related to the definition, creation and usage of classes, objects and methods.
- Discuss the principles of inheritance and polymorphism and demonstrate though problem analysis assignments how they relate to the design of methods, abstract classes and interfaces.
- Provide the foundation of good programming skills by discussing keys issues to the design of object-oriented software, including programming design patterns, automatic documentation techniques and programming testing.
- Cover the basics of creating APIs as well as allow students to explore the Java Abstract Programming Interface (API) and Java Collection Framework through programming assignments.
- Discuss basic principles and tools of collaborating programming (versioning systems, code review) and study their usage through group programming projects.

UNIT 1	**Java Fundamentals:** Features of Java ,OOPs concepts , Java virtual machine , Reflection byte codes ,Byte code interpretation , Data types, variable, arrays, expressions, operators, and control structures , Objects and classes
UNIT 2	**Java Classes:** Abstract classes ,Static classes ,Inner classes ,Packages,Wrapper classes Interfaces ,This ,Super ,Access control
UNIT 3	**Exception handling:** Exception as objects ,Exception hierarchy ,Try catch finally ,Throw, throws
UNIT 4	**IO package:** Input streams ,Output streams ,Object serialization ,De serialization ,Sample programs on IO files ,Filter and pipe streams
UNIT 5	**Multi threading:** Thread Life cycle ,Multi threading advantages and issues ,Simple thread program ,Thread synchronization .GUI: Introduction to AWT programming, Layout and component managers ,Event handling ,Applet class ,Applet life-cycle ,Passing parameters embedding in HTML ,Swing components – JApplet, JButton, JFrame, etc. Sample swing programs

Outcome:

At the end of this course student will:

Course Outcome: At the end of this course student will:

CO1:Understand how object-oriented concepts are incorporated into the Java programming language

CO2: Develop problem-solving and programming skills using OOP concept

CO3:Understand the benefits of a well structured program

CO4:Develop the ability to solve real-world problems through software development in high-level programming language like Java

CO5:Develop efficient Java applets,threading and applications using OOP concept

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcomes	Program OutComes												Program Specific Outcomes		
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1		M		M	H				M		H		M	H	
CO2	L		M		H		L	L		M		M		H	M
CO3		M		H	M	L		L		M	H		M	H	
CO4			H	M			L		M		H		M	H	
CO5			H	M		L						M	H	H	

- H = Highly Related; M = Medium L = Low

References:

1. Programming with Java A Primer, E.Balaguruswamy Tata McGraw Hill Companies

2. Java Programming John P. Flynt Thomson 2nd

3. Java Programming Language Ken Arnold Pearson

4. The complete reference JAVA2, Herbert schildt. TMH

Samatrix/spl	Big Data Analytics with Spark	2-0-0[2]

Software Req: Spark

Objectives: The objective of this course is to teach students Spark unified data processing platform,how to run Spark in Spark Shell or Databricks, RDD, Deal with structured data and Spark StructureStreaming

UNIT – I

Apache Spark and Installation: Big Data and Distributing computing at Google, Yahoo, Spark at AMPLab, What is ApacheSpark, Speed, Ease of Use, Modularity, extensibility of Spark, Unified Analytics, Spark SQL, Spark Machine Learning Library (MLLIB), Spark structured streaming, Apache Spark Distributed Execution and concepts, Distributed data and partitions, Who uses Spark for What, Data Science Task, Download Spark, Spark directory and files, Spark application concepts, spark session, Spark UI, build stand-alone application in Spark,

UNIT – II

Resilient Distributed Datasets (RDD) and SQL DataFrames: Introduction to RDD, RDD Operations, Creating RDDs, Transformations, map, flatMap, filter, union, intersection, substract, distinct,sample, Actions, Working with key/value pair RDD, Data Shuffling, Spark SQL, SQL Tables and Views,unmanaged and managed tables, create SQL database and tables, create view, reading tables into DataFrame, DataFrameReader, DataFrameWriter, Parquet, JSON, reading JSON file into DataFrame,reading CSV file, reading Avro, ORC, Image file,

UNIT – III

Spark Streaming: Evolution of Apache Spark Stream Processing Engine, Micro-batch stream processing, DStreams, philosophy of structured streaming, programming model, Stream DataSource and sink, structured streaming application, streaming DataFrame Operations, joining two streaming DataFrames, working with socket Data Source, Rate Data Source, File Data Source, Kafka Data Source, Custom Data Source, Working with Data Sinks, Kafka Data Sinks, Foreach Data Sinks, Console Data Sinks, Memory Data Sinks, Output modes and Triggers

UNIT – IV

Machine Learning with Spark: Spark machine learning library, machine learning pipelines, transformers, Binarizer transformer, Bucketizer transformer, One Hot Encoder transformer, tokenizer transformer, StopWordsRemover transformer, Estimators, String Indexer estimator, OneHotEncoderEstimator, Word2Vec Estimator, StandardScaler Estimator, Pipeline, ML Persistence,Model Tuning, Model Hyperparameter, Cross Validator, TrainValidationSplit

BCO 068A	Programming in Java Lab	0-0-2

LIST OF EXPERIMENTS

1. **Operators and Expressions**
   a. To write a java program to find the area of rectangle
   b. To write a java program to find the result of the following expressions
      i) (a<<2) + (b>>2)
      ii) (b>0)
      iii) (a+b*100) /10
      iv) a & b
      Assume a=10 and b=5
   c. To write a java program to print the individual digits of a 3 digit number.

2. Decision Making Statements

a. write a java program to read two integers and print the larger number followed by the words "is larger "If the numbers are equal print the message "These numbers are equal".

b. To write a java program to read an integer and find whether the number is odd or eve

c.To write a java program find the biggest of three integers.

3. Looping Statements

a. To write a java program to find the sum of digits of a given number

b. To write a java program to find the first 15 terms of Fibonacci sequence.

c. To write a java program to print the Armstrong numbers.

4. Array

a. To write a java program to find the largest and smallest number in an array.

5. Strings

a. To write a java program that creates a string object and initializes it with your name and performs the following operations

i) To find the length of the string object using appropriate String method.

ii) To find whether the character 'a' is present in the string. If yes find the number of times 'a' appear in the name and the location where it appears

6. String Buffer

a. To write a java program to create a StringBuffer object and illustrate how to append characters and to display the capacity and length of the string buffer

b. To write a java program to create a StringBuffer object and illustrate how to insert characters at the beginning

c. To write a java program to Create a StringBuffer object and illustrate the operations of the append () and reverse () methods.

7. Classes and Objects

a. To write a java program to display total marks of 5 students using student class. Given the following attributes: Regno(int), Name(string), Marks in subjects(Integer Array), Total (int).

b. To write a program in java with a class Rectangle with the data fields width, length, area and colour. The length, width and area are of double type and colour is of string type.The methods are get_length(), get_width(), get_colour() and find_area().

Create two objects of Rectangle and compare their area and colour. If the area and colour both are the same for the objects then display "Matching Rectangles", otherwise display "Non-matching Rectangle".

8. Inheritance

a. write a java program to create a Player class and inherit three classes Cricket_Player,

Football_Palyer and Hockey_Player.

9. Interfaces

a. To write a java program to show how a class implements two interfaces.

b. To write a java program to show that the variables in an interface are implicitly static and final and methods are automatically public

10. Packages

a. To write a java program to create a package for Book details giving Book name, Author name, price and year of publishing.

11. Applets & AWT

a. To write a java applet program to change the color of a rectangle using scroll bars to change the value of red, green and blue

b. To write an applet program for creating a simple calculator to perform Addition, subtraction, Multiplication and Division using Button, Label and TextField component.

12. Exception Handling

a. To write a java program to catch more than two exception

b. To write a java program to create our exception subclass that throws exception if the sum of two integers is greater than 99.

13. Multithreading

a. To write a java program for generating two threads, one for generating even number and one for generating odd number.

Course Outcome:

At the end of this course student will:

CO1:Understand how object-oriented concepts are incorporated into the Java programming language

CO2: Develop problem-solving and programming skills using OOP concept

CO3:Understand the benefits of a well structured program

CO4:Develop the ability to solve real-world problems through software development in high-level programming language like Java

CO5:Develop efficient Java applets,threading and applications using OOP concept

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcomes	Program OutComes												Program Specific Outcomes		
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1		M		M	H				M		H		M	H	
CO2	L		M		H		L	L		M		M		H	M
CO3		M		H	M	L		L		M	H		M	H	
CO4			H	M			L		M		H		M	H	
CO5			H	M		L						M	H	H	

H = Highly Related; M = Medium L = Low

BCO 025A	DESIGN& ANALYSIS OF ALGORITHMS LAB	0-0-2

List of Experiments

1. Write a Program to Explore a Binary Heap
2. Write a Program for Merging of two search trees
3. Write a program to implement Huffman tree construction
4. Write a Program for Computing a spanning tree having smallest value of largest edge
5. Write a Program for Finding the decimal dominant in linear time
6. Write a Program for Problems on Graphs. Etc.
7. Write a program to find Greatest Common Divisor
8. Write a program for fractional Knapsack problem
9. Write a program for 0/1 Knapsack problem
10. Write a program to implement Naive algorithm,
11. Write a program to implement KMP matcher algorithms,

12. Write a program to implement Boyer-Moore algorithm

13. Write a program to implement modular arithmetic

14. Write a program to implement Set cover,

15. Write a program to implement vertex cover

OUTCOMES: After study of this subject student will be able to know
CO1: Various methods of calculating complexity
CO 2: Finding out the best method for different algorithms
CO3: About computational geometry, like Lower bound theory, modular arithmetic and CRT
CO4: Various Decision Problems like NP Complete, NP hard
CO5: Knowledge of Graph and its algorithm

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	H	M		M			M				L	H	M	
CO2	H	H			H							L	H	M	M
CO3	M	H	L		M			M				M			M
CO4	H	L	M		M							L	M	H	
CO5	H	M	M		M			L				L	M	M	L

B.Tech CSE Semester VI

Samatrix/spl	**DATA SCIENCE WITH HADOOP**	**3-0-0 [3]**

Software: Apache Hadoop, Apache Pig, Apache Hive, Apache Spark, Apache Avro, Ubuntu/Centos, Java

Objectives: The objective of this course is to teach students the conceptual framework of Big Data, Virtualization, MapReduce, HDFS, Pig, Hive, Spark, ZooKeeper, HBase

UNIT – I

Big Data: Fundamentals of Big Data, defining big data, building successful big data management architecture, big data journey

Big Data Types: Structured and unstructured data types, real time and non-real time requirements

Distributed Computing: History of distributed computing, basics of distributed computing

UNIT – II

Big Data Technology Foundation: Big Data stack, redundant physical infrastructure, security infrastructure, operational databases, organising data services and tools, analytical data warehouse, big data analytics

Virtualization: Basics of virtualization, hypervisor, abstraction and virtualization, implementing virtualization with big data

Cloud and Big Data: Defining cloud, cloud deployment and delivery models, cloud as an imperative for big data, use the cloud for big data

UNIT – III

Operational Databases: Relational database, nonrelational database, key-value pair databases, document databases, columnar databases, graph databases, spatial databases

MapReduce Fundamentals: Origin of MapReduce, map function, reduce function, putting map and reduce together, optimizing map reduce

Hadoop: Discovering Hadoop, Hadoop distributed file system, Hadoop MapReduce, Hadoop file system, dataflow, Hadoop I/O, data integrity, compression, serialization, file-based data structure

UNIT – IV

Avro: Avro data types and schemas, in-memory serialization and deserialization, avro datafiles, schema resolution

Pig: Comparison with databases, pig latin, user defined functions, data processing operators

Hive: Running hive, comparison with traditional databases, HiveQL, tables, querying data, user defined functions

Spark: Resilient distributed datasets, shared variables, anatomy of a spark job run, executors and cluster managers,

HBase: HBasics, concepts, clients, HBase vs RDBMS, Praxis

ZooKeeper: ZooKeeper services, building application with ZooKeeper

BCO 028A	COMPILER CONSTRUCTION	3-0-0 [3]

OBJECTIVE: At the end of the course, the student should be able to:

- Apply the principles in the theory of computation to the various stages in the design of compilers;
- Explain the stages involved in the translation process;
- Analyse problems related to the stages in the translation process;
- Design a compiler for a simple programming language; and
- Implement a compiler based on its design.

UNIT 1	Overview of compilation- The structure of a compiler and applications of compiler technology; Lexical analysis - The role of a lexical analyzer, specification of tokens, recognition of tokens, hand-written lexical analyzers, LEX, examples of LEX programs. Introduction to syntax analysis -Role of a parser, use of context-free grammars (CFG) in the specification of the syntax of programming languages, techniques for writing grammars for programming languages (removal left recursion, etc.), non- context-free constructs in programming languages, parse trees and ambiguity, examples of programming language grammars.
UNIT 2	Top-down parsing- FIRST & FOLLOW sets, LL(1) conditions, predictive parsing, recursive descent parsing, error recovery. LR-parsing - Handle pruning, shift-reduce parsing, viable prefixes, valid items, LR(0) automaton, LR-parsing algorithm, SLR(1), LR(1), and LALR(1) parsing. YACC, error recovery with YACC and examples of YACC specifications.
UNIT 3	Syntax-directed definitions (attribute grammars)-Synthesized and inherited attributes, examples of SDDs, evaluation orders for attributes of an SDD, Dependency graphs-attributed and L-attributed SDDs and their implementation using LR-parsers and Recursive Descent parsers respectively.
UNIT 4	Semantic analysis- Symbol tables and their data structures. Representation of "scope". Semantic analysis of expressions, assignment, and control-flow statements, declarations of variables and functions, function calls, etc., using S- and L-attributed SDDs (treatment of arrays and structures included). Semantic error recovery.
UNIT 5	Intermediate code generation - Different intermediate representations – quadruples, triples, trees, flow graphs, SSA forms, and their uses. Translation of expressions (including array references with subscripts) and assignment statements. Translation of control-flow statements – it- then-else, while-do, and switch. Short-circuit code and control-flow translation of Boolean expressions. Back patching. Examples to illustrate intermediate code generation for all constructs. Run-time environments:- Stack allocation of space and activation records. Access to non-local data on the stack in the case of procedures with and without nesting of procedures.

Course Outcome

At the end of this course students will have:

CO1 To apply the knowledge of lex tool & yacc tool to devleop a scanner & parser.
CO2 To design parser and Intermediate Code Generation in compiler.
CO3 To deal with different translators.
CO4 To learn the new code optimization techniques to improve the performance of a program in terms of speed & space.
CO5 To use the knowledge of patterns, tokens & regular expressions for solving a problem.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specific Outcome		
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1	H	H	L		H			L					H	M	
CO2		H				L							H		
CO3		L		H	L										
CO4		H					H							H	H
CO5		H		L		H									

H = Highly Related; M = Medium L = Low

Text Books:

1. Compilers: Principles, Techniques, and Tools, by A.V. Aho, Monica Lam, Ravi Sethi, and J.D. Ullman, (2nd ed.), Addison-Wesley, 2007 (main text book, referred to as ALSU in lab assignments).
2. K.D. Cooper, and Linda Torczon, Engineering a Compiler, Morgan Kaufmann, 2004.

Reference Books:

1. K.C. Louden, Compiler Construction: Principles and Practice, Cengage Learning, 1997.
2. D. Brown, J. Levine, and T. Mason, LEX and YACC, O‟Reilly Media, 1992.

BCO 024A	ADVANCED COMPUTER ARCHITECTURE	3-0-0 [3]

OBJECTIVES:-

- Tocover the underlying concepts and techniques used in Advance Computer Architecture.
- To discusses principles of parallel algorithmsdesign and different parallel programming models
- To have general idea of Computer Organization. In addition, a familiaritywith Memory organization, Computational models required.

UNIT 1	Introduction - What is computer architecture? Software-hardware interface. Performance andPower.Performancemetrics.Performancemeasurement.Benchmarkprograms.
UNIT 2	Instructions- Instruction Set. Operations. Operands and addressing modes.Roleof compilers and system software.Understanding implementation of function calls andreturns, array references, pointers.
UNIT 3	Computer Arithmetic- Signed integers. Floating point.Rounding and accuracy.Addition and Subtraction.Multiplication.Division; Processor - Data path elements.Data pathcontrol.
UNIT 4:	Pipelining Speedup. Pipeline hazards.Stalling.Forwarding.Branchprediction.Exceptions.Speculation.Multiple issue.Dynamic scheduling; Cache memory- Locality ofreference.Cache organization and access. Multilevelcaches.Performance.Cache coherence.
UNIT 5	Virtual memory- Hardware support for address translation, page fault handling.Translation look aside buffer. Hardware-software interface.

OUTCOMES:-Upon completion of the syllabus the students will be able to know :

- Understand the concepts and techniques used in Advance Computer Architecture.
- Use of the principles of parallel algorithmsdesign and different parallel programming models.
- Familiar with general idea of Computer Organization. In addition, a familiarity with Memory organization, Computational models required

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcomes	Program OutComes												Program Specific Outcomes		
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1															
CO2															
CO3															
CO4															
CO5															

- H = Highly Related; M = Medium L = Low

Text Books:

1.David A. Patterson and John L. Hennessy, Computer Organization and Design: The Hardware and SoftwareInterface, Morgan Kaufmann Publishers, Fourth Edition.(2009)

Reference Books:

1.John L. Hennessy and David A. Patterson, Computer Architecture: A Quantitative Approach, MorganKaufmann Publishers (2007)

BCO 037A	ADVANCE PROGRAMMING IN JAVA	4-0-0 [4]

OBJECTIVES:-Students will be able to know the following

- To learn the Java programming language: its syntax, idioms, patterns, and styles.
- To become comfortable with object oriented programming: Learn to think in objects
- To learn the essentials of the Java class library, and learn how to learn about other parts of the library when you need them.
- To introduce event driven Graphical User Interface (GUI) programming

UNIT 1	Revisited of GUI, Database Programming using JDBC Introduction to JDBC ,JDBC Drivers & Architecture CURD operation Using JDBC Connecting to non-conventional Databases. Connectivity with SQL server, Oracle and MS access.
UNIT 2	Networking , Networking Basics ,The Networking Classes and Interfaces InetAddress ,Factory Methods ,Instance Methods ,Inet4Address and Inet6Address, TCP/IP Client Sockets ,URL,URLConnection,Http URL Connection, The URI Class,Cookies, TCP/IP Server Sockets,Datagram, DatagramSocket ,DatagramPacket,
UNIT 3	RMI (Remote Method Invocation) RMI overview RMI architecture, Designing RMI application, Executing RMI application. Example demonstrating RMI
UNIT 4	Servlet: Web Application Basics. Architecture and challenges of Web Application.Introduction to servlet life cycle Developing and Deploying Servlets Exploring Deployment Descriptor (web.xml). Handling Request and Response Initializing a Servlet Accessing Database Servlet Chaining Session Tracking & Management Dealing with cookies Transferring Request Accessing Web Context Passing INIT and CONTEXT Parameter Sharing information using scope object Controlling concurrent access User Authentication Filtering Request and Response Programming Filter Filter Mapping Servlet Listeners .
UNIT 5	Basic JSP Architecture Life Cycle of JSP (Translation, compilation) JSP Tags and Expressions Role of JSP in MVC-2 JSP with Database JSP Implicit Objects Tag Libraries JSP Expression Language (EL) Using Custom Tag JSP Capabilities: Exception Handling Session Management Directives JSP with Java Bean.

OUTCOMES:-

Upon end of this course, students will be able to:

- About the Java programming language: its syntax, idioms, patterns, and styles.

- Becomecomfortable with object oriented programming: Learn to think in objects
- Learnthe essentials of the Java class library, and learn how to learn about other parts of the library when you need them.
- Introduce event driven Graphical User Interface (GUI) programming

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcomes	Program Outcomes												Program specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1			H	L	H		L			M	H			H	
CO2		L	H		H	L		L					M	H	
CO3			H	M					L	L		M		H	
CO4				M	H			M			L			H	M
CO5		L			H	M			M					H	

Text Books:

1. J2EE: The complete Reference by James Keogh
2. Java 6 And J2Ee 1.5, Black Book by kogent
3. Java Server Programming Java EE6 (J2EE 1.6), Black Book by kogent

Reference books:-

1. Programming with Java A Primer, E.Balaguruswamy Tata McGraw Hill Companies
2. Java Programming John P. Flynt Thomson 2nd
3. Java Programming Language Ken Arnold Pearson

Samatrix/spl	**DEVOPS FOR WEB DEVELOPMENT**	**0:0:2 [2]**

Software Req: Jenkins, Git, Chef, Ansible, Docker, Kubernetes

Objectives: *The objective of this course is to teach students how to utilize business resources effectively to increase productivity and collaboration, use open source DevOps tools for Continuous Integration and Continuous Delivery (CI/CD)*

UNIT – I

DevOps Infrastructure: What is DevOps, Implement Continuous Integration (CI), Continuous Delivery (CD), and Continuous Delivery (CD), understand Infrastructure as Code (IaC) practices, Business drivers for DevOps adoption, data explosion, cloud computing, Big data, data science and machine learning, in-memory computing, planning DevOps strategy, benefits of DevOps

UNIT – II

DevOps Framework: DevOps process, Source code management, code review, configuration management, build management, Artifacts repository management, release management, test automation, continuous integration, continuous delivery, continuous deployment, routine automation, DevOps maturity life cycle, DevOps Maturity Map, DevOps progression framework, DevOps Maturity checklists, Agile framework

UNIT – III

DevOps – Continuous Integration, Delivery and Deployment: Best Practices for CI/CD, Jenkins setup, Git (SCM) integration with Jenkins, Integrating GitHub with Jenkins, Maven (Build) tool integration with Jenkins, Building Jobs with Jenkins, Source Code Review – Gerrit, Installation of Gerrit, Repository Management, Testing with Jenkins, Continuous Delivery – Build Pipeline, DevOps continuous Deployment, Chef landscape components, features of Chef, Chef Automate workflow, Features of Ansible, Ansible CMDB, Playbooks, Modules, Inventory, Plugins, Ansible Tower, Ansible Vault, Ansible Galaxy, Monitoring, Aplunk, Nagios Monitoring Tool,

UNIT – IV

Containerized Applications with Docker and Kubernetes: Installing Docker, Creating Dockerfile, Building and running a container on a local machine, pushing an image to Docker Hub, managing containers with Kubernetes, Technical requirements of Kubernetes, Kubernetes architecture overview, Installing Kubernetes on a local machine, Installing Kubernetes Dashboard, Using HELM as package manager, AKS,

BCO 031A	Compiler Design Lab	0:0:2 [2]

List Of Experiments

1 Familiarization with LEX by writing simple specifications for tokens such as identifiers, numbers, comments in C/C++, etc. All LEX specifications must be compiled andexecuted with appropriate inputs. At least ten such exercises must be completed in two labclasses.

2 LEX specification for tokens of the small language in ALSU‟s book

3 Complete the specifications in (2) above to make a complete lexical analyzer. (1 lab class)

4 Familiarization with YACC by writing simple specifications for desk calculator, variable declarations in C (only numbers and array). All YACC specifications must be compiled and executed with appropriate inputs. Note that this exercise also requires LEX specifications o the tokens involved. (2 lab classes)

5 YACC specifications for the syntax of the small language in ALSU‟s book (appendix A)(1 lab class)

6 Adding error recovery to (5) above to make a complete parser. (1 lab class)

7 S-attributed specification of the semantics of the small language in ALSU‟s book

8 Adding semantic error recovery to the semantic analyzer in (7) above to make a complete semantic analyzer. (1 lab class)

9 Intermediate code generation for the constructs of the small language in ALSU‟s book (appendix A) to be incorporated into the semantic analyzer of (8) above. Students doing this last assignment may be awarded bonus marks. (3 lab classes)

10 Write a programme to parse using Brute force technique of Top-down parsing.

11 Write a program for generating for various intermediate code forms
i) Three address code ii) Polish notation

12 Develop an operator precedence parser (Construct parse table also)

13 Develop a recursive descent parser

14 Develop a lexical analyser to recognize a few patterns.

BCO 069A	Advance Programming in Java Lab	0-0-2

Exp.No.	Name of the Experiment
1	Design a registration page using HTML.
2	**Implementing JDBC** Program 2(A)Write a program by using JDBC to execute insert, select and update query by using PreparedStatement and display the results. Program 2(B) Write a program by using JDBC to execute an update query by using PreparedStatement and display the results. Program 2(C) Write a program and execute ResultSetMetaData Interface by using JDBC.
3	**Implementing Servlet** Program 3(A) Write a program and execute a simple servlet demonstrating servlet lifecycle. Program 3(B) Write a program and execute a servlet program that receives input from html page. Program 3(C) Write a program and execute ServletRequest and ServletResponse Interfaces with methods. Program 3(D) Write a program and execute HttpServlet Class doGet() and doPost() Methods. Program 3(E) Write a program to store the user information into Cookies. Write another program to display the above stored information by retrieving from Cookies.
4	**Implementing JSP,JSP Custom Tags and Directives** Program4(A) Write a program to connect HTML page,JSP page and mysql database.Program 4(B) Write a program and implement custom tags in JSP Program 4(C) Write a program and implement JSP directives.
5	**Implementing JavaBean** Program 5 Write a program and implement Javabeans using JSP page.
6	**Implementing JSP Standard ActionElements** Program 6 Write a program and implement JSP StandardActionElements.
7	**Implementing JSP Scripting Elements** Program 7:Write a program and execute JSP Scriptlets,Declarations and Expressions.
8	**Learning session management** Program 8(A):Write program and execute session management using URL rewriting

Program 8(B) :Write program and execute session management using Hidden Fields.

Program 8(C): Write program and execute session management using Cookie

Program 8(D): Write a program and execute session management using Session Objects.

9 **Remote Method Invocation (RMI)**

Program 9(A):Write a program and execute Remote Method Invocation

10 **Configure web.xml**

Program 10:Write a code to deploy web.xml file

11 **Performing Client-Server Communication and Networking**

Program 11(A): WAP to implement Client-Server Program
Program 11(B): WAP to implement InetAddress.
Program 11(C): WAP for Sending Email in java

12. **Implementing Multithreading**

Program 12: WAP to implement multithreading(three threads using single run method).

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcomes	Program Outcomes												Program specific Outcomes		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1			H	L	H		L			M	H			H	
CO2		L	H		H	L		L					M	H	
CO3			H	M					L	L		M		H	
CO4				M	H			M			L			H	M
CO5		L			H	M			M					H	

12

B.Tech CSE Semester VII

Samatrix/spl	DATA VISUALIZATION	3-0-0 [3]

Software: Microsoft Office 2013 or 2016, Tableau Desktop, Power BI

UNIT - I

INTRODUCTION TO DATA HANDLING Overview of Data analysis, Introduction to Data visualization, Working with statistical formulas - Logical and financial functions , Data Validation & data models,Power Map for visualize data , Power BI-Business Intelligence , Data Analysis using statistical methods, Dashboard designing.

UNIT - II

INTRODUCTION TO DATA MANIPULATION USING FUNCTION: Heat Map, Tree Map, Smart Chart,Azure Machine learning , Column Chart, Line Chart , Pie,Bar, Area, Scatter Chart, Data Series, Axes ,Chart Sheet , Trendline , Error Bars, Sparklines, Combination Chart, Gauge, Thermometer Chart ,Gantt Chart , Pareto Chart etc , Frequency Distribution, Pivot Chart, Slicers , Tables: Structured References, Table Styles , What-If Analysis: Data Tables, Goal Seek, Quadratic Equation ,Transportation Problem, Maximum Flow Problem, Sensitivity Analysis, Histogram, Descriptive,Statistics, Anova, F-Test, t-Test, Moving, Average, Exponential Smoothing | Correlation model |Regression model, Practical Lab

UNIT - III

TABLEAU SOFTWARE: GETTING STARTED WITH TABLEAU SOFTWARE: What is Tableau? What does the Tableau product suite comprise of? How Does Tableau Work? Tableau Architecture, What is My Tableau Repository? Connecting to Data & Introduction to data source concepts, Understanding the Tableau workspace, Dimensions and Measures, Data Types & Default Properties, Building basic views, Saving and Sharing your work-overview, Practical Lab

UNIT - IV

TABLEAU: BUILDING VIEWS (REPORTS): Date Aggregations and Date parts, Cross tab & Tabular charts, Totals & Subtotals, Bar Charts & Stacked Bars, Trend lines, Forecasting, Filters, Context filters, Line Graphs with Date & Without Date, Tree maps, Scatter Plots

BCO 030A	PRINCIPLES OF INFORMATION SYSTEM SECURITY	3-0-0 [3]

Objective:

- To explain the objectives of information security
- To analyse the trade-offs inherent in security
- To describe the enhancements made to IPv4 by IPSec
- To understand the basic categories of threats to computers and networks
- To discuss issues for creating security policy for a large organization

UNIT 1	Information Security: Introduction, History of Information security, What is Security, CNSS Security Model, Components of Information System, Balancing Information Security and Access, Approaches to Information Security Implementation, The Security Systems Development Life Cycle.
UNIT 2	Cryptography: Concepts and Techniques, symmetric and asymmetric key cryptography, steganography, Symmetric key Ciphers: DES structure, DES Analysis, Security of DES, variants of DES, Block cipher modes of operation , AES structure, Analysis of AES , Key distribution Asymmetric key Ciphers: Principles of public key cryptosystems, RSA algorithm, Analysis of RSA, Diffie-Hellman Key exchange
UNIT 3	Message Authentication and Hash Functions: Authentication requirements and functions, MAC and Hash Funtions, MAC Algorithms: Secure Hash Algorithm, Whirlpool, HMAC, Digital signatures, X.509, Kerberos
UNIT 4	Security at layers(Network, Transport, Application): IPSec, Secure Socket Layer(SSL), Transport Layer Security(TLS), Secure Electronic Transaction(SET), Pretty Good Privacy(PGP), S/MIME
UNIT 5	Inruders, Virus and Firewalls: Intruders, Intrusion detection, password management, Virus and related threats, Countermeasures, Firewall design principles, Types of firewalls

OUTCOMES: At the end of the course, the student should be able to:

CO1: Explain the objectives of information security and analyze the importance of information Security in real world.
CO2: Analyse the trade-offs inherent in security and designing and analysis of different encryption Algorithms.
CO3: Implementation of MAC and Hash functions, security at different layers of a network
CO4: Understand the basic categories of threats to computers and networks and explore different types of intruders and viruses.
CO5: Discuss issues for creating security policy for a large organization

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome												Program Specifice Outcome		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	H					M					M			M
CO2	H	H	H	M								M		M	
CO3	H			L									H		
CO4	L				H						M			L	
CO5	M				H	H	H	M		L	M				M

H = Highly Related; M = Medium L=Low

Text Books –

1. Stalling Williams: Cryptography and Network Security: Principles and Practices, 4th Edition, Pearson Education, 2006.

2. Kaufman Charlie et.al; Network Security: Private Communication in a Public World, 2nd Ed., PHI/Pearson.

Reference Books:

1. Pieprzyk Josef and et.al; Fundamentals of Computer Security, Springer-Verlag, 2008.

2. Trappe & Washington, Introduction to Cryptography, 2nd Ed. Pearson.

BCO 029A	DATA MINING &WAREHOUSING	3:0:0 [3]

OBJECTIVE:

- To compare and contrast different conceptions of data mining.
- To explain the role of finding associations in commercial market basket data.
- To characterize the kinds of patterns that can be discovered by association rule mining.
- To describe how to extend a relational system to find patterns using association rules.
- To evaluate methodological issues underlying the effective application of data mining.

UNIT 1	Introduction: Basic concepts of data mining, including motivation and definition; different types of data repositories; data mining functionalities; concept of interesting patterns; data mining tasks; current trends, major issues and ethics in data mining
UNIT 2	Data: Types of data and data quality; Data Preprocessing: data cleaning, data integration and transformation, data reduction, discretization and concept hierarchy generation; Exploring Data: summary statistics, visualization, multidimensional data analysis
UNIT 3	Association and Correlation Analysis: Basic concepts: frequent patterns, association rules - support and confidence; Frequentitemset generation - Apriori algorithm, FP-Growth algorithm; Rule generation, Applications of Association rules; Correlation analysis.
UNIT 4	Clustering Algorithms and Cluster Analysis: Concept of clustering, measures of similarity, Clustering algorithms: Partitioning methods - k-means and k-medoids, CLARANS, Hierarchical methods - agglomerative and divisive clustering, BIRCH, Densitybased methods - Subspace clustering, DBSCAN; Graph-based clustering - MST clustering; Cluster evaluation; Outlier detection and analysis.
UNIT 5	Classification: Binary Classification - Basic concepts, Bayes theorem and Naïve Bayes classifier, Association based classification, Rule based classifiers, Nearest neighbor classifiers, Decision Trees, Random Forest; Perceptrons; Multi-category classification; Model overfitting, Evaluation of classifier performance - cross validation, ROC curves. Applications: Text mining, Web data analysis, Recommender systems.Prerequisites: Familiarity with basic Linear Algebra and Probability will be assumed.

OUTCOMES: : At the end of the course, the student should be able to:

- Compare and contrast different conceptions of data mining.
- Explain the role of finding associations in commercial market basket data.
- Characterize the kinds of patterns that can be discovered by association rule mining.
- Describe how to extend a relational system to find patterns using association rules.

- Evaluate methodological issues underlying the effective application of data mining.

MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcomes	Program OutComes												Program Specific Outcomes		
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
CO1															
CO2															
CO3															
CO4															
CO5															

H = Highly Related; M = Medium L = Low

Text Books:

1. Pang-Ning Tan, Michael Steinbach and Vipin Kumar, Introduction to Data Mining. Pearson (2005), India.ISBN 978-8131714720

2. Jiawei Han and MichelineKamber, Data Mining: Concepts and Techniques, Morgan Kaufmann, 3rd edition (July 2011) 744 pages. ISBN 978-0123814791

Reference Books:

1. T. Hastie, R. Tibshirani and J. H. Friedman, The Elements of Statistical Learning, Data Mining, Inference, andPrediction.Springer, 2nd Edition, 2009.768 pages. ISBN 978-0387848570

2. C. M. Bishop, Pattern Recognition and Machine Learning.Springer, 1st edition, 2006.738 pages. ISBN 978-0387310732

3. Ian H. Witten and Eibe Frank, Data Mining: Practical Machine Learning Tools and Techniques, MorganKaufmann, 3rd edition (January 2011).664 pages. ISBN 978-0123748560.