Bharati Vidyapeeth
(Deemed to be University)
Pune, India

College of Engineering, Pune

B.Tech. (Information Technology)

Program Curriculum (2021 Course)

BHARATI VIDYAPEETH
(DEEMED TO BE UNIVERSITY),
COLLEGE OF ENGINEERING, PUNE

Department of Information Technology

VISION OF UNIVERSITY:

Social Transformation Through Dynamic Education

MISSION OF UNIVERSITY:

- To make available quality education in different areas of knowledge to the students as per their choice and inclination.
- To offer education to the students in a conducive ambiance created by enriched infrastructure! and academic facilities in its campuses.
- To bring education within the reach of rural, tribal, and girl students by providing them substantive fee concessions and subsidized hostel and mess facilities.
- To make available quality education to the students of rural, tribal, and other deprived sections of the population.

VISION OF THE INSTITUTE

To be a world-class Institute for Social Transformation through Dynamic Education.

MISSION OF THE INSTITUTE

- To provide quality technical education with advanced equipment, qualified faculty members, infrastructure to meet the needs of the profession and society.
- To provide an environment conducive to innovation, creativity, research, and entrepreneurial leadership.
- To practice and promote professional ethics, transparency, and accountability for social community, economic and environmental conditions.

Program Vision

To be a leading Program transforming students into skilled IT professionals.

Program Mission

- Amplify the student's technical skills by conducting continuing education programs, organizing and participating in various technical events.
- Provide comprehensive support in synchronization with industry to achieve professional and technological excellence.
- Provide an environment for effective social and ethical skills.

Program Educational Objectives:

PEO1: Cultivate IT, graduates, for industry pertaining to Information Technology solutions.

PEO2: Practice technical competency and teamwork abilities.

PEO3: Exhibit social responsibilities by following ethical practices in graduates' professional pursuits.

Program Specific Outcomes

- **PSO 1:** Use knowledge of core and allied courses for developing a computer-based system to deliver a quality product for real-world problems of society.
- **PSO 2:** Apply Modern IT tools and techniques for perusing their professional career by practicing effective communication with team members.
- **PSO 3:** Develop time-bound, cost-effective, and sustainable solutions by following professional ethics.

Program Outcomes

1. Apply knowledge of mathematics and computer science to analyze computer-based information systems.
2. Apply logical and programming skills to identify, formulate and analyze for solving computational problems.
3. Examine complex problems by a diagnosis of available information to provide an appropriate conclusion.
4. Design applications with suitable consideration of societal needs.
5. Use functional skills of modern IT tools and techniques for modeling and implementation.
6. Play the role of a team player to accomplish a common goal.
7. Convey technological concepts through significant documentation and presentation skills.
8. Demonstrate professional conduct by following norms of the engineering practice.
9. Apply software engineering methodologies for sustainable development.
10. Follow ethical and legal practices related to the functioning of the IT industry.
11. Apply management skills and techniques for creating time-bound and Cost-effective projects.
12. Exhibit lifelong learning by upgrading to state-of-the-art IT practices and Technology.

COURSE COMPONENTS OF UNDERGRADUATE ENGINEERING PROGRAMME

Sr. No.	Category	Number of Courses
1	Basic Science Courses (BSC)	04
2	Engineering Science Course (ESC)	03
3	Core Course (CC)	38
4	Elective Course (EC)	02
5	Project (PROJ)	04
6	Internship (INT)	01
7	Vocational Course (VC)	04
8	Massive Open Online Course (MOOC)	03
9	Research Paper Publication (Research)	01
10	Social Activities (SA)	02
11	Mandatory Course (MC)	02
12	Internal Assessment (IA)	-
13	University Examination (UE)	-
TOTAL		**64**

CORRELATION BETWEEN GRADUATE ATTRIBUTES AND PROGRAMME OUTCOMES

Graduate Attributes/ Program Outcomes	a	b	c	d	e	f	g	h	i	j	k	l
Engineering Knowledge	✓											
Problem Analysis		✓										
Design/Development of solutions			✓									
Conduct Investigations of Complex Problems				✓								
Modern Tool Usage					✓							
The Engineer and Society						✓						
Environment and Sustainability							✓					
Ethics								✓				
Individual and teamwork									✓			
Communication										✓		
Project management and finance											✓	
Life-long learning												✓

CREDIT DISTRIBUTION TO COURSE COMPONENTS OF UNDERGRADUATE ENGINEERING PROGRAMME

Sr. No.	Category	Breakup of Credits
1	Basic Science Course (BSC)	19
2	Engineering Science Course (ESC)	15
3	Core Course (CC)	145
4	Elective Course (EC)	10
5	Project (PROJ)	18
6	Internship (INT)	03
7	Vocational Courses (VC)	08
8	Massive Open Online Course (MOOC)	06
9	Research Paper Publication (Research)	02
10	Social Activities (SA)	04
11	Mandatory Course (MC)	Non-Credit
TOTAL		**230**

Program: B.TECH. (Information Technology) Semester – I CBCS 2021 Course

Sr. No.	Course Code	Name of Course	Teaching Scheme (Hrs./Week)			Examination Scheme (Marks)						Credits			
			L	P	T	UE	IA	TW	TW & OR	TW & PR	Total	L	P TW/OR/PR	T	Total
1		Mathematics for Computing-I	3	-	1	60	40	-	-	-	100	3	-	1	4
2		Physics for Computing System	3	2	-	60	40	25	-	-	125	3	1	-	4
3		Computer Aided Drafting	4	2	-	60	40	50	-	-	150	4	1	-	5
4		Digital Electronics	4	2	-	60	40	25	-	-	125	4	1	-	5
5		Structured Programming	4	2	-	60	40	-	-	100	200	4	1		5
6		Computer System Workshop Technology	-	4	-	-	-	-	-	50	50	-	2	-	2
	Total		**18**	**12**	**1**	**300**	**200**	**100**	**-**	**150**	**750**	**18**	**6**	**1**	**25**

Program: B.TECH. (Information Technology) Semester – II CBCS 2021 Course

Sr. No.	Course Code	Name of Course	Teaching Scheme (Hrs./Week)			Examination Scheme (Marks)						Credits			
			L	P	T	UE	IA	TW	TW & OR	TW & PR	Total	L	P TW/OR/PR	T	Total
1		Mathematics for Computing-II	3	-	1	60	40	-	-	-	100	3	-	1	4
2		Organic and Electrochemistry	3	2	-	60	40	25	-	-	125	3	1	-	4
3		Electrical Technology	4	2	-	60	40	25	-	-	125	4	1	-	5
4		Object Oriented Programming	4	2	-	60	40	-	-	75	175	4	1	-	5
5		Programing Paradigms	4	2	-	60	40	-	50	-	150	4	1	-	5
6		Web Programming	-	4	-	-	-	-	-	75	75	-	2	-	2
	Total		**18**	**12**	**1**	**300**	**200**	**50**	**50**	**150**	**750**	**18**	**6**	**1**	**25**

Program: B.TECH. (Information Technology) Semester – III CBCS 2021 Course

Sr. No.	Course Code	Name of Course	Teaching Scheme (Hrs./Week)			Examination Scheme (Marks)						Credits			
			L	P	T	UE	IA	TW	TW & OR	TW & PR	Total	L	P TW/OR/PR	T	Total
1		Discrete Structures and Graph Theory	4	2	-	60	40	25	-	-	125	4	1	-	5
2		Data Structures	4	2	-	60	40	-	-	50	150	4	1	-	5
3		Database Management System	3	2	-	60	40	-	-	50	150	3	1	-	4
4		Software Engineering*	4	2	-	60	40	25	-	-	125	4	1	-	5
5		Computer Communication and Networks	3	2	-	60	40	25	-	-	125	3	1	-	4
6		Information Technology Laboratory - I	-	4	1	-	-	-	25	50	75	-	2	1	3
7		Vocational Course-I	-	-	-	-	-	-	50	-	50	-	2	-	2
8		MOOC - I	-	-	-	-	-	-	-	-	-	-	-	-	2
9		Environmental Studies** (Mandatory Audit Course)	-	-	-	-	-	-	-	-	-	-	-	-	-
	Total		**18**	**14**	**1**	**300**	**200**	**75**	**75**	**150**	**800**	**18**	**9**	**1**	**30**

*** Industry Taught Course -I** **** 100 Marks Theory Examination**

Vocational Course - I

Sr. No.	Course Name	Offered By	Offered By	Offered By
1.	Database Basics	Oracle	Microsoft	IBM
2.	Data Structures	Coursera	CodeChef	edX Certification
3.	Front-End Web Developer	The World Wide Web Consortium (W3C)	Udemy	Microsoft
4.	Object Oriented Programming	Dakota State University	North Hennepin Community College	Udemy
5.	SQL	Oracle	UpGrad	Coursera

Program: B.TECH. (Information Technology) Semester – IV CBCS 2021 Course

Sr. No.	Course Code	Name of Course	Teaching Scheme (Hrs./Week)			Examination Scheme (Marks)						Credits			
			L	P	T	UE	IA	TW	TW & OR	TW & PR	Total	L	P TW/OR/PR	T	Total
1		IT Infrastructure Management*	4	2	-	60	40	25	-	-	125	4	1	-	5
2		Formal Languages and Computation Theory	4	2	-	60	40	25	-	-	125	4	1	-	5
3		Microprocessor and Microcontrollers	3	2	-	60	40	-	-	50	150	3	1	-	4
4		Applied Algorithms	4	2	-	60	40	25	-	-	125	4	1	-	5
5		Operating System	3	2	-	60	40	-	-	50	150	3	1	-	4
6		Information Technology Laboratory – II	-	4	1	-	-	-	25	50	75	-	2	1	3
7		Vocational Course - II	-	-	-			-	50	-	50	-	2	-	2
8		Social Activities – I	-	-	-			-	-	-	-	-	-	-	2
9		Disaster Management** (Mandatory Audit Course)	-	-	-			-	-	-	-	-	-	-	-
	Total		**18**	**14**	**1**	**300**	**200**	**75**	**75**	**150**	**800**	**18**	**9**	**1**	**30**

*** Industry Taught Course – II** **** 100 Marks Theory Examination**

Vocational Course - II

Sr. No.	Course Name	Offered By	Offered By	Offered By
1.	Full Stack Web Development	The Technical University of Munich, Harvard	Oracle	World Wide Web Consortium (W3C)
2.	Java SE Programmer	Oracle	UpGrad	Udemy
3.	Microprocessor and Microcontrollers	Purdue University Fort Wayne, Vignan Institute of Technology and Science, Telangana, Electronics & ICT Academy at IIT Kanpur	Udemy	Coursera
4.	Operating System	Oracle	Linux foundation	Coursera
5	IT Infrastructure Management	LinkedIn Learning	CompTIA	Udemy

Program: B.TECH. (Information Technology) Semester – V CBCS 2021 Course

Sr. No.	Course Code	Name of Course	Teaching Scheme (Hrs./Week)			Examination Scheme (Marks)						Credits			
			L	P	T	UE	IA	TW	TW & OR	TW & PR	Total	L	P TW/OR/PR	T	Total
1		Human Computer Interaction	3	2	-	60	40	25	-	-	125	3	1	-	4
2		Artificial Intelligence and Machine Learning	4	2	-	60	40	25	-	-	125	4	1	-	5
3		Computer Architecture and Organization	3	-	-	60	40	-	-	-	100	3	-	-	3
4		Advanced Database System*	3	2	-	60	40	-	-	50	150	3	1	-	4
5		Mobile Application Development	3	2	-	60	40	-	-	50	150	3	1	-	4
6		Information Technology Laboratory- III	-	4	-	-	-	-	-	50	50	-	2	-	2
7		Vocational Course - III	-	-	-	-	-	-	50	-	50	-	2	-	2
8		Project - I Stage - I	-	2	-	-	-	-	100	-	100	-	4	-	4
9		MOOC-II	-	-	-	-	-	-	-	-	-	-	-	-	2
	Total		**16**	**14**	**0**	**300**	**200**	**50**	**150**	**150**	**850**	**16**	**12**	**-**	**30**

*** Industry Taught Course – III** **Vocational Course - III**

Sr. No.	Course Name	Offered By	Offered By	Offered By
1.	Mobile App Development	Harvard Curtin University	Massachusetts Institute of Technology	IBM
2.	Python	Google	Coursera	edX
3.	Game Development	Professional Certificate in Computer Science for Game Development- Harvard	UpGrad	Intel
4.	Mobile Computing with App Inventor	CS Principles - Trinity College, Dublin	University of Hartford, Connecticut	Google
5.	Certificate Program In Machine Learning & AI with Python	Indian Institute of Technology Bombay	Coursera	edX

Program: B.TECH. (Information Technology) Semester – VI CBCS 2021 Course

Sr. No.	Course Code	Name of Course	Teaching Scheme (Hrs./Week)			Examination Scheme (Marks)						Credits			
			L	P	T	UE	IA	TW	TW & OR	TW & PR	Total	L	P TW/OR/PR	T	Total
1		Cloud Computing*	3	2	-	60	40	-	50	-	150	3	1	-	4
2		Software Testing and Quality Assurance	3	2	-	60	40	-	-	50	150	3	1	-	4
3		Data Warehousing and Data Mining	4	-	-	60	40	-	-	-	100	4	-	-	4
4		Agile Methodologies	4	-	-	60	40	-	-	-	100	4	-	-	4
5		Information Technology Laboratory- IV	-	4	-	-	-	-	-	50	50	-	2	-	2
6		Quantitative Techniques, Communication and Values	2	2	-	60	40	-	-	-	100	3	-	-	3
7		Project - I Stage - II	-	2				-	100	-	100	-	4	-	4
8		Internship	-	-	-			-	50	-	50	-	3	-	3
9		Vocational Course- IV	-	-	-			-	50	-	50	-	2	-	2
	Total		**16**	**12**	**-**	**300**	**200**	**-**	**250**	**100**	**850**	**17**	**13**	**-**	**30**

***Industry Taught Course - IV**

Vocational Course- IV

Sr.No.	Course Name	Offered By	Offered By	Offered By
1.	Cloud Computing	Udemy, IIT Roorkee and Wiley	IBM	Amazon,Google
2.	Computational Thinking and Data Science	Massachusetts Institute of Technology, Cambridge, MA USA	Coursera	UpGrad
3.	Machine Learning with Python	Massachusetts Institute of Technology, Cambridge, MA USA	IIT Roorkee	NIT Warangal
4.	Data Warehousing	Oracle	Microsoft	Coursera
5.	Software Testing	Edureka	LambdaTest	ISTQB (International Software Testing Qualifications Board)

Program: B.TECH. (Information Technology) Semester – VII CBCS 2021 Course

Sr. No.	Course Code	Name of Course	Teaching Scheme (Hrs./Week)			Examination Scheme (Marks)						Credits			
			L	P	T	UE	IA	TW	TW & OR	TW & PR	Total	L	P TW/OR/PR	T	Total
1		Project Planning & Management	4	-	-	60	40	-	-	-	100	4	-	-	4
2		Web Services*	4	2	-	60	40	-	-	50	150	4	1	-	5
3		Business Intelligence	4	2	-	60	40	-		50	150	4	1	-	5
4		Elective – I	4	2	-	60	40	-	50	-	150	4	1	-	5
5		Information Technology Laboratory -V	-	4	1	-	-	-	25	75	100	-	2	1	3
6		Project - II Stage – I	-	4	-	-	-	-	200	-	200	-	4	-	4
7		MOOC-III	-	-	-			-	-	-	-	-	2	-	2
8		Research Paper Publication	-	-	-			-	-	-	-	-	2	-	2
	Total		**16**	**14**	**1**	**240**	**160**	**-**	**275**	**175**	**850**	**16**	**13**	**1**	**30**

***Industry Taught Course – V**

Elective - I

1. Software Architecture
2. Information Retrieval
3. User Experience
4. Storage Area network

Program: B.TECH. (Information Technology) Semester – VIII CBCS 2021 Course

Sr. No.	Course Code	Name of Course	Teaching Scheme (Hrs./Week)			Examination Scheme (Marks)						Credits			
			L	P	T	UE	IA	TW	TW & OR	TW & PR	Total	L	P TW/OR/PR	T	Total
1		Information Security	4	2	-	60	40		-	50	150	4	1	-	5
2		Elective – II	4	2	-	60	40	-	50	-	150	4	1	-	5
3		Internet of Things*	4	2	-	60	40	-	-	50	150	4	1	-	5
4		Data Engineering	4	-	-	60	40	-	-	-	100	4	-	-	4
5		Information Technology Laboratory-VI	-	4	1	-	-	-	25	75	100	-	2	1	3
6		Project - II Stage - II	-	4	-	-	-	-	200	-	200	-	6		6
7		Social Activities - II	-	-	-	-	-	-	-	-	-	-	-	-	2
	Total		**16**	**14**	**1**	**240**	**160**	**-**	**275**	**175**	**850**	**16**	**11**	**1**	**30**

***Industry Taught Course – VI**

Elective - II

1. Semantic Web Mining
2. Social Analytics in Digital Marketing
3. Management Information System
4. Cyber security

MOOC I, II, III Courses:

MOOC-I	MOOC-II	MOOC-III
Data Structure and Algorithms Using Java	Computer Architecture & Organization	Data Analytics with Python
Computer Graphics	Software Project Management	Big Data Computing
Discrete Mathematics	Advanced Database Management System	Information security - IV
Introduction to Database Systems	Introduction to Artificial Intelligence	Introduction to Human Computer Interaction
Programming in Java	Google Cloud Computing Foundation Course	Data Mining
Software Engineering	Modern Application Development	Introduction to Internet of Things
An Introduction to Probability in Computing	Machine Learning, ML	Cloud computing
Numerical Methods	The Joy of Computing using Python	Wireless Ad Hoc and Sensor Networks
Problem Solving through programming in C	Probability and Statistics	Introduction to Cyber Security
Communication Skills	Linear Algebra	Social Networks

B.Tech. (Information Technology)

Semester-I

Mathematics for Computing-I

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
Lecture:	**3 Hours/Week**	**End Semester Examination:**	**60 Marks**	**Theory**	**3**
Tutorials:	**1 Hours/Week**	**Internal Assessment:**	**40 Marks**	**Tutorial**	**1**
Total	**4 Hours/Week**		**100 Marks**		**4**

Course Objective:

To study

1. Linear equations and its basis and dimension.
2. Linear mapping and its matrix representation.
3. Orthogonalization and diagonalization of matrices.

Prerequisite: The students should have knowledge of algebra of matrices and determinants.

Course Outcomes: On completion of the course, students will have the ability to:

1. Apply rank of matrix in solving system of equations.
2. Identify basis and dimension of matrix.
3. Solve problems on kernel and image of linear transformation.
4. Apply linear operator to represent matrix.
5. Evaluate orthogonalization of inner product space.
6. Use methods to find eigen values and eigen vectors.

Unit I **06 Hours**

System of Linear Equation: Vectors and linear combinations, Rank of a matrix, Gaussian elimination, LU Decomposition, Solving Systems of Linear Equations using the tools of Matrices.

Unit II **06 Hours**

Vector Spaces: Definition, linear combination, spanning sets subspaces, linear dependence and independence, basis and dimension, rank of matrix.

Unit III **06 Hours**

Linear Mapping: Linear mapping, Kernel and image of linear mapping, rank and nullity of a linear mapping, singular and non-singular linear mapping

Unit IV **06 Hours**

Linear mapping and matrices: Matrix representation of linear operator, change of base, similarity matrices

Unit V **06 Hours**

Inner Product space and orthogonalization: Inner product space, Cauchy-schwarz equality, orthogonality, orthogonal sets and bases, projections, Gramschidt orthogonalization, orthogonal and positive definite matrices, matrix representation of inner produc**t**

Unit VI **06 Hours**

Diagonalization Eigen values and eigen vectors:

Characteristic polynomial, Cayley-Hamilton theorem, eigen values and eigen vectors, properties.

List of Assignment for Internal Assessment will be framed by respective Course Coordinator.

Textbooks/Reference Books

1. P. N. Wartikar and J. N. Wartikar, Applied Mathematics (Volumes I and II), 7th Ed., Pune Vidyarthi Griha Prakashan, Pune, 2013.
2. B. S. Grewal, Higher Engineering Mathematics, 42nd Ed., Khanna Publication, Delhi
3. B.V. Ramana, Higher Engineering Mathematics, 6th Ed., Tata McGraw-Hill, New Delhi, 2008.
4. Erwin Kreyszig, Advanced Engineering Mathematics, 10th Ed., John Wiley & Sons, Inc., 2015.
5. Peter V. O'Neil, Advanced Engineering Mathematics, 7th Ed., Cengage Learning, 2012.
6. Michael Greenberg, Advanced Engineering Mathematics, 2nd Ed., Pearson Education, 1998.

Project Based Learning Assignments*

Note: - *Students in a group of 3 to 4 shall complete any one project from the following list)

Students are expected prepare report on any one topic, write its definition, applications and illustrate with few examples. Also, write pseudo code for it, wherever applicable.

1. Gauss Elimination method.
2. LU-decomposition method
3. Rank of matrix
4. Linear combination
5. Basis and dimension
6. Spanning sets
7. Kernel and image of linear transformation
8. Rank-nullity theorem
9. Non-singular linear mapping
10. Linear operator
11. Similarity matrices
12. Change of base
13. Cauchy Schwarz equality
14. Orthogonality
15. Gram schmidt Orthoginalization
16. Matrix representation of matrix
17. Cayley-Hamilton theorem
18. Eigen values and Eigen vectors

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Physics for Computing System

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
Lecture:	**03 Hours/Week**	**End Semester Examination:**	**60Marks**	**Theory**	**3**
Practical:	**02 Hours/Week**	**Internal Assessment:**	**40 Marks**		
		Term Work:	**25 Marks**	**Practical**	**1**
Total	**5 Hours/Week**		**125 Marks**		**4**

Course Objective: To impart knowledge of basic concepts in physics relevant to engineering applications in a broader sense with a view to lay foundation for the Computer Engineering and Science.

Prerequisite: Students are expected to have a basic understanding of physics and calculus.

Course Outcomes: At the completion of the course, the students should be able to**:**

1. Interpret the properties of charged particles to develop modern instruments such as electron microscopy.
2. Appraise the wave nature of light and apply it to measure stress, pressure and dimension etc.
3. Summarize the structure and properties of lasers to their performance and intended applications.
4. Classify the optical fiber, understanding the structure, types and its applications in the field of communication.
5. Solve quantum physics problems to micro level phenomena and solid-state physics.
6. Explain mechanical properties of solid matter, and connect to applications in the field of engineering.

Unit I **06 Hours**

Modern Physics: Motion of a charged particle in electric and magnetic fields, Electrostatic and Magnetostatics focusing, Electron microscope, Wavelength and resolution, Specimen limitation, Depth of field and focus, Transmission electron microscope (TEM), Scanning electron microscope (SEM), Separation of isotopes by Bainbridge mass spectrograph, Cathode ray tube (CRT).

Unit II **06 Hours**

Wave Optics

Interference: Interference of waves, interference due to thin film (Uniform and nonuniform (only formula-no derivation is expected), Newton's ring, Applications of interference (optical flatness, highly reflecting films, non-reflecting coatings).

Diffraction: Introduction, Classes of diffraction, Diffraction at a single slit (Geometrical method), Conditions for maximum and minimum, Plane diffraction grating, Conditions for principal maxima and minima

Polarization: Introduction, Double refraction and Huygen's theory, Positive and negative crystals, Nicol prism, Dichroism.

Unit III **06 Hours**

Lasers: Principle of laser, Einstein's coefficients, Spontaneous and stimulated emission, Population inversion, Ruby laser, Helium-Neon laser, Semiconductor laser, Single Hetro-junction laser, Gas laser: CO_2 laser, Properties of lasers, Laser speckles, Applications of lasers (Engineering/ industry, medicine, Computers).

Unit IV **06 Hours**

Fiber Optic: Principle of fiber optics, Construction, Numerical Aperture for step index fiber; critical angle, angle of acceptance, V number, number of modes of propagation, types of optical fibers, Fiber optic communication system, advantages, and disadvantages of fiber optics.

Unit V **06 Hours**

Quantum Mechanics: Dual nature of matter, DeBroglie's hypothesis, Heisenberg's uncertainty principle with illustrations, Physical significance of wave function, Schrodinger's time dependant and time independent wave equation, Application of Schrodinger's time independent wave equation to the problems of Particle in a rigid box, step potential and potential barrier (analytical discussion), tunnelling effect.

Unit VI **06 Hours**

Solid state physics: Free electron theory, Density of states, Bloch theorem (Statement only), Origin of band gap, Energy bands in solids, Effective mass of electron, Fermi-Dirac probability function and position of Fermi level in intrinsic semi-conductors (with derivation) and in extrinsic semi-conductors, Band structure of p-n junction diode under forward and reverse biasing, Conductivity in conductor and semi-conductor, Hall effect and Hall coefficient, Photovoltaic effect, Solar cell and its characteristics.

Textbooks

1. A Textbook of Engineering Physics, M N Avadhanulu, P G Kshirsagar and TVS Arun Murthy, S. Chand Publishing (2018).
2. Engineering Physics, R K Gaur and S L Gupta, Dhanpat Rai Publishing Co Pvt Ltd (2015)
3. Concepts of Modern Physics**,** Arthur Beiser**,** Shobhit Mahajan and S. Rai Choudhury, McGraw Hill Education (2017).

Reference Books

1. Fundamentals of Physics, Jearl Walker, David Halliday and Robert Resnick, JohnWiley and Sons (2013).
2. Optics, Francis Jenkins and Harvey White, Tata Mcgraw Hill (2017).
3. Principles of Physics, John W. Jewett, Cengage publishing (2013).
4. Introduction to Solid State Physics, C. Kittel, Wiley and Sons (2004).
5. Principles of Solid-State Physics, H. V. Keer, New Age International (1993).
6. Laser and Non-Linear Optics, B. B. Laud, New Age International Private Limited (2011).
7. Nanotechnology: Principles and Practices, Dr. S. K. Kulkarni, Capital Publishing Company (2014).
8. Science of Engineering Materials- C.M. Srivastava and C. Srinivasan, New Age International Pvt. Ltd. (1997).
9. Introduction to Electrodynamics –David R. Griffiths, Pearson (2013).
10. Renewable Energy: Power for a Sustainable Future, Boyle, Oxford University Press (2012).

List of Laboratory Exercise (Any Eight of the Following)

1. Study of lissajous figure by Cathode Ray Oscilloscope (CRO).
2. Determination of e/m by Thomson method.
3. Determination of radius of planoconvex lens/wavelength of light/Flatness testing by Newton's rings.
4. Determination of wavelength of light using diffraction grating.
5. Determination of resolving power of telescope.

6. Determination of thickness of a thin wire by air wedge.
7. Determination of refractive index for O-ray and E-ray.
8. Determination of divergence of a laser beam.
9. Particle size by semiconductor laser.
10. Determination of wavelength of laser by diffraction grating.
11. To study Hall effect and determine the Hall voltage.
12. Calculation of conductivity by four probe methods.
13. Study of solar cell characteristics and calculation of fill factor.
14. Determination of band gap of semiconductor.
15. Determination of Planck's Constant by photoelectric effect.

Project Based Learning Assignments*

Note: - *Students in a group of 3 to 4 shall complete any one project from the following list.

1. Measurement and effect of environmental noise in the college
2. Design and simulation of automatic solar powered time regulated water pumping
3. Solar technology: an alternative source of energy for national development
4. Design and construction of digital distance measuring instrument
5. Design and construction of automatic bell ringer.
6. Design and construction of remote-control fan
7. Design and construction of sound or clap activated alarm
8. Electronic eye (Laser Security) as auto switch/security system
9. Electric power generation by road power
10. Determination of absorption coefficient of sound absorbing materials
11. Determination of velocity of O-ray and E-ray in different double refracting materials
12. Need of medium for propagation of sound wave
13. Tesla Coil
14. Thin film interference in soap film-formation of colors
15. LiFi- wireless data transfer system using light

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Computer Aided Drafting

TEACHING SCHEME		EXAMINATION SCHEME	CREDIT SCHEME	
Lecture:	**4 Hours/Week**	**End Semester Examination: 60 Marks**	**Theory**	**4**
Practical:	**2 Hours/Week**	**Internal Assessment: 40 Marks** **Term Work :50Marks**	**Practical**	**1**
Total:	**6 Hours/Week**	**150 marks**		**5**

Course Objectives:

To provide knowledge about

1. Fundamentals of engineering drawing and curves.
2. Isometric views and projection.
3. Projections of points, lines, planes & solids.
4. Use of CAD tools.

Prerequisite: The students should have knowledge of
Basics of mathematics at secondary school level

Course Outcomes: On completion of the course, students will have the ability to:

1. Understand dimensioning methods and drawing of engineering curves.
2. Draw orthographic projections using 1st angle method of projection*.
3. Draw Isometric views from given orthographic projections*.
4. Draw projection of Lines, its traces, and projections of planes*.
5. Draw projection of different solids*.
6. Draw development of lateral surfaces of solids*.

*Using CAD tools

Unit I **08 Hours**

Lines and Dimensioning in Engineering Drawing and Engineering Curves: Different types of lines used in drawing practice, Dimensioning – linear, angular, aligned system, unidirectional system, parallel dimensioning, chain dimensioning, location dimension and size dimension.

Ellipse by Arcs of Circle method, Concentric circle method. Involutes of a circle, Cycloid, Archimedean Spiral, Helix on cone & cylinder, Introduction to Auto CAD commands.

Unit II **08 Hours**

Orthographic Projections: Basic principles of orthographic projection (First and Third angle method). Orthographic projection of objects by first angle projection method only. Procedure for preparing scaled drawing, sectional views, and types of cutting planes and their representation, hatching of sections. (Also using AutoCAD commands).

Unit III **08 Hours**

Isometric Projections: Isometric view, Isometric scale to draw Isometric projection, non-Isometric lines, and construction of Isometric view from given orthographic views and to construct Isometric view. (Also using AutoCAD commands).

Unit IV **08 Hours**

Projections of Points, Lines and Planes: Projections of points, projections of lines, lines inclined to one reference plane, Lines inclined to both reference planes. (Lines in First Quadrant Only) Traces of lines,

Projections of Planes, Angle between two planes, Distance of a point from a given plane, Inclination of the plane with HP, VP. (Also using AutoCAD commands).

Unit V **08 Hours**

Projection of Solids: Projection of prism, pyramid, cone and cylinder by rotation method. (Also using AutoCAD commands).

Unit VI **08 Hours**

Development of Lateral Surfaces of Solids: Introduction to development of lateral surfaces and its Industrial application, draw the development of lateral surfaces of cone, pyramid, and prism. (Also using AutoCAD commands).

Textbooks

1. "Elementary Engineering Drawing", N. D. Bhatt, Charotar Publishing house, Anand India.
2. "AutoCAD 2020 Beginning and Intermediate", Munir Hamad, Mercury Learning & Information Publication, 2019.
3. "Engineering Drawing and Graphics", Venugopal K., New Age International publishers.

Reference Books

1. "Textbook on Engineering Drawing", K. L. Narayana & P. Kannaiah, Scitech Publications, Chennai.
2. "Fundamentals of Engineering Drawing", Warren J. Luzzader, Prentice Hall of India, New Delhi.
3. "Engineering Drawing", M. B. Shah and B.C. Rana, 1st Ed, Pearson Education, 2005
4. "Engineering Drawing", P. J. Shah, C. Jamnadas and Co., 1st Edition,1988
5. "Engineering Drawing (Geometrical Drawing)", P. S. Gill, 10th Edition, S. K. Katariaand Sons, 2005.

List of Laboratory Exercise

All sheets should be completed using AutoCAD.

List of Drawing Sheets

1. Types of lines, Dimensioning practice, free-hand lettering, 1nd and 3rdangle methods symbol.
2. Engineering curves.
3. Orthographic Projections.
4. Isometric views.
5. Projections of Points and Lines and planes.
6. Projections of Solids.
7. Development of lateral surfaces

List of Assignments: Assignment questions are supposed to be solved in A3 size sketchbook

1. At least 4 questions on engineering curves.
2. At least 2 questions on orthographic projections without sections.
3. At least 2 questions on sectional orthographic projections.

4. At least 2 questions on isometric views.
5. At least 4 questions on projections of lines.
6. At least 4 questions on projections of planes.
7. At least 4 questions on projections of solids.
8. At least 4 questions on development of lateral surfaces.

Project Based Learning Assignments*

Note:- *Students in a group of 3 to 4 shall complete any one project from the following list)

1. To obtain industrial drawings to identify the types of lines, dimensioning methods and method of projection.
2. To develop the model/charts based on engineering curves.
2. To prepare model/chart for identification of engineering curves in nature for industrial, societal etc application.
4. To demonstrate different methods of orthographic projection.
5. To demonstrate projection of Points.
6. To demonstrate projection of Lines.
7. To demonstrate projection of Planes.
8. To demonstrate projection of Solids.
9. To demonstrate developments of surfaces for solids.
10. To demonstrate industrial application of development of surfaces such as steam carrying pipes, Ducts of air conditioning systems, etc.
11. To demonstrate Isometric projection method through model of a cube.
12. To obtain industrial drawings to identify the types of lines, dimensioning methods, and method of projection.
13. To develop the model/charts based on engineering curves.
14. To prepare model/chart for identification of engineering curves in nature for industrial, societal, etc application.
15. To demonstrate different methods of orthographic projection.

Syllabus for Unit Tests:

Unit Test-1	Unit-I, Unit-II, Unit-III
Unit Test-2	Unit-IV, Unit-V, Unit-VI

Digital Electronics

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
Lectures:	**4 Hours/Week**	**End Semester Examination:**	**60 Marks**	**Theory**	**4**
Practical:	**2 Hours/Week**	**Internal Assessment:**	**40 Marks**	**Practical**	**1**
		Term Work:	**25 Marks**		
Total	**6 Hours/Week**		**125 Marks**		**5**

Course Objective:

1. To present the Digital fundamentals, Boolean algebra and its applications in digital systems.
2. To familiarize with the design of various combinational digital circuits using logic gates
3. To introduce the analysis and design procedures for synchronous and asynchronous sequential circuits.
4. To understand the various semiconductor memories and related technology.

Prerequisite: Physics, Mathematics, Basics of electrical engineering

Course Outcomes: On completion of the course, students will have the ability to:

1. Comprehend different number systems and Boolean algebraic principles.
2. Apply logic design minimization techniques to simplify Boolean expressions
3. Analyze and design combinational logic circuits.
4. Demonstrate the operations of systems with sequential circuit elements.
5. Comprehend characteristics and structure of Programmable Logic Devices and Memory.
6. Draw ASM charts for sequential circuit design.

Unit I **08 Hours**

Digital systems:

Number Systems: Introduction to Number Systems-Decimal, Binary, Octal, Hexadecimal, Conversion of number system, Representation of Negative Numbers, 1's complement and 2's complement.

Binary Arithmetic: Binary addition, Binary subtraction, Subtraction using 1's complement and 2's complement, Binary multiplication, and division.

Digital Codes: BCD code, Excess-3 code, Gray code and ASCII code.

Logic Gates: Logical Operators, Logic Gates-Basic Gates, Universal Gates, realization of other gates using universal gates.

Unit II **08 Hours**

Logic Design Minimization: Boolean algebra, De Morgan's Theorems, Standard representation of logic functions, Sum of Product (SOP) form, Product of Sum (POS) form, Simplification of logical functions, Minimization of SOP and POS forms using Karnaugh-Maps up to 4 variables Don't care condition, Quine-McCluskey Method.

Unit III **08 Hours**

Combinational Circuits: Binary and BCD arithmetic, Half Adder, Full Adder, Half Subtractor, Full Subtractor, Binary Adder (IC 7483), BCD adder, Code converters, Multiplexers, De multiplexer, Decoder (IC 74138) and their use in combinational logic design, Priority Encoder, Digital Comparators, Parity generators and Checker (IC 74180), ALU.

Unit IV **08 Hours**

Sequential Circuits: Flip- flop: SR, JK, D, T flip flops, Truth Tables and Excitation tables, Conversion from one type to another type of Flip Flop.
Registers: Buffer register, Shift register.
Counters: Asynchronous counters, Synchronous counters, Modulus counters

Unit V **08 Hours**

FSM and ASM charts: Introduction to FSM, Moore and Mealy State machine, state machine as a sequential controller. Design of state machines: state table, state assignment, transition/excitation table, excitation maps and equations, logic realization, ASM chart notations, ASM block, State diagram, ASM chart for sequential circuits, Multiplexer Controller.

Unit VI **08 Hours**

Memory and PLD: Semiconductor memories: memory organization, memory expansion, Classification and characteristics of memories, RAM, ROM, EPROM, EEPROM, NVRAM, SRAM, DRAM. Programmable logic devices: Study of PROM, PAL, PLAs. Architecture of PLA, designing combinational circuits using PLDs.

List of Internal Assignment will be framed by respective Course Coordinator.

Textbooks

1. M. Morris Mano and M. D. Ciletti, Digital Design, Pearson Education.
2. R. P. Jain, Modern Digital Electronics, Tata McGraw Hill Publication.
3. F.J. Hill and G.L. Peterson, Switching Theory and Logic Design, John Wiley
4. J.F. Wakerly "Digital Design: Principles and Practices", 3rd edition, 4th reprint, Pearson Education, 2
5. David J. Comer, Digital Logic & State Machine Design, Oxford University Press.
6. Digital Integrated Electronics- H.Taub & D.Shilling, Mc Graw Hill.

List of Laboratory Exercises:

1. Verify truth tables of logic gates. (AND, OR, XOR, NOT, NAND, NOR). Simplify the given Boolean expression using K-map and implement using gates
2. State De-Morgan's theorem and write Boolean laws. Implement NAND and NOR as Universal gates.
3. Design (truth table, K-map) and implement half and full adder/subtractor.
4. Design (truth table, K-map) and implement 4-bit BCD to Excess-3 Code converters.
5. Study of magnitude Comparator using IC 7485.
6. Implement of logic functions using multiplexer IC 74151 (Verification, cascading & logic function implementation).
7. Implement logic functions using 3:8 decoder IC 74138.
8. Verify truth tables of different types of flip flops.
9. Design (State diagram, state table & K map) and implement 3 bits Up and Down Asynchronous and Synchronous Counter using JK flip-flop.
10. Design and implement modulo 'n' counter with IC 7490.

Project Based Learning Assignments*

Note: - *Students in a group of 3 to 4 shall complete any one project from the following list)

1.Survey report of basic gates ICs 7432, 4011, 4050, 4070,4071,40106
2. Implement combinational logic Circuit of given Boolean Equation.
3. Implement Half Adder and Half Subtractor.

4. Implement Full Adder using two Half Adders
5. Build 4-bit parallel Adder / Subtractor using IC.
6. Build Code Converters: Binary to Gray
7. Build Code Converters: Excess 3 to Binary)
8. Implement Two Bit Magnitude Comparator using IC 7485
9. Implement given combinational logic using MUX
10. Implement 7 segment decoder driver using IC 7447.
11. Build a Decade counter and Up-Down Counter.
12. Build a Shift Registers: SISO and SIPO
13. Implement the Johnson Counter and Ring Counter.
14. Survey Report on Static I/O and transfer Characteristic of TTL and CMOS.

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit –V, Unit - VI

Structured Programming

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
Lecture:	**4 Hours/Week**	**End Semester Examination:**	**60 Marks**	**Theory**	**4**
Practical:	**2 Hours/Week**	**Internal Assessment:**	**40 Marks**		
		Term Work and Practical:	**100 Marks**	**Practical:**	**1**
Total	**6 Hours/Week**		**200 Marks**		**5**

Course Objective:

1. To build the programming skills using 'C' to solve real world problems.
2. To provide an overview of fundamental principles, concepts, and constructs of computer programming.

Prerequisite:

Basic knowledge of Computer Handling.

Course Outcomes: On completion of the course, students will have the ability to:

1. Apply steps towards problem solving.
2. Apply fundamental concepts of programming language.
3. Implement conditional, branching and iteration
4. Decompose a problem into functions.
5. Apply programming to solve simple numerical method problems.
6. Exercise structures to formulate programs.

Unit I **08 Hours**

Introduction to Computing: Components of computer system, concept of hardware and software, introduction to system software- operating system, editor, compiler, assembler, linker, loader, introduction to computer programming, types of programming languages, software development life cycle, problem solving techniques- fundamental stages of problem solving, define the problem, -designing-development of an algorithm, algorithm design tools- flowcharts, pseudo codes.

Unit II **08 Hours**

Programming language 'C': Features of C, header files, pre-processor directives, compiling and executing a C program, syntax and semantic errors, libraries, structure of a C program, declarations, constants, variables, data types, operators and expressions, precedence and associativity of operators, type conversions, input, and output functions- printf and scanf.

Unit III **08 Hours**

Control Structures: if-else statement, nested if-else, use of logical operators, Loop control structure: for, while, do-while loops, use of break and continue, Case control structure: switch case

Pointers: Concept, pointer declaration, assignment, initialization, and access.

Unit IV **08 Hours**

Function: Types of functions, function definition and declaration, function prototype, calling and returning function, passing values between functions, standard library functions and user defined functions, passing array as function parameter, call-by-value, call-by-reference, recursive function.

Unit V **08 Hours**

Arrays: Concept, declaration, initialization, processing with array, one and multidimensional array, pointer to an array, use of array for searching techniques: linear and binary search. sorting techniques: bubble sort, insertion sort, selection sort, applications of array in image processing.
Strings: concept, declaration, initialization, and standard string library functions.

Unit VI **08 Hours**

Structures: Concept, declaration, accessing structure elements, array of structures, pointer to structures, self-referential structures, use of structures, union.
Introduction command line concepts, programs using command line argument.

List of Internal Assignment will be framed by respective Course Coordinator.

Textbooks

1. Brian W. Kernighan, Dennis M. Ritchie, "The C Programming Language", Prentice Hall, ISBN 0131103628.
2. Donald E. Knuth, "The Art of Computer Programming", Addison-Wesley, ISBN-10: 0201485419, ISBN13: 978-0201485417.
3. T. E. Bailey, "Program design with pseudo code", Brooks/Cole Publisher, ISBN-10 : 0534055745, ISBN-13: 978-0534055745.
4. Kanetkar Yashavant P, "Let us C", BPB publications.
5. Subrata Saha and Subhodip M., "Basic Computation and Programming with C", Cambridge University of Press, India, ISBN:9781316601853.

Reference Books

1 Lamey Robert, "Logical problem solving", Prentice Hall, ISBN: 9780130618825.
2 Henry Mullish, Herbert L. Cooper, "The Spirit of C", Thomson Learning, ISBN 0314285008.

List of Laboratory Exercise

1. Write a program to accept the length of three sides of a triangle and to test and print the type of triangle as equilateral, isosceles or right angled or none.
2. Write a program to check whether input number is Prime or not with and without use of recursive function.
3. Write a program to separate digits of input 4-digit integer, separate and display its digits.
4. Write a program to implement linear and binary search techniques.
5. Write a program to implement sorting techniques: Bubble, Selection, and Insertion sorting.
6. Write a program to accept a string and to display the following:
   (a) Total number of characters in the string.
   (b) Total number of vowels in the string.
   (c) Total number of occurrences of character in the string.
   (d) Check whether string is palindrome or not.
7. Write a program to carry out following operations on strings using library functions.
   (a) To concatenate a string S2 to string S1.
   (b) To find the length of a given string.
   (c) To compare two strings S1 and S2.
   (d) To copy a string S2 to another string S1.
8. A class teacher wants to keep record of 10 students in the class along with the names and marks obtained in 5 subjects. Write a C program with function that displays.
   (a) Name of the student with highest marks in a particular subject.
   (b) Overall percentage result of the class.

(c) Total number of passing students in the class.
(d) Total number of students failing in one subject.

9. Write a program with function to swap values of two elements (call by reference).

Project Based Learning Assignments*

Note:- *Students in a group of 3 to 4 shall complete any one project from the following list)

1. Design and develop a project for Diary management System
2. Design and develop a project for Calendar using C
3. Design and develop a project for Contact Management System
4. Design and develop a project for Library Management System
5. Design and develop a project for Snake Game
6. Design and develop a project for Bus Reservation system
7. Design and develop a project for Hospital Management system
8. Design and develop a project for Employee management system
9. Design and develop a project for Diary management System
10. Design and develop a project for Calendar using C
11. Design and develop a project for Contact Management System
12. Design and develop a project for Library Management System

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit – III
Unit Test -2	Unit – IV, Unit –V, Unit - VI

Computer System Workshop Technology

TEACHING SCHEME	EXAMINATION SCHEME	CREDIT SCHEME
Practical: 4 Hours/Week	**Term Work and Practical : 50 Marks**	**Practical: 2**
Total 4 Hours/Week	**50 Marks**	**2**

Course Objective:
Provide student a knowledge of computer hardware and networking, enabling them to identify computer hardware, software and network related problems, and develop an ability to use the basics of computing, necessary for computing courses.

Prerequisite: Basic knowledge of Computer and Electronics.

Course Outcomes: On completion of the course, students will have the ability to:

1. Identify the architecture of a computer and its different components, including their technology evolution.
2. Apply their knowledge about computer peripherals to identify problems.
3. Install and uninstall given software step-by-step.
4. Configure local area network to access the Internet.
5. Prepare document using Latex.
6. Use GitHub tool for coding and collaboration.

Unit I **06 Hours**

Computer hardware peripherals: Introduction to hardware components, random access memory (RAM), Types Of RAM & their speed, tips for buying ram, how to add memory to a computer, problems when installing memory, Central Processing Unit (CPU), Types Of CPU: considerations when buying a new CPU (Types & Differences), different speeds available for CPU and what do they mean, 32 Bit vs 64 Bit – Which One To Choose &Why? How to choose a CPU type for different needs? Graphic Card & Types, How to install a Graphics Card, Installing a CD or DVD burner, Jumper Switch settings, Hard Disk upgrade, Different ports and why we use them - USB, PS2, DivX, Graphic card & types, Virtual Memory and how to configure it for optimum system performance.

Unit II **06 Hours**

Assembly of Computer and Software Installations: Assembling the motherboard, Replacing fan, how to avoid common mistakes during assembly, Installation of system software: Operating system (Windows and Linux),Installations step for operating system, Dual booting, Configure the BIOS, Installation of Antivirus, Installation of the open source software such as Scilab, Latex Installation of MS Office.

Unit III **06 Hours**

Basic Diagnostic of Hardware and Software: Diagnosis of Power Up problem, Boot Drive, Errant Keyboard, mouse problems, slow computer performance, Computer freezes and displays BSOD (Blue screen of death), no display on monitor, no sound, computer rebooting or turning itself off, how to troubleshoot a computer that does not boot, Registry Cleaner.

Unit IV **06 Hours**

Computer network environments: Network connecting devices. Configure the TCP/IP setting, connect to Local Area Network and access the Internet, Configuring Wireless network. Server and Its Configuration, Email Clients, Browsers, Office tools, customize web browsers with the LAN proxy settings, bookmarks, search toolbars and pop-up blockers, Browsing netiquettes and cyber laws. Cloud Access Tools.

Unit V **06 Hours**

Configuration of External devices: Physical set-up of Printers- Performing test print out, Printing of document etc, Scanner set-up, Webcam, Bluetooth device, Memory card reader etc.

Unit VI **06 Hours**

Productivity tools: Open-Source Tools Such as Latex, GitHub.
Latex: Format words, lines, and paragraphs, design pages, create lists, tables, references, and figures in LaTeX. Introduction to LaTeX Packages and classes. Using Git, Version Control Systems, interacting with GitHub, Reverting Changes, Creating Pull Requests.

List of Internal Assignment will be framed by respective Course Coordinator.

Textbooks:

1. Introduction to Information Technology, ITL Education Solutions limited, Pearson Education.
2. PC Hardware and A+Handbook – Kate J. Chase PHI (Microsoft).
3. LaTeX Companion – Leslie Lamport, PHI/Pearson.
4. https://nptel.ac.in/courses/106/105/106105081/.
5. http://nptel.ac.in/courses/106105084/.
6. https://guides.github.com/.
7. Introduction to Linux: Installation and Programming, N B Venkateswarlu, BS Publication.

Reference Books

1. IT Essentials PC Hardware and Software Companion Guide Third Edition by David Anfinson and Ken Quamme. – CISCO Press, Pearson Education.
2. Computer Fundamentals, MS Office, and Internet & Web Technology by Dinesh Maidasani.

List of Laboratory Exercise

1. Demonstrate the Computer Hardware Components and explain its working.
2. Demonstrate the Networking Components and explain its working.
3. Installation of operating system MS windows, Unix on the personal computer
4. Installation of Application software Latex, MS office on the personal computer
5. Troubleshooting hardware related problem.
6. Customize web browsers with the LAN proxy settings, bookmarks, search toolbars and pop-up blockers. Also, plug-ins like Macromedia Flash and JRE for applets should be configured.
7. Execution of Important “layout” and formatting commands in Latex,
8. Installation of Antivirus and customize the browsers to block pop ups, block active x downloads to avoid viruses and/or worms

9. Assignment on Pull request, code review and collaboration using GitHub.

Project Based Learning Assignments

Note: - *Students in a group of 3 to 4 shall complete any one project from the following list

1. Collect specifications of similar types of hardware and software and prepare report comparing them
2. Assembling and disassembling the PC back to working condition.
3. Installation of operating systems LINUX on Server and different packages on a PC.
4. Practice hardware troubleshooting exercises related to various components of computer like monitor, drives, memory devices, printers etc. and software troubleshooting related to BIOS etc
5. To start your own computer repair workshop. What would your initial planning involve? What would you look for in terms of building, furnishings, tools and any other equipment that you can think of?
6. Cyber Hygiene: Installing antivirus for Windows.
7. Prepare the report of need of programming language in 21^{st} century.
8. Collect various types of computer hardware and prepare summary report
9. Prepare Seminar report using LaTeX.
10. Prepare Project report using LaTeX.

B.Tech(Information Technology)

Semester-II

Mathematics for Computing-II

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
Lecture:	**3 Hours/Week**	**End Semester Examination:**	**60 Marks**	**Theory**	**3**
Tutorials	**1 Hour/Week**	**Internal Assessment:**	**40 Marks**	**Tutorial**	**1**
Total	**4 Hours/Week**		**100 Marks**		**4**

Course Objectives:

1. Fourier series and integral transforms.
2. Multiple integrals and its applications.
3. Vector calculus and its applications.

Prerequisite: The students should have knowledge of vector algebra, derivative and integration.

Course Outcomes: On completion of the course, students will have the ability to:

1. Use periodic functions as Fourier series.
2. Apply methods of finding Fourier and Z-transforms.
3. Apply methods of Laplace transform of piecewise continuous functions.
4. Identify concepts of double and triple integrals.
5. Apply vector derivative for physical quantities.
6. Evaluate line, surface, and volume integrals.

Unit I **06 Hours**

Fourier Series: Definition, Dirichlet's conditions, Fourier Series and Half Range Fourier Series, Harmonic Analysis.

Unit II **06 Hours**

Fourier Transform (FT): Complex Exponential Form of Fourier series, Fourier Integral Theorem, Sine & Cosine Integrals, Fourier Transform, Fourier Sine and Cosine Transform and their Inverses. Introductory.

Z-Transform (ZT): Definition, Standard Properties, ZT of Standard Sequences and their Inverses. Solution of Simple Difference Equations.

Unit III **06 Hours**

Laplace Transform and its application: Definition of LT, Inverse LT. Properties & theorems. LT of standard functions. LT of some special functions viz., Periodic, Unit Step, Unit Impulse, ramp, jump, Problems on finding LT & inverse LT. Applications of LT and Inverse LT for solving ordinary differential equations.

Unit IV **06 Hours**

Multiple Integrals and their Application: Double and Triple integrations, Applications to Area, Volume, Mean and Root Mean Square Values, moment of inertia, centre of gravity.

Unit V **06 Hours**

Vector Differential Calculus: Physical interpretation of Vector differentiation, Vector differential operator, Gradient, Divergence and Curl, Directional derivative, Solenoidal, Irrotational and Conservative fields, Scalar potential, Vector identities.

Unit VI **06 Hours**

Vector Integral Calculus and Applications: Line, Surface and Volume integrals, Work-done, Green's Lemma, Gauss's Divergence theorem, Stoke's theorem. Applications to problem in engineering.

List of Internal Assignment will be framed by respective Course Coordinator.

Textbooks/Reference books:

1.P. N. Wartikar and J. N. Wartikar, Applied Mathematics (Volumes I and II), 7th Ed., Pune Vidyarthi GrihaPrakashan, Pune, 2013.
2.B. S. Grewal, Higher Engineering Mathematics, 42nd Ed., Khanna Publication, Delhi
3.B.V. Ramana, Higher Engineering Mathematics, 6th Ed., Tata McGraw-Hill, New Delhi, 2008.
4.Erwin Kreyszig, Advanced Engineering Mathematics, 10th Ed., John Wiley & Sons, Inc., 2015.
5.Peter V. O'Neil, Advanced Engineering Mathematics, 7th Ed., Cengage Learning, 2012.

Project Based Learning Assignments

Note:- *Students in a group of 3 to 4 shall complete any one project from the following list

Students are expected prepare report on any one topic, write its definition, applications and illustrate with few examples. Also, write pseudo code for it, wherever applicable.

1. Fourier series
2. Harmonic analysis
3. Fourier transform
4. Z-Transform
5. Laplace transform technique to solve ODE
6. Multiple Integral to evaluate area and volume
7. Directional derivative
8. Divergence and curl
9. Greens theorem
10. Gauss Divergence Theorem
11. Stokes theorem
12. Unit step function
13. Solenoidal and irrotational fields
14. Simple difference equation

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Organic and Electrochemistry

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
Lectures:	**3 Hours/Week**	**End Semester Examination:**	**60 Marks**	**Theory**	**3**
Practical:	**2 Hours/Week**	**Internal Assessment:**	**40 Marks**	**Practical**	**1**
		Term Work:	**25 Marks**		
Total	**5 Hours/Week**		**125 Marks**		**4**

Course Objective:

The student should acquire the knowledge of

1. To develop the interest among the students regarding chemistry and their applications in engineering.
2. To develop confidence among students about chemistry, how the knowledge of chemistry is applied in technological field.
3. The student should understand the concepts of chemistry to lay the groundwork for subsequent studies in the computing field.

Prerequisite: Capacitor, insulator, classification and properties of polymers, electromagnetic radiation, electrochemical series

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

1. Differentiate between ionic and covalent bonding and classify the bonding in a compound as ionic or covalent.
2. Develop a working knowledge of the twelve fundamental principles of green chemistry and what it is all about.
3. Apply standard reduction potential data to determine the relative strength of oxidizing/reducing agents.
4. Demonstrate the knowledge of polymer materials for futuristic engineering applications.
5. Describe the properties of materials and Application of semiconductor electronics
6. Describe the manufacturing and refining process of fuels and lubricants.

Unit I **06 Hours**

Chemical Bonding in Molecules: MO theory, Structure, bonding and energy levels of bonding and shapes of many atom molecules, Coordination Chemistry, Electronic spectra and magnetic properties of complexes with relevance to bio-inorganic chemistry, organ metallic chemistry.

Unit II **06 Hours**

Green Chemistry: Introduction, Twelve Principles of Green chemistry, numerical on atom economy, synthesis, adipic acid and indigo. Organic dye-Traditional methods of organic dye. Green solvents (ionic liquid supercritical CO2), and products from natural materials.

Unit III **06 Hours**

Electrochemistry: Electrochemical cells and Galvanic cells, EMF of a cell, Single electrode potential, Nernst equation, Electrochemical series, Types of electrodes, Reference electrodes, pH, pOH, acids and basis, Fuel cells, Construction and Working of - Acid and Alkaline Storage Battery, Dry Cell, Ni-Cd Batteries, Li-Ion Batteries, Li-Po Batteries.

Unit IV **06 Hours**

Polymers for the Electronics Industry: Polymers, Conduction mechanism, Preparation of conductive polymers, Polyacetylene, Poly (p-phenlylene), Polyhetrocyclic systems, Polyaniline Poly (Phenylene sulphide), Poly (1,6-heptadiyne), Applications, Photonic applications.

Unit V **06 Hours**

Semi-Conductors, Insulators and Superconductors: Semi conductivity in non-elemental materials, Preparations of semiconductors, Chalcogen photoconductors, photocopying process Introduction to Superconductors, types of Superconductors, Properties of superconductors, Applications of Superconductors, Electrical insulators, or Dielectrics.

Unit VI **06 Hours**

Fuels and Lubricants: Classification of fuels, Calorific values, Comparison between solid, liquid and gaseous fuels, Theoretical calculation of calorific value of a fuel, Selection of coal, analysis of coal, Natural Gas, Producer gas, water gas, Lubricants, Mechanism of lubrication, classification of lubricants, lubricating oils, Solid lubricants, Greases or Semi-Solid lubricants, Synthetic lubricants, Lubricating emulsions, Properties of lubricating oils.

List of Internal Assignment will be framed by respective Course Coordinator.

Textbooks

1. Polymer Science and technology (2nd Edition), P. Ghosh, Tata McGRAW Hill, 2008.
2. Polymers: Chemistry & Physics of Modern Materials (2nd edition) J.M.G.Cowie, Blackie Academic & Professional, 1994.
3. A Textbook of Engineering Chemistry, Shashi Chawla, Dhanpat Rai & Co, 2004.
4. Engineering Chemistry (16th Edition) Jain, Jain, Dhanpat Rai Publishing Company, 2013.
5. Inorganic Chemistry (4th edition), D. F. Shrives and P. W. Atkins, Oxford University, Oxford, 2006.
6. Applications of Absorption Spectroscopy of Organic Compounds (4th edition), John R. Dyer, Prentice Hall of India Pvt. Ltd., 1978.
7. Reactions, Rearrangements and Reagents (4th edition), S. N. Sanyal, Bharti Bhawan (P & D), 2003.

List of Laboratory Exercise

1. Determination of Hardness of water sample by EDTA method.
2. Determination of Chloride content in water sample by precipitation titration method.
3. To determine strength of acid by pH – metric Titration
4. To measure the Conductance of a solution by conductometric titration
5. Measurement of Surface tension of a given liquid by Stalagmometer.
6. Determination of viscosity of a given liquid by Ostwald's Viscometer.
7. Determination of Saponification value of an oil sample.
8. To determine alkalinity water sample.
9. Determination of Hardness of water sample by EDTA method.
10. Determination of Chloride content in water sample by precipitation titration method.
11. To determine strength of acid by pH – metric Titration
12. To Prepare Phenol formaldehyde/Urea formaldehyde resin.
13. To study set up of Daniel cell.

Project Based Learning Assignments

Note: - *Students in a group of 3 to 4 shall complete any one project from the following list

1. Green Chemistry approach to Nano-Structured Electronics
2. Assessment of Environmentally Benign Photopolymers as an Alternative to the Use of Formaldehyde Based Textile Finishing Agents
3. Solvent-Free Synthesis of Phthalocyanines
4. Synthesis of Conjugated Polymers and Molecules Using Sugar Reagents and Solventless Reactions
5. Environmentally Benign Control of Polymer Solubility: Photoresist Materials Using DNA Mimics
6. Enzymatic Synthesis of Non-Formaldehyde Phenolic Polymers: Control of Hydrogen Peroxide Concentration.
7. The materials chemistry and electrochemistry of lithium and sodium-ion batteries
8. Electroplating- the principles, how different metals can be used and the practical applications
9. Electroplating, Metal Polishing, Anodizing, Phosphating Metal Finishing and Powder Coating Projects.
10. To determine calorific value of a fuel by any suitable method
11. To study various properties of lubricants
12. To study various types of lubricants and its properties.
13. To determine quality of coal sample & its analysis.
14. To study mechanism of lubrication.
15. To study coal analysis & its significance.

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit –V, Unit - VI

Electrical Technology

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
Lecture:	**4 Hours/Week**	**End Semester Examination:**	**60 Marks**	**Theory**	**4**
Practical:	**2 Hours/Week**	**Internal Assessment:**	**40 Marks**		
		Term Work:	**25 Marks**	**Practical**	**1**
Total	**6 Hours/Week**		**125 Marks**		**5**

Course Objective:

To study of power system basics, magnetic circuits electrical machines, transformers, wiring, measurements, illumination and batteries.

Course Outcomes: On completion of the course, students will have the ability to:

1. Explain the various parameters related to magnetic circuit.
2. Describe basic concepts of AC fundamentals and circuits.
3. Illustrate constructional features and describe different parameters of transformer.
4. Describe basic concepts of power system and three phase circuits.
5. Demonstrate AC and DC electrical machines.
6. Classify types of batteries.

Unit I **08 Hours**

Magnetic Circuits: Magnetic effect of electric current, Cross & Dot Convention, Right hand thumb rule, Concept of flux, flux linkages, magnetic field, magnetic field strength, magnetic field intensity, absolute permeability, relative permeability Kirchhoff's laws for magnetic circuits. Magnetic circuit concepts, analogy between electric & magnetic circuits, magnetic circuits with DC and AC excitations, magnetic leakage, B-H curve, hysteresis and eddy current losses, magnetic circuit calculations, mutual coupling

Unit II **08 Hours**

AC Fundamentals and circuits: AC Fundamentals: Sinusoidal, square and triangular waveforms – average and effective values, form and peak factors, concept of phasor, phasor representation of sinusoidal varying voltage and current. Analysis of series, parallel and series parallel RLC Circuits: apparent, active & reactive powers, power factor, causes and problems of low power factor, power factor improvement; resonance in series and parallel circuits, bandwidth and quality factor (simple numerical problems.

Unit III **08 Hours**

Single Phase Transformer: Faradays law of electromagnetic induction, statically and dynamically induced e.m.f, self-inductance, mutual inductance, coefficient of coupling. Single Phase Transformer: Principle of operation, construction, e .m. f. equation, voltage ratio, current ratio, KVA rating ,determination of efficiency and regulation by direct load test, equivalent circuit, power losses,(simple numerical problems), introduction to auto transformer. Three phase transformer and its different winding connections.

Unit IV **08 Hours**

Introduction to Power System and Three Phase Circuits: General layout of electrical power system and functions of its elements, standard transmission and distribution voltages, concept of grid (elementary treatment only) Power generation to distribution through overhead lines and underground cables with single line diagram. Three phase system-its necessity and advantages, meaning of phase

sequence, star and delta connections, balanced supply and balanced load, line and phase voltage/current relations, three phase power and its measurement (simple numerical problems).

Unit V **08 Hours**

Electrical Machines: DC & AC: Principles of electromechanical energy conversion, DC machines: types, e. m. f. equation of generator and torque equation of motor, characteristics, and applications of dc motors (simple numerical problems). Single Phase Induction motor: Principle of operation and introduction to methods of starting, applications. Three Phase Induction Motor: types, Principle of operation, slip-torque characteristics, applications (numerical problems related to slip only).

Unit VI **08 Hours**

Batteries: Basic idea of primary and secondary cells, Construction, working principle and applications of Lead-Acid, Nickel Cadmium and Silver-Oxide batteries, Charging methods used for lead-acid battery (accumulator), Care and maintenance of lead-acid battery, Series and parallel connections of batteries, General idea of solar cells, solar panels and their applications, Introduction to maintenance free batteries, Safe disposal of Batteries; Fuel cell: Principle & Types of fuel cell.

List of Internal Assignment will be framed by respective Course Coordinator.

Textbooks:

1. B.L.Theraja, A Textbook of Electrical Technology, Vol.1, S.Chand& Company Ltd. New Delhi
2. V.K.Mehta, Basic Electrical Engineering, S Chand & Company Ltd. New Delhi.
3. J.Nagarath and Kothari, Theroy and applications of Basic Electrical Engineering, Prentice Hall of India Pvt. Ltd.

Reference Books:

1. Electrical Technology - Edward Huges (Pearson).
2. Basic Electrical Engineering - D. P. Kothari, J Nagarath (TMC).
3. Electrical power system technology - S. W. Fordo, D. R. Patric (Prentice Hall).
4. Electrical, Electronics Measurements and Instruments - (Satya Prakashan).

Project Based Learning Assignments

Note: - *Students in a group of 3 to 4 shall complete any one project from the following list

1. Building a small resistive load lamp bank.
2. Building a small resistive load lamp bank for various types of connections like series, parallel, star, delta
3. Building a small inductive load lamp bank for various types of connections like series, parallel, star, delta
4. Building a small capacitive load lamp bank for various types of connections like series, parallel, star, delta
5. Building a small resistive load lamp bank
6. Building a staircase wiring model on a board
7. Building a Go down wiring model on a board
8. Rewinding of a choke
9. Rewinding of a small transformer
10. Building a small rectifier circuit on bread board
11. Building a mobile charger circuit on a bread board

12. Building an electric buzzer circuit
13. Building a solar charger for mobile phone
14. Building a small wind turbine
15. Small Agricultural pump model with DC motor
16. Small Agricultural pump model with AC motor

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Object Oriented Programming

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT CHEME	
Lecture:	**4 Hours/Week**	**End Semester Examination:**	**60 Marks**	**Theory**	**4**
Practical:	**2 Hours/Week**	**Internal Assessment:**	**40 Marks**	**Practical**	**1**
		Term Work and Practical:	**75 Marks**		
Total	**6 Hours/Week**		**175 Marks**		**5**

Course Objective:
The course focuses on the understanding and practical mastery of object-oriented concepts such as classes, objects, data abstraction, methods, method overloading, inheritance, and polymorphism.

Prerequisite:
Basics of C Programming.

Course Outcomes: On completion of the course, students will have the ability to:

1. Differentiate between top-down and bottom-up programming approach.
2. Associate the object-oriented programming approach in connection with C++.
3. Apply the concepts of array and operator overloading.
4. Implement basic concepts of inheritance.
5. Illustrate the process of data file manipulations using C++
6. Use the concepts of Templates and Exceptions.

Unit I **08 Hours**

Introduction to OOP: Programming characteristics of object-oriented languages. Comparison between C and C++. C++ Programming basics: Output using cout. Directives. Input with cin. Type bool. The setw manipulator. Type conversions.

Unit II **08 Hours**

Functions: Returning values from functions. Reference arguments. Overloaded function. Inline function. Default arguments. Returning by reference.
Object and Classes: Making sense of core object concepts (Encapsulation, Abstraction, Polymorphism, Classes, Messages Association, Interfaces) Implementation of class in C++, C++ Objects as physical object, C++ object as data types of constructor. Object as function arguments. The default copy constructor, returning object from function. Structures and classes. Classes objects and memory static class data. Const and classes.

Unit III **08 Hours**

Arrays and string: arrays fundamentals. Arrays as class Member Data: Arrays of object, string, The standard C++ String class
Operator overloading: Overloading unary operations. Overloading binary operators, data conversion, pitfalls of operators overloading and conversion keywords. Explicit and Mutable.

Unit IV **08 Hours**

I**nheritance**: Concept of inheritance. Derived class and based class. Derived class constructors, member function, inheritance in the English distance class, class hierarchies, inheritance and graphics shapes, public and private inheritance, aggregation: Classes within classes, inheritance, and program development.

Pointer: Addresses and pointers. The address of operator and pointer and arrays. Pointer and Faction pointer and C-types string. Memory management: New and Delete, pointers to objects, debugging pointers.

Unit V **08 Hours**

Virtual Function: Virtual Function, friend function, Static function, Assignment and copy initialization, this pointer, dynamic type information.

Streams and Files: Streams classes, Stream Errors, Disk File I/O with streams, file pointers, error handling in file I/O with member function, overloading the extraction and insertion operators, memory as a stream object, command line arguments, and printer output.

Unit VI **08 Hours**

Templates and Exceptions: Function templates, Class templates Exceptions

The Standard Template Library: Introduction algorithms, sequence containers, iterators, specialized iterators, associative containers, strong user-defined object, function objects.

List of Internal Assignment will be framed by respective Course Coordinator.

Textbooks:

1 Object Oriented Programming with C++ Author: E. Balagurusamy.
2 C++: The complete Reference Author: Herbert Schildt.

Reference Books:

1 Object Oriented Programming C++, Fourth Edition, By Pearson.
2 Object Oriented Programming in C++ Author: Robert Lafore.

List of Laboratory Exercise:

1. Describe the OOP Concepts.
2. Demonstrate class concept using suitable programmes.
3. Demonstrate array concepts using suitable programmes.
4. Demonstrate Operator Overloading concepts using suitable programmes.
5. Demonstrate Inheritance and its types using suitable programmes.
6. Demonstrate the use of Pointer using suitable programmes.
7. Demonstrate the types of functions using suitable programmes.
8. Demonstrate File Handling using suitable programmes.
9. Demonstrate Templates using suitable programmes.
10. Implement User define Exception.

Project Based Learning Assignments

Note:- *Students in a group of 3 to 4 shall complete any one project from the following list

1. Login and Registration System using C++
2. Car Rental System using C++
3. Bookshop inventory system using C++
4. Student Report Management System using C++
5. Sudoku Game using C++
6. Credit Card Validator using C++
7. Using Graphics to Draw and Move Shapes using C++
8. Banking Record System using C++
9. Hotel Management System using C++

10. Student Management System using C++
11. Bus reservation System using C++

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Programming Paradigms

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
Lecture:	**4 Hours/Week**	**End Semester Examination:**	**60 Marks**	**Theory**	**4**
Practical:	**2 Hours/Week**	**Internal Assessment:**	**40 Marks**		
		Term Work and Oral:	**50 Marks**	**Practical**	**1**
Total	**6 Hours/Week**		**150 Marks**		**5**

Course Objectives:

1. To introduce the basic building blocks that underlie programming languages.
2. To introduce the basics of programming language design and implementation.

Prerequisite:

Introduction to computing and programming environment.

Course Outcomes:

On completion of the course, students will have the ability to:

1. Compare and contrast a range of programming paradigms.
2. Apply functional programming language features.
3. Implement the concepts of object orientation.
4. Interpret the features of logic programming paradigm.
5. Summarize the use and types of system programs.
6. Discuss the appropriateness of the using a given programming paradigm within a given environment.

Unit I **08 Hours**

Introduction to Programming: Role of programming languages, need to study programming languages, Characteristics of Programming Languages

The Nature of Programming Languages: Imperative languages and non-imperative languages, Functional Language, Scripting languages, Data-oriented languages, Object-oriented languages, Event-driven Programming, Language Standardisation

Programming Environments: Compilers and Interpreters, Interactive development tools, Run-time support environments, Debugging Tools, Testing Tools, Configuration Management.

Unit II **08 Hours**

Functional Programming: Definition of a function and Subprogram control: domain and range, total and partial functions, strict functions, subprogram sequence control, attributes of data control, shared data in subprograms, different parameter passing methods, lifetime of variables, Recursion, Referential transparency, Storage management. Desirable and undesirable characteristics of procedural programming.

Unit III **08 Hours**

Object Orientation: Basic concepts: Objects, classes, methods, overloading methods, messages inheritance: overriding methods, single inheritance, multiple.

Inheritance, Interfaces (e.g., in Java), encapsulation, polymorphism, Implementing object-oriented programming, desirable characteristics of object-oriented programming, Comparative study of C++ and JAVA.

Unit IV **08 Hours**

Logic programming Paradigm: Introduction, Logic programming language model, Brief Introduction to Predicate Calculus, Predicate Calculus and Proving Theorems, An Overview of Logic Programming, The Origins of Prolog, The Basic Elements of Prolog, Deficiencies of Prolog, Applications of Logic Programming Limitations of Logic Programming.

Unit V **08 Hours**

System Programming: Types and functions of system Programs: Language processors and language processing activities, Assemblers, Macro processor, Linker, Loader, Interpreter, Compiler (steps in compilation).

Unit VI **08 Hours**

Additional Programming Paradigms: Data flow programming design principles, Database programming design principles, Network programming design principles, Socket programming in JAVA, Internet programming design principles, windows programming.

List of Internal Assignment will be framed by respective Course Coordinator.

Textbooks:

1. Roosta Seyed, "Foundations of Programming Languages Design & Implementation", 3rd Edition, Cenage learning, ISBN-13:978-81-315-1062-9.
2. Pratt T.W., Zelkowitz **"**Programming Languages: Design and Implementation" PHI, 2002, 3rd Edition, ISBN-81-203-1038-1
3. Sebesta R. W., "Concepts of programming languages", Pearson Education 2001, 4th edition, ISBN-81-317-0837-3.
4. D.M. Dhamdhere, "Systems Programming and Operating Systems", Tata McGraw-Hill, ISBN- 13:978-0-07-463579-7
5. Max Bramer, "Logic Programming with Prolog", 2nd Edition, Springer, ISBN-13 978-1447154860

Reference Books:

1. Sethi Ravi, **"**Programming Languages: Concepts and Constructs**"** Pearson Education, ISBN:9788177584226
2. Herbert Schildt, "C++: The Complete Reference, 4th Edition", McGraw Hill Education; 4th edition, ISBN-13 : 0070532465-978

List of Laboratory Exercises:

1. Implement parameter passing using functional programming approach.
2. Implement recursion using functional programming approach.
3. Implement and comparing lifetime of variable using functional and object-oriented programming approach.
4. Implement and comparing reference passing using functional and object-oriented programming approach.
5. Implement encapsulation in object-oriented programming approach.
6. Case study of Prolog.
7. Implement and compare functions in functional and object-oriented programming approach.
8. Implement concept of binding in functional and object-oriented programming approach
9. Implement inheritance using object-oriented programming approach.

10. Study of a website/software to identify event driven programming elements used.

Project Based Learning Assignments

Note:- *Students in a group of 3 to 4 shall complete any one project from the following list

1. Make a project in C to maintain student record using files. The project should be able to read, write, modify, add and search records.
2. Make a project in C++ to maintain employee data using files and dynamic object. The project should be able to read, write, modify, add and search records.
3. Implementation of a simple calculator with memory functions in C++ using polymorphism. The screen should continuously display numbers, signs, and symbols similar to calculator. Use shortcut keys for operations and memory functions.
4. Implementation of a simple predicate logic system for diagnosis and applicable medicines using prolog
5. Develop a simulator for assembler. It should accept a assembly program and separate the components of the program as per the data structures of assembler.
6. Develop a macro-processor like program which should identify the macro definitions, macro calls in an assembly program. It should also replace macro calls with macro definitions.
7. Implement a phone book using C/C++.
8. Develop a simple 3-page website to show event elements. It should have at least one registration page to communicate data to and from a server.
9. Implement result calculation system for student marks using each structured programming and object-oriented programming. Make use of files. Compare the difference in both implementation and identify the pros and cons of both implementations with the features of the programming types used.
10. Implement event driven programming on at least one webpage.

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit – III
Unit Test -2	Unit – IV, Unit – V, Unit – VI

Web Programming

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
Practical:	**4 Hours/Week**	**Term Work and Practical:**	**75 Marks**	**Practical:**	**2**
Total:	**4 Hours/Week**		**75 Marks**		**2**

Course Objectives:
To develop the skill & knowledge of Web page design.

Prerequisite:
Basic knowledge in HTML tags & skill of creating web pages should be known

Course Outcomes: On completion of the course, students will have the ability to:

1. Use HTML in website designing according to theme.
2. Design web pages with attributes
3. Design various layout of websites.
4. Implement responsive web design.
5. Implement front end framework with Bootstrap Elements.
6. Build website with Content Management System.

Unit I **06 Hours**

HTML Tags: Choose a Website Topic, Overview of HTML Tags, The HTML 5 Template, The Head, Formatting Content, Compound Tags, Character Entities, Commenting and Formatting Code, Other HTML Tags.

HTML Attributes and Images:
Acquiring Images, Graphics File Formats, Editing Images, The img Tag, Absolute Links, Embedding Media, Relative Links, Validating Code

Unit II **06 Hours**

CSS – Styling Tags and Page Layout: CSS Basics, Colors and Inline Styles, Internal Style Sheets and Basic Formatting, External Stylesheets, Common Properties. Classes IDs Divs Spans, The Box, Boxes in Boxes, Styling Page Divisions, Additional Resources

Designing with Sections: - Sections and Background Colors, Background Images, Adding a Navigation Bar

Unit III **06 Hours**

Publishing Websites: -FTP and Web Servers

JavaScript: - Adding a jQuery Animated Scrolling Effect.

Responsive Design: - Media Queries, Multiple Media Queries, Targeting Devices, Images and Video, Columns and Tweaks, The Viewport

Unit IV **06 Hours**

Front End Frameworks: Explore Bootstrap Elements, Downloading Bootstrap, downloading a Bootstrap Example, Reviewing the Example Code, Replacing Page Content, Customizing the Design.

Unit V **06 Hours**

Web API: Working of APIs, Relationship between JavaScript, APIs, and other JavaScript tools.

Common browser APIs :- APIs for manipulating documents, APIs that fetch data from the server, APIs for drawing and manipulating graphics, Audio and Video APIs, Device APIs, Client-side storage APIs.
Common third-party APIs :- YouTube API, Facebook suite of APIs, Twitter API.

Unit VI **06 Hours**

Content Management Systems: Setting up WordPress, Creating Posts, and Creating Pages, Working with Media, Themes and Widgets.

List of Internal Assignment will be framed by respective Course Coordinator.

Textbooks:

1) Getting Started with Web Components: Build modular and reusable components using HTML, CSS and JavaScript by Prateek Jadhwani.
2) Jump Start Bootstrap: Get Up to Speed With Bootstrap in a Weekend By Syed Fazle Rahman.
3) Fronted Web Development/Web Designing, HTML, CSS & JavaScript Basic Tutorial by Sachin Srivastav
4) Web Design and Development: Website Technologies Fundamentals By Steven Bright.

Reference Books

1) HTML and C Learn HTML, CSS, and JavaScript and Build a Website, App, and Game by Young Rewired State and Duncan Beedie.
2) Mastering HTML, CSS & Javascript Web Publishing by Laura Lemay, Rafe Colburn HTML & CSS, and JavaScript & JQuery (2 book set) by Jon Duckett.

List of Laboratory Exercise:

1) Design home page for any website according to domain.
2) Implement various functionality using different tags of HTML while designing web pages.
3) Implement web pages formatting and content formatting using CSS.
4) Implement responsive approach in website designing
5) Explorer front end framework using Bootstrap Elements
6) Demonstrate website design using content management system.

Project Based Learning Assignments

Note:- *Students in a group of 3 to 4 shall complete any one project from the following list

1. Design website for department and college
2. Design website for e-commerce platform.
3. Design website for reservation system (eg. bus, train, air)
4. Design website for online food delivery system.
5. Design website for CRM (database management).
6. Design website for hospital management system.
7. Design website for advertisement of products.
8. Design website for customer support system.
9. Design website for Business Portfolio.
10. Design website for Quiz Game.
11. Design website for E-library system.
12. Design website for survey system.
13. Design website for Banking system.
14. Design website for social media.
15. Design matrimonial website.

B.Tech(Information Technology)

Semester-III

Discrete Structures and Graph Theory

TEACHING SCHEME		EXAMINATION SCHEME		CREDITSCHEME	
Lecture:	**4 Hours/Week**	**End Semester Examination:**	**60 Marks**	**Theory**	**4**
Practical	**2 Hours/Week**	**Internal Assessment:**	**40 Marks**	**Term Work**	**1**
		Term Work:	**25 Marks**		
Total	**06 Hours/Week**		**125 Marks**		**5**

Course Objectives:

1. To apply and relate knowledge of mathematics in computer science.
2. To learn proof theory with propositional calculus and induction.
3. To map, represent and solve network problem with trees and graphs.

Prerequisite:

Basic mathematics and programming fundamentals.

Course Outcomes: On completion of the course, students will have the ability to

1. Formulate real world problems into statement forms using sets and relations which
2. can be solved or proved mathematically using set theory and logic.
2. Design mathematical model from theoretical statements.
   Apply counting techniques to real world problems.
4. Apply knowledge of graphs to solve network problems.
5. Design searching algorithm efficiently by applying tree and tree traversal logic.
6. Apply algebraic structure and coding theory in computer science.

Unit I **08 Hours**

Propositional Logic and Proof Theory: Sets, Set operations, Finite and Infinite sets, Venn diagram, Principle of inclusion and exclusion, Multisets. Propositions, Conditional Propositions, Logical Connectivity, Propositional calculus, Universal and Existential Quantifiers, Normal forms, methods of proofs, Principal of mathematical induction.

Unit II: **08 Hours**

Relations and Functions: Properties of Binary Relations, Closure of relations, Warshall's algorithm, Equivalence, Relations and partitions, Partial ordering relations and lattices, Chains and Anti chains. Functions, Composition of functions, Invertible functions, Pigeonhole Principle.

Unit III **08 Hours**

Counting and Recurrence Relations Basic counting principles, permutations, combinations, generalized permutations and combinations (with/without repetitions), Probability theory, Permutations with indistinguishable objects, Binomial coefficients, and identities. Linear Recurrence Relations with constant Coefficients, Homogeneous Solutions, Total solutions.

Unit IV **08 Hours**

Graph theory: Basic terminology, multi graphs and weighted graphs, paths and circuits,shortest path in weighted graph, Dijkstra's algorithm, Hamiltonian and Euler paths and circuits, factors of a graph, planer graph and Travelling salesman problem.

Unit V **08 Hours**

Trees: Trees, rooted trees, path length in rooted trees, prefix codes, binary search trees,tree traversal, spanning trees and cut set, minimal spanning trees, Kruskal's and Prim's algorithms for minimal Spanning tree. The Max flow- Min Cut Theorem (Transport network). Case Study- Game Tree, Mini-Max Tree.

Unit VI **08 Hours**

Algebraic Structures: The structure of algebra, Algebraic Systems, Semi Groups, Monoids, Groups, Homomorphism and Normal Subgroups, Congruence relations, Rings, IntegralDomains and Fields, coding theory, Polynomial Rings and polynomial Codes.

List of Internal Assignment will be framed by respective Course Coordinator.

Textbooks:

1. Kenneth H. Rosen, Discrete Mathematics and its Applications, 7th Edition, McGraw Hill.
2. C. L. Liu, D. P. Mohapatra, Elements of Discrete Mathematics: A Computer Oriented Approach, 4th Edition, McGraw Hill.

Reference Books

1. Seymour Lipzschutz, M. Lipson, Discrete Mathematics, 3rd Edition, McGraw Hill.
2. P. Tremblay, R. Manohar, Discrete Mathematical Structures With Applications to Computer Science, McGraw Hill.

List of Laboratory Exercise

1. Write a program to implement set operations. (Set size and elements to be taken from user at runtime).
2. Write a program to calculate value of polynomial for variable x. (Highest degree and coefficients to be taken from user at runtime).
3. Write a program to find value of composite function: fogoh. (f(x), (g(x) and h(x) tobe taken from user.)
4. Write a program to implement Warshall's algorithm.
5. Write a program to check whether Eulerian circuit is present in the given graph.
6. Write a program to find shortest path between the vertices in given graph.
7. Write a program to create binary search tree for the values taken from user.
8. Write a program to implement various tree traversals.
9. Write a program to implement Kruskal's algorithm.
10. Write a program to implement Prim's algorithm.

Project Based Learning Assignments

Note:- *Students in a group of 3 to 4 shall complete any one project from the following list

1. Study the writings of Lewis Carroll on symbolic logic. Describe in detail some of the models he used to represent logical arguments and the rules of inference he used in these arguments.
2. Describe a variety of different applications of the Fibonacci numbers to the biological and the physical sciences.

3. Explain how graph theory can help uncover networks of criminals or terrorists by studying relevant social and communication networks.
4. Explain what community structure is in a graph representing a network, such as a social network, a computer network, an information network, or a biological network. Define what a community in such a graph is, and explain what communities represent in graphs representing the types of networks listed.
5. Describe how Euler paths can be used to help determine DNA sequences.
6. Describe some of the strategies and algorithms used to solve the traveling salesperson problem.
7. Five men with different nationalities and with different jobs live in consecutive houses on a street. These houses are painted different colors. The men have different pets and have different favorite drinks. Determine who owns a zebra and whose favorite drink is mineral water (which is one of the favorite drinks) given these clues: The Englishman lives in the red house. The Spaniard owns a dog. The Japanese man is a painter. The Italian drinks tea. The Norwegian lives in the first house on the left. The green house is immediately to the right of the white one. The photographer breeds snails. The diplomat lives in the yellow house. Milk is drunk in the middle house. The owner of the green house drinks coffee. The Norwegian's house is next to the blue one. The violinist drinks orange juice. The fox is in a house next to that of the physician. The horse is in a house next to that of the diplomat.
8. Explain how graph multicolorings can be used in a variety of different models.
9. Define a heap and explain how trees can be turned into heaps. Why are heaps useful in sorting?
10. Describe the techniques used by chess-playing programs such as Deep Blue or stockfish.
11. Discuss the algorithms used in IP multicasting to avoid loops between routers.
12. Compare and contrast some of the most important sorting algorithms in terms of their complexity and when they are used.
13. Describe an algorithm for finding the minimum spanning tree of a graph such that the maximum degree of any vertex in the spanning tree does not exceed a fixed constant k
14. Describe the origins of mathematical induction. Who were the first people to use it and to which problems did they apply it?
15. Explain how the ideas and concepts of program correctness can be extended to prove that operating systems are secure.

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit – VI

Data Structures

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
Lecture:	**4 Hours/Week**	**End Semester Examination:**	**60 Marks**	**Theory**	**4**
Practical:	**2 Hours/Week**	**Internal Assessment:**	**40 Marks**	**Practical**	**1**
		Term Work and Practical:	**50 Marks**		
Total	**6 Hours/week**		**150 Marks**		**5**

Course Objective:

The objective of the course is to familiarize students with fundamentals of data structures and algorithms.

Prerequisite:

Fundamental knowledge programming and problem-solving steps

Course Outcomes: On completion of the course, students will have the ability to:

1. Understand the fundamentals of data structure and algorithms
2. Execute linear sequential data structures
3. Implement linear linked organization data structures
4. Execute nonlinear data structure-trees
5. Implement nonlinear data structure-graph
6. Know hashing and file organization concepts

Unit I **08 Hours**

Introduction to Algorithm and Data Structures: Introduction to data structures, types of data structure, abstract data types (ADT), introduction to algorithms, characteristics of algorithms, algorithm design tools: pseudo code and flowchart, relationship among data, data structure and algorithms, analysis of algorithms, asymptotic notation.

Unit II **08 Hours**

Sequential Organization Data Structures: Stacks: primitive operations, stack as an ADT, realization of stacks using array,stack operations, multi-stack, applications of stack, expression evaluation and conversion, simulating recursion using stack

Queue: primitive operations, queues as ADT, realization of queue using array, circular queue, double ended queue, priority queue, applications of queue.

Unit III **08 Hours**

Linked Organization Data Structures: Introduction, comparison of sequential and linked organizations, comparison of static and dynamic memory allocation, realization of linked lists, dynamic memory management, linked list as ADT, types of linked list, polynomial manipulations, linked stack, linked queue, generalized linked list (GLL) concept, applications oflink list.

Unit IV **08 Hours**

Non-Linear Data Structure-Tree: Tree terminology, types of trees, binary tree as an ADT, realization of tree, tree traversals, binary search tree, operations on BST, threaded binary tree, AVL tree,heap tree, applications of trees.

Unit V **08 Hours**

Non-Linear Data Structure-Graph: Graph terminologies, graph as an ADT, realization of graphs using adjacency matrix and adjacency list, graph traversals: breadth first search traversal, depth first search traversal, spanning tree, prim's and kruskal's algorithms, topological sorting, applications of graph

UNIT VI **08 Hours**

Hashing and File Organization:

Hashing: introduction, key terms, hash function, Collision Resolution strategies,hash table overflow, skip list, comparison of hashing and skip lists.

File: concept of file, file organization, sequential file organization, direct access file organization, indexed sequential file organization.

List of Internal Assignment will be framed by respective Course Coordinator.

Textbooks:

1. Y. Langsam, M. Augenstin, A. Tannenbaum, "Data Structures using C and C++", Prentice Hall of India, , ISBN-81-203-1177-9.
2. E. Horowitz, S. Sahni, D. Mehta, "Fundamentals of Data Structures in C++", Galgotia Book Source, New Delhi, ISBN 16782928
3. S. Lipschutz, "Data Structures", McGraw Hill Pub.
4. Patil V., "Data Structures using C++", Oxford university press, ISBN 0-19-806623-6
5. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein,"Introduction to Algorithms"

Reference Books

1. G. A.V, PAI , "Data Structures and Algorithms ", McGraw Hill, ISBN -13: 978-0-07-066726-6
2. M. Welss, "Data Structures and Algorithm Analysis in C++", Pearson Education, ISBN-81-7808-670-0

List of Laboratory Exercise

1. Write a program to implement functions (insert, delete, display) on stack, queue and circular queue data structure.
2. Write a program to convert and solve expression from
3. (a)Infix to Prefix
   (b) Infix to Postfix

Evaluate Postfix expression

3. Write a program to implement Singly Linked List manipulation for storing student information (PRN, Name, Marks).
   a. Display data of top rank student.

How many students secure first class and above rank?

4. Write a program to implement Doubly Linked List manipulation for storing Employee information (Name, Salary, Age).
   a. Display data of employees having salary more than 50,000.

Display list of employees having age less than 30 and salary greater than 30,000.

5. Write a program to implement Binary Search Tree storing city names and Traversal in BST (Inorder, Preorder, Postorder).
6. Write a program to implement Threaded Binary Tree and its Traversals.
7. Write a program to implement graph traversals: BFS and DFS.
8. Write a program to implement Prims and Krushkals algorithms MST.

Project Based Learning Assignments

Note:- *Students in a group of 3 to 4 shall complete any one project from the following list

1. Design and develop a project for Election System
2. Design and develop a project for Flight ticket booking
3. Design and develop a project for Tourism Management system
4. Design and develop a project for Simple Result system
5. Create a mini project to construct game: Tic-Tac-Toe
6. Design and develop a project for Phone Directory using doubly link list
7. Create a mini project to construct game: Snakes and Ladder
8. School fee enquiry Management System

9. Telecom Billing Management System

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Database Management System

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
Lecture:	**3 Hours/Week**	**End Semester Examination:**	**60 Marks**	**Theory**	**3**
Practical:	**2 Hours/Week**	**Internal Assessment:**	**40 Marks**		
		Term Work and Practical:	**50 Marks**	**Practical:**	**1**
Total	**5 Hours/week**		**150 Marks**		**4**

Course Objective

The objective of the course is to present an introduction to database management systems,with an emphasis on how to organize, maintain and retrieve - efficiently, and effectively - information. from a Database Management Systems.

Prerequisite:

Students should have knowledge of

1) Basic understanding of data and data structure.
2) Basic understanding of programming language.

Course Outcomes: On completion of the course, students will have the ability to:

1. Model an application's data requirements using conceptual modeling tools.
2. Implement concepts of relational algebra and SQL queries.
3. Demonstrate concepts of relational database design.
4. Interpret the query processing and optimization activities in database.
5. Interpret the transaction activities in database.
6. Recognize the emerging database applications and security concerns.

Unit I **06 Hours**

Introduction: Introduction to Database system architecture, Data Abstraction, Data Independence.

Data models: Extended Entity-relationship model, network model, relational and object oriented data models, integrity constraints, data manipulation operations.

Unit II **06 Hours**

Relational algebra: Fundamental and extended relational algebra operations, Tuple and domain relational calculus

Introduction to SQL: Data definition language, Data Manipulation Language, Joined relations, Views.

Introduction to PL/SQL: Functions, Procedures, Triggers, Cursors.

Unit III **06 Hours**

Integrity constraints: What are constraints, types of constrains

Relational database design: Domain and data dependency, Armstrong's axioms, Functional Dependencies, Normal forms (1NF, 2NF, 3NF, BCNF, 4NF), Dependency preservation, Lossless design.

Unit IV **06 Hours**

Storage strategies: Indices, B trees, B+ trees, Hashing

Query processing and optimization: Evaluation of relational algebra expressions, Query equivalence, Join strategies, Query optimization algorithms

Unit V **06 Hours**

Transaction processing: Concurrency control, ACID property, Serializability of scheduling, Locking and timestamp-based schedulers, Multi-version and optimistic Concurrency Control schemes, Database recovery.

Unit VI **06 Hours**

Data Intensive Computing: Introduction to big data, unstructured data processing using Hadoop, NoSQL database using MangoDB
Database Security: Authentication, Authorization and access control, DAC, MAC and RBAC models, Intrusion detection, SQL injection.

List of Internal Assignment will be framed by respective Course Coordinator.

Textbooks

1 Silberschatz, Korth, "Data base System Concepts", 7th ed., McGraw hill.
2 Raghu Ramakrishnan and Johannes Gehrke, "Database Management Systems" (3/e), McGraw Hill.
3 Ramez Elmasri and Shamkant B. Navathe, Fundamentals of Database Systems (5/e), Pearson Education.
4 C. J. Date, Kannan, "An Introduction to Database Systems", 8e, Addison-Wesley
5 Ivan Bayross, "SQL, PL/SQL the Programming Language of Oracle", BPB Publication

Reference Books

1 Peter Rob and Carlos Coronel, Database Systesm- Design, Implementation and Management (7/e), Cengage Learning, 2007
2 Hector Garcia-Molina, Jeff Ullman, and Jennifer Widom, "Database Systems: The Complete Book" (2nd edition), Pearson Prentice Hall

List of Laboratory Exercise

1. Conceptual Designing using ER Diagrams (Identifying entities, attributes, keys and relationships between entities, cardinalities, generalization, specialization etc).
2. Convert ER Model to Relational Model (Represent entities and relationships in Tabular form, represent attributes as columns, identifying keys).
3. Remove the redundancies and anomalies in the above relational Tables, Normalize up to Third Normal Form.
4. Study and implementation of SQL: DDL
   Creation of above Tables using SQL- Data types in SQL, Creating Tables (along with Primary and Foreign keys), Altering Tables and Dropping Tables.
5. Study and implementation of SQL: DML, Querying with set operations and wildcards
6. Study and implementation of aggregate functions, joins, nested subqueries in SQL from querying above tables.
7. Study and implementation of views in SQL.
8. Study and implementation of PL/SQL – Control statements.
9. Study and implementation of PL/SQL Functions and stored procedure.
10. Study and implementation of Triggers.
11. Study and implementation of Cursors.

Project Based Learning Assignments

Note:- *Students in a group of 3 to 4 shall complete any one project from the following list

1. Make a project to maintain employee data using files and dynamic object/structure. The project should be able to read, write, modify, add, and search records. Also demonstrate the effect of performing change in employer data definition after few records have been added.
2. Make an extended ER diagram for insurance management system. Transform this into relation design and implement these relations with appropriate domain and integrity constraints.
3. Employ various data control restrictions on databases, relations, and attributes of relations.
4. Create a phonebook which enables user to save contacts with additional information and

provides various retrieval mechanisms. Provisions should be made to view data in multiple ways.

5. Design and develop a library management system. The relations in the system should be normalized up to BCNF
6. Design and develop a inventory management system and create multiple views on the relations so that users not authorized to edit the relations should be able to views the data.
7. Implement of audit trails and backup on relations.
8. Create a student result calculation system. However, when updating results after calculation should be only of students who paid complete fees, such that transaction of each row is executed separately. Hint- use explicit cursor
9. Develop a student data management system using hash files.
10. Installation of a NoSQL database and implementing a simple student database to compare with SQL database.

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Software Engineering

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
Lecture:	**4 Hours/Week**	**End Semester Examination:**	**60 Marks**	**Theory:**	**4**
Practical:	**2 Hours/Week**	**Internal Assessment:**	**40 Marks**	**Term Work:**	**1**
		Term Work :	**25 Marks**		
Total	**6 Hours/Week**		**125 Marks**		**5**

Course Objective:

This course presents modern software engineering techniques and examines the software life-cycle, including software specification, design, implementation, testing. The course is organized as a project where the students work in a team to address a real-world software engineering assignment. The project is supplemented by exercises and lectures that provide insight into the assignment students are working on and software engineering in general.

Prerequisite: Programming knowledge

Course Outcomes: On completion of the course, students will have the ability to:

1. Compare various software development methods.
2. Identify requirements for project.
3. Apply software analysis principles.
4. State steps involved in software designing.
5. Show working of software engineering tools.
6. Execute a thorough software test.
7. Function effectively as a member of a team engaged in software engineering activities.

Unit I **08 Hours**

Introduction to Software Development: Software Development Challenges, Software Scope , Software Engineering Discipline, Software Methodologies and Related Process Models, The Human Side of Software Development , Traditional Life Cycle Models o Waterfall , Incremental Evolutionary, Spiral, CBSE, Alternative Process models: Unified Process, Rapid Application Development, Introduction to Agile Software Engineering Process Models: Extreme Programming o Agile Software Development, DevOps, Site Reliability Engineering. Quality and Process Standards: ISO 9000, SWEBOK, ISO 15504, SEI's Capability Maturity Model (CMM).

Unit II **08 Hours**

Requirement engineering: Requirements Development Methodology, Specifying Requirements, Eliciting Accurate Requirements, Documenting Business Requirements, Defining User Requirements, Validating Requirements, Achieving Requirements Traceability, Managing Changing Requirements, Reviews, Walkthroughs, and Inspections, Requirements Modelling, Agile Requirements Engineering.

Business Model Engineering: Business Model Capture Tools, Process Modelling, Capturing the Organization and Location Aspects, Developing a Process Model.

Unit IV **08 Hours**

System Design: Problem partitioning, abstraction, top-down and bottom-up design, Structured approach. Design Concepts, The Design model architecture, cohesion and coupling, Data Design, Architectural Styles and Patterns, Architectural Design, Mapping Data flow into Software Architecture. Functional versus object-oriented approach.

Coding: Programming languages and development tools Selecting languages and Tools, Good programming practices Coding Standards.

08 Hours

Unit V

Software configuration management (SCM): Elements of SCM, Base lines, Software configuration items, SCM Repository, SCM process:

Software Engineering Tools: Requirements Management Tools (e.g., IBM Rational Doors), Design Tools (e.g., Sparx Enterprise Architect), Development Tools o IDEs (e.g., Xcode, Eclipse, IntelliJ IDEA, NetBeans, Microsoft Visual Studio, Atom), Source Control Management (e.g., GitHub), Release Orchestration (e.g., Open Make), Collaboration (e.g., Jira, Trello, Slack).

08 Hours

Unit VI

Testing Strategies: Levels of Testing, Functional Testing, Structural Testing, Test Plan, Test Case Specification, Test case design, A strategic approach to software Testing: Verification and Validation Testing, organizing for software Testing, Software Testing Strategy for conventional Architecture: Unit Testing Integration Testing, Validation Testing, System Testing, Debugging, White-box, Black-box testing, Basis path Testing, Control structure testing.

List of Internal Assignment will be framed by respective Course Coordinator.

Textbooks

1. Roger Pressman, Software Engineering: A Practitioner's Approach, 6th edition, McGraw Hill, 2005. ISBN 0-07-285318-2
2. Somerville, Ian (2001) Addison-Wesley Software Engineering 7th Edition). Massachusetts: Addison Wesley, ISBN 0-321-21026-3
3. Fundamentals of Software Engineering by Rajib Mall

Reference Books

1. Kniberg, H. (2015) Scrum and XP from the Trenches - 2nd Edition,
2. Pro Git: http://git-scm.com/

List of Laboratory Exercise

1. Preparing Software Requirements Specifications
2. Performing domain analysis
3. Perform E-R Modeling
4. Perform Data-Flow-Modeling
5. Draw State Diagram
6. Designing Test Suites
7. Calculate cyclomatic Complexity for code snippet

Project Based Learning Assignments

Note:- *Students in a group of 3 to 4 shall complete any one project from the following list

1. Android task monitoring
2. Sentiment analysis for product rating
3. Fingerprint-based ATM system
4. Advanced employee management system
5. Image encryption using AES algorithm

6. Fingerprint voting system
7. Weather forecasting system
8. Android local train ticketing system
9. Railway tracking and arrival time prediction system
10. Android Patient Tracker
11. Opinion mining for social networking platforms
12. Automated payroll system with GPS tracking and image capture
13. Data leakage detection system
14. Credit card fraud detection
15. AI shopping system

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit – VI

Computer Communication and Networks

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
Lecture:	**3 Hours/Week**	**End Semester Examination:**	**60 Marks**	**Theory**	**3**
Practical:	**2 Hours/Week**	**Internal Assessment:**	**40 Marks**	**Practical:**	**1**
		Term Work and Practical:	**25 Marks**		
Total	**5 Hours/week**		**125 Marks**		**4**

Course Objectives:

1. Build an understanding of the fundamental concepts of computer networking.
2. This course will enable students to understand the layering architecture of OSI reference model and TCP/IP protocol suite, protocols associated with each layer.
3. Learn the different networking architectures and their representations and able tolearn the various routing techniques and the transport layer services.

Prerequisite:

Students should have knowledge of

1. How computer networks operate and the fundamentals of data communication.
2. Concepts and fundamental design principles of modern computer networking in a top-down approach, focusing on the Internet's architecture and protocols.

Course Outcomes: On completion of the course, students will have the ability to:

1. Find the components required to build different types of networks.
2. Recognize the different types of network Transmission Media and Technologies.
3. Explain the layered architecture of computer networks and distinguish between the OSI reference model and TCP/IP protocol suite.
4. Match the division of network functionalities with the layers.
5. Distinguish the basic network Layer services and Protocols associated with each network.
6. Identify the protocols and functions associated with the transport layer services.

Unit I **06 Hours**

Introduction to data communication and networking: Data Communications: Components, Representations, Data Flow. Digital Transmission: Analog-to-Digital Conversion, Digital-to-Digital Conversion. Analog Transmission: Digital-to-analog Conversion, Analog-to- Analog Conversion.

Networks: Physical Structures, Introduction to Networks – Building Network and Network Types: LAN, WAN, MAN and PAN, Overview of Topology, Concepts of Communication Modes and Transmission Modes. Categories of Networks Internet works.

Unit II **06 Hours**

Data Transmission Media and Technologies: Transmission Media: Types of transmission media, principal, Specification of Medium, Performance, and Transmission Impairments. Applications of different transmission media.

Introduction to switching: Switching, Circuit-switched Networks, Packet Switching, Datagram Switching and Datagram networks, Virtual circuit networks, Structure of circuit and packet switch.

Unit III **06 Hours**

Network Models: Protocol Layering: Scenarios, Principles, Logical Connections. Reference Models, Functions of the layers of The OSI Model, TCP/IP Protocol Suite: Layered Architecture, Layers in TCP/IP suite and its functioning, Description of layers, services, sockets and ports Encapsulation and D-encapsulation, Addressing, Multiplexing and De-multiplexing, Types of Multiplexing and Multiplexing applications. The OSI Model: OSI Versus TCP/IP.

Unit IV **06 Hours**

Networking Devices: Networking Devices: Hubs, Switch, Router, Repeaters, Bridges, Gateway, Modem and Access Point, Backbone networks.
Data-Link Layer:
Introduction: Nodes and Links, Services, Categories of link, Sub layers, Link Layer addressing: Types of addresses, ARP, RARP. Data Link Control (DLC) services: Framing, Flow and Error Control, Data Link Layer Protocols: Simple Protocol, Stop and Wait protocol, Piggybacking.

Unit V **06 Hours**

Network Layer: Introduction, Network Layer services: Packetizing, Routing and Forwarding, Other services. IPV4 Addresses: Address Space, Classful Addressing, Classless Addressing, DHCP, Network Address Resolution, Forwarding of IP Packets: Based on destination Address and Label. Overview of IPv6 Addressing – Transition from IPv4 to IPv6 Comparison of IPv4 and IPv6.
Network layer Protocols:
Internet Protocol (IP): Datagram Format, Fragmentation, Options, Security of IPv4 Datagrams, ICMPv4: Messages, Debugging Tools.
Routing: Introduction to Types of Routing, Routing Algorithms: Distance Vector Routing, Link State Routing, Path vector routing, Unicast Routing Protocol: Internet Structure, Routing Information Protocol, Open Shortest Path First, Border Gateway Protocol Version 4.

Unit VI **06 Hours**

Transport Layer: Introduction: Transport Layer Services, Connectionless and Connection oriented Protocols, Transport Layer Protocols: Simple protocol, Stop and wait protocol, Go- Back-N Protocol, Selective repeat protocol, User Datagram Protocol: User Datagram, UDP Services, UDP Applications, Transmission Control Protocol: TCP Services, TCP Features, Transmission Policy, Segment header, Connection, State Transition diagram, Windows in TCP, Flow control, Error control, TCP congestion control, Timer Management. Application Layer Paradigms – Client Server Programming – World Wide Web and HTTP, DNS, Electronic Mail (SMTP, POP3, IMAP, MIME).

List of Internal Assignment will be framed by respective Course Coordinator.

Textbooks:

1. Data Communications and Networking, Forouzan, 5th Edition, McGraw Hill, 2016 ISBN: 1-25-906475-3.
2. James F. Kurose, Keith W. Ross, ―Computer Networking - A Top-Down Approach Featuring the Internet, Seventh Edition, Pearson Education, 2016.

Reference Books:

1. Computer Networks, James J Kurose, Keith W Ross, Pearson Education, 2013,
2. Introductions to Data Communication and Networking, Wayarles Tomasi, Pearson Education,
3. Nader. F. Mir, ― Computer and Communication Networks, Pearson Prentice Hall Publishers,
4. 2nd Edition, 2014.
5. Ying-Dar Lin, Ren-Hung Hwang, Fred Baker, ―Computer Networks: An Open Source
6. Approach‖, Mc Graw Hill Publisher, 2011.
7. Larry L. Peterson, Bruce S. Davie, ―Computer Networks: A Systems Approach‖, Fifth Edition, Morgan Kaufmann Publishers, 2011.

List of Laboratory Exercise

1. Study and execution of Network commands.
2. Socket programming Client Server using RPC.
3. Demonstration of different types of cables used in data communication.
4. Perform various line coding formats and compare transmission characteristic of each formats.
5. Perform digital carrier modulation techniques used in wireless communication.
6. Study and demonstration of CISCO packet tracer with data transmission.
7. Study and demonstration of CISCO packet tracer with data loss.
8. Perform serial data communication between two data terminal equipment using optical link.
9. Perform Installation of LAN and troubleshooting of frequently occurred problems.
10. Create and test wireless sensor networks using zigbee.
11. To study various aspects of data communication by field visit at data centre.

Project Based Learning Assignments

Note:- *Students in a group of 3 to 4 shall complete any one project from the following list

1. Network Desktop Manager. Example Modules: Desktop Sharing, Desktop locking and unlocking, IP Port Scanning.
2. Analysis of IPv4/IPv6 protocols over 3G mobile networks
3. Network Traffic Monitoring & windows Remote Manager. Example Modules: Remote Desktop, Remote Chat, Monitoring
4. Learner's Interaction with Information and Communication Technologies.
5. Use of Information-Centric Networks in Revision Control Systems
6. TCP Performance in an EGPRS system
7. Real-Time Networking based Computer Ideas
8. An Internet Voting System Supporting User Privacy
9. Use of Information-Centric Networks in Revision Control Systems.
10. Networking and Security Projects
11. IP based Patient Monitoring System
12. Network Admission Control (NAC) Securing End Point Devices

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit – VI

Information Technology Laboratory - I

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
Practical:	**4 Hours/Week**	**Term Work and Practical:**	**50 Marks**	**Practical:**	**2**
Tutorials:	**1 Hour/Week**	**Term Work and Oral:**	**25 Marks**		
Total	**5 Hours/week**		**75 Marks**		**2**

Course Objective

1. Compute time and space complexity for a given program.
2. Demonstrate concepts OOPS using java
3. Solve specified requirement
4. Infer various approaches to decide the efficiency of the given approach.
5. Formulate a given problem by providing the proof of behavior of the given model.
6. Design an application using a platform-independent approach.

Prerequisite:

Basic understanding of Object-Oriented Programming language and logic to solve. given problem.

Course Outcomes: On completion of the course, students will have the ability to:

1. Design a solution to a given problem applying logic and features of the java language.
2. Develop their logical skill through various assignments and practicals.
3. Divide complex problem into subpart and then handle every part to achieve the Goal.
4. Model a solution to any real-world problem.
5. Analyze the significance of platform independence.
6. Design application using object-oriented norms.

Unit I **06 Hours**

Introduction to Java: 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 - if, switch, and loops like for, do-
while, while. Introduction to Objects and classes.

Unit II **06 Hours**

Classes and objects: Java Classes, Abstract classes Static classes Inner classes Packages, Wrapper classes. Interfaces This Super Access control, embedded style information Inheritance, Encapsulation, Polymorphism, Data Binding, data abstraction.

Unit III: **06 Hours**

String and Arrays: One dimensional Array, Multidimensional array, Array of an object, Introduction
to vector. String, StringBuilder, String Buffer, String methods, manipulations.

Unit IV **06 Hours**

Exception Handling: Checked exceptions, unchecked exceptions, and Errors, try-catch block, throws,User-defined exception – Throw, Common exception classes.

Unit V **06 Hours**

Threading and multithreading: Lifecycle of Thread, Basic functions of thread, multithreading, synchronization.

Unit VI **06 Hours**

Collections and Generics: Introduction to collection framework, List, Set, Maps, utility class, Reflection API, Generics.

Textbooks

1 OCA Java SE 8 Programmer I Study Guide (Exam 1Z0-808) (Oracle Press) 3rd Edition. by Edward Finegan, Robert Liguori.
2 OCA Java SE 8 Programmer, Exam Guide (Exams 1Z0-808) 1st Edition, Kathy Sierra, Bert Bates.
3 Programmer's Guide to Java SE 8 Oracle Certified Associate (OCA), Khalid A. Mughal and Rolf W Rasmussen.

Reference Books

1 Headfirst Java, 2nd Edition by Kathy Sierra, Bert Bates.
2 Java: The Complete Reference, Eleventh Edition 11th Edition, Herbert Schildt.
3 OCAJP Associate Java 8 Programmer Certification Fundamentals: 1Z0-808, Hanmant Deshmukh.

List of Laboratory Exercise

1. Maintain record of students and perform CRUD functionality.
2. Write a program to redirect a request using a dynamic approach.
3. Write a program to pass the data using session.
4. Write a servlet to remove spam.
5. Maintain the record of faculty member using jsp action tags and directives.
6. Design a tag to perform the necessary editing in a given report.
7. Design reusable components of the form using taglib.
8. Implement sending and receiving mail utility using Java Mail API.
9. Implement Java Message Service queue.
10. Understand working of framework – struts- case study.

Project Based Learning Assignments

Note:- *Students in a group of 3 to 4 shall complete any one project from the following list

1. Implement assignment and project submission system.
2. Implement a program to issue Leaving Certificate, Transcripts and Bonafede certificate to student.
3. Implement a program to assign problem statement for practical examination in secured environment.
4. Design a template for NBA report.
5. Design an application for Feedback Management System.
6. Design application to maintain track of research paper with indexing per year.
7. Design a post customized as per social media platform.
8. Design an interface to collect job opportunities and disseminate to eligible student
9. Design a project to track details of Industrial Training.
10. Design notice board application to communicate with students.

B.Tech(Information Technology)

Semester-IV

IT Infrastructure Management

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
Lecture:	**4 Hours/Week**	**End Semester Examination:**	**60 Marks**	**Theory**	**4**
Practical:	**2 Hours/Week**	**Internal Assessment:**	**40 Marks**	**Practical**	**1**
		Term Work			
Total-	**6 Hours/Week**		**125 Marks**		**5**

Course Objectives:

Students undergoing this course are expected.

1. To introduce basic postulates of IT Infrastructure Management and shows the correlation
2. between system and service management process
3. Able to Know the Storage and database Management in Information Technology.
4. Infer various approach to decide efficiency of given approach.
5. Able to know the Security Management in IT.
6. To provide detailed knowledge of IT recent trends in globally.

Prerequisite:

Object Oriented Programming language and Logic to solve given problem.

Course Outcomes: On completion of the course, students will have the ability to:

1. Outline IT Infrastructure, management challenges and requirement.
2. Select Service Delivery and Service Support Processes required in IT infrastructure management.
3. Breakdown complex problem into subpart and then handle every part to achieve the goal.
4. Categorize various storage levels in IT.
5. Select security techniques in information technology.
6. Explain new communication mechanism based on emerging trends in information technology.

Unit I **08 Hours**

Introduction & It Infrastructure: Information Technology, IT Infrastructure Management, Introduction—IT Infrastructure Management, Challenges in IT Infrastructure Management, Design Issues of IT Organizations and IT Infrastructure, Determining Customers' Requirements, IT Systems Management Process, IT Service Management Process, Information System Design Process

Unit II **08 Hours**

Service Delivery Process & Service Support Process: Service Level Management, Financial Management, IT Service Continuity Management, Capacity Management, Availability Management, Configuration Management, Incident Management, Problem Management, Change Management.

Unit III **08 Hours**

Storage Management: Introduction to Storage, Backup and Storage, Archive, Retrieve, Disaster Recovery, Space Management, Database and application Protection, Bare, Machine Recovery (BMR), Data Retention

Unit IV **08 Hours**

Security Management: Computer Security, Internet Security, Physical Security, Identity Management, Access Control System, Intrusion Detection, Intellectual Property.

Unit V **08 Hours**

IT Ethics: Introduction to Cyber Ethics, Intellectual Property, Privacy and Law, Computer Forensics, Ethics and Internet, Cyber Crimes.

Unit VI **08 Hours**

Emerging Trends in It: Introduction, Electronic Data Interchange, Infrared Technology, Bluetooth, GSM, WiFi, Standards of Wifi, WiMax, 5G Wireless Technology.

List of Internal Assignment will be framed by respective Course Coordinator.

Textbooks

1. Gupta, "IT Infrastructure & Its Management", First Edition, Tata McGraw-Hill Education
2. IT Infrastructure and Management by Manoj Kumar Choubey - Published by Pearson Education

Reference Books

1. Firewalls for dummies, Brain Komar, Ronald Beekelaar, Joern Wettern, for Firewall Security, 70-662 MCTS exchage 2010 microsoft press.

List of Laboratory Exercise

1. Enlist and Illustrate Design Issues of IT Organisations and IT Infrastructure.
2. Demonstrate IT Service Continuity Management and Change Management.
3. Design and Implement various Storage Management and Recovery techniques.
4. Summarize different Security Management policies with assistance of Intellectual Property.
5. Setup and maintenance of Storage – Archive, Retrieve, Backup policies.
6. Configuration and Customization of Access Control List and Active Directory.
7. Discriminate various privacy and Cyber Laws with suitable example.
8. Demonstrate different internet security policies with suitable example.
9. Discover different Problem Management within Service Delivery Process.
10. Case Study- Disaster Recovery within Storage Management.

Project Based Learning Assignments

Note:- *Students in a group of 3 to 4 shall complete any one project from the following list

1. Develop Infrastructure Management System project for Server Management and Maintenance.
2. Develop Infrastructure Management System project for Software Management and Document Management
3. Secure File Storage on Local Machine Using Hybrid Physical Security Techniques.
4. Design security management for New Data center setup
5. Infrastructure Management System project for Monitoring of Bandwidth.
6. Tracking System for Defects. (For Example: Bug tracking and error system based on the web)
7. Implement Secure Backup Software System.
8. Design system for Detecting Data Leaks within storage management.
9. Implement enterprise management of electronic data interchange systems. (For example: Process mining, Good Security Practice)
10. Develop system for Bluetooth Controlled Electronic Home Appliances.

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit – VI

Formal Languages and Computation Theory

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
Lecture:	**4 Hours/Week**	**End Semester Examination:**	**60 Marks**	**Theory**	**4**
Practical:	**2 Hours/Week**	**Internal Assessment:**	**40 Marks**	**Practical**	**1**
		Term Work:	**25 Marks**		
Total	**6 Hours/Week**		**125 Marks**		**5**

Course Objective:

Students will learn about a variety of issues in the mathematical development of computer. science theory, particularly finite representations for languages and machines. Students will gain more formal understanding of algorithms and procedures.

Prerequisite:

Students should have knowledge of set theory and state transition diagrams.

Course Outcomes: On completion of the course, students will have the ability to

1.Design automata machines for strings given.
2.Write a regular expression for the given string and find set of strings if regular expression is given.
3.Write grammar rules for the strings given.
4.Design push down automata for the string and grammar.
5.Design Turing machine and apply the same to solve algorithmic problems.
6.Apply knowledge computation in complexity theory.

Unit I : **08 Hours**

Finite Automata: Introduction to Finite Automata, Structural Representations, Deterministic finite Automata (DFA)-Formal Definition, Simplified notation: State transition graph, Transition table, Language of DFA, Nondeterministic finite Automata (NFA), NFA with epsilon transition, Language of NFA, Equivalence and Minimization of Automata, Conversion of NFA with epsilon to DFA Equivalence of Moore and Mealy Machine. Applications and Limitation of FA.

Unit II **08 Hours**

Regular expressions: Regular expression (RE), Definition, Operators of regular expression and their precedence, Algebraic laws for Regular expressions, Kleen's Theorem, Regular expression to DFA, DFA to Regular expression, Non-Regular Languages, Pumping Lemma for regular Languages, Closure properties of Regular Languages, Applications of regular expressions.

Unit III **08 Hours**

Grammar: Definition, Production rules, Derivation trees, Ambiguous Grammar, Removal of ambiguity, Regular Grammar, Inter-conversion between RE and Grammar, Reduced form of grammar. Linear grammar: left & right linear grammar, Inter- conversion. Chomsky hierarchy of languages, Context Free Grammar- Definition, Context free language (CFL. Normal Forms- Chomsky Normal Form (CNF), Griebach Normal Form (GNF).

Unit IV **08 Hours**

Push Down Automata (PDA): Limitations of FA, PDA: Definition, Uses, Equivalence between FA and PDA, Designing of PDA, Deterministic Push Down Automata and Non-Deterministic Push Down Automata- Definition, Language accepted by PDA, Designing a PDA for CFG, Properties of CFL, Pumping Lemma for CFL. Limitations of PDA, Applications of PDA.

Unit V **08 Hours**

Turing Machine (TM): Definition, Model, Comparison of TM, FSM, PDA, Design of TM, Examples of TM- Combinational TM, Iterative TM, Recursive TM, Universal TM, TM as a language acceptor, Some Problems that cannot be solved by Turning Machines, Language accepted by TM, Recursive sets, partially recursive functions. Church's Turing hypothesis, Multitask TM, TM limitations.

Unit VI **08 Hours**

Computational Complexity: Decidable problems concerning regular languages, Decidable problems concerning context-free languages, Un-decidability, Halting Problem of TM, Reducibility: Un-decidable Problems from Language Theory, A Simple Un-decidable Problem PCP, Mapping Reducibility Time Complexity: Measuring Complexity, The Class P, Examples of problems in P, The Class NP, NP- completeness.

List of Internal Assignment will be framed by respective Course Coordinator.

Textbooks

1. "Introduction to Automata Theory, Languages and Computation", Hopcroft J, Motwani R, Ullman ,Addison-Wesley, ISBN 81-7808-347-7, Third Edition .
2. "Introduction to Theory of Computation", Michael Sipser, Course Technology, ISBN-10: 053494728X, Forth Edition. ISE.

Reference Books

1. "Introduction to Languages and Theory of Computation", John Martin. Fifth Edition, McGrawHill.
2. "Computational Complexity", Christos H. Papadimitriou, Pearson Education.

List of Laboratory Exercise

1. Solve problems on designing finite automata.
2. Design and inter-convert Moore and Mealy Machine for same problems.
3. Form grammar rules for language of set of regular expression or strings given.
4. Design Push Down Automata for grammar or given string.
5. Construct Turing Machine to solve given problem.
6. Study Assignment on Complexity Theory.

Project Based Learning Assignments

Note:- *Students in a group of 3 to 4 shall complete any one project from the following list

1. Describe the process of designing the computer. How is it related with the simple automata?
2. Write project based on famous computer scientist Alan Turing. Select suitable material for reference and summarize.
3. Describe the set of problems which can be represented using machines. What are the criteria we can enlist for such representations?

4. Relate the computational theory to World War II. What is the role of cryptography in World War II?
5. Invention of computer as a machine is related to formal automata. How today's complex and high-end computer systems can be mapped to these simple automata. Describe in detail.
6. Select a real-world problem and represent it mathematically. Design an automaton to solve this problem. Write detailed explanation of the entire process.
7. Study any text editor. Enlist its features. Map these features with the concepts you learned in the subject.
8. Enlist set of problems which can be solved, and which cannot be solved by memoryless automata. How memory affects the power of automata? Explain in detail and justify your answer with example.
9. Why Ethereum blockchain must be deterministic? Study and explain application of computation theory to blockchain technology.
10. Can human brain be simulated by Tring machine? Write detailed essay and justify your conclusions with theorem you learned.
11. Study research paper published by Alan Turing and write a summary in your words.
12. What are the similarities and differences between human brain and machine? Support your answers with suitable mathematical model.
13. Study any chess game software. Write the process of developing such software. Describe how this is related to Turing machine.

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit – VI

Microprocessors and Microcontrollers

TEACHING SCHEME	EXAMINATION SCHEME	CREDIT SCHEME	
Lecture: 3 Hours/Week	**End Semester Examination: 60 Marks**	**Theory**	**3**
Practical: 2 Hours/Week	**Internal Assessment: 40 Marks**	**Practical**	**1**
	Term Work and Practical: 50 Marks		
Total 5 Hours/Week	**150 Marks**		**4**

Course Objective

This course facilitates the learners with the basic knowledge of microprocessors and microcontrollers. Also, the course supports the learners with detailed study of ARM processor and AVR Microcontroller.

Prerequisite:

Digital Electronics, C/C++/Java Programming

Course Outcomes: On completion of the course, students will have the ability to:

1. Learn basics of 16/32-bit Microprocessors.
2. Cognize the ARM Cortex Processor with its architecture and programming.
3. Discover Intel Pentium and i7 processor with its architecture and pipelining.
4. Comprehend basics of 8/16-bit Microcontrollers.
5. Uncover the details of AVR Microcontroller with its architecture and programming.
6. Understand the basics of Arduino and Raspberry Pi Controllers.

Unit I: **06 Hours**

Introduction to Microprocessors: Basics of 16-bit and 32-bit processor (Intel 8086 and 80386 processors), Multicore Architecture, Hyperthreading Technology, Instruction Set Architectures (ISA), Multiprocessor Organizations, Inter-Processor Communication (IPC).

Unit II: **06 Hours**

Intel Pentium Processor: Features and Internal Architecture, Superscalar Operation, Integer &Floating- Point Pipeline Stages, Branch Prediction Logic, Cache Organization and MESI Protocol, Comparative study of 8086, 80386, Pentium I, Pentium II and Pentium III, Hyper Threading technology and its use in Pentium 4, Intel i7 processor: Features, Architecture, Memory System, Pipelining.

Unit III **06 Hours**

ARM Cortex: ARM Micro-architecture (ARMv7/v8/v9/v11), ARM architectures: Generic Interrupt Controller (GIC), Server Base System Architectures, Trusted Base System Architecture (TBSA), System Memory Management Unit (SMMU), Pipelining, ARM OS, ARM Programming.

Unit IV: **06 Hours**

Introduction to Microcontrollers: Microprocessors vs Microcontrollers, Basics of 8-bit and 16-bit Microcontrollers (Intel 8051 and 8096 microcontrollers), Applications of microcontrollers.

Unit V **06 Hours**

AVR Microcontroller: Types of AVR Microcontrollers, ATmega16/32 8-bit AVR microcontroller: Features, Pin Description, Internal Architecture, Data and Program Memory, AVR Programming using C/Java/Assembly language, Study of VR Studio/Amtel, Studio 7, Visual Micro Lab.

Unit VI **06 Hours**

Introduction to Arduino and Raspberry Pi : Introduction, Difference, Arduino Uno and Raspberry Pi Pico (RP2040), microcontrollers, Programming concepts of Arduino Uno with C/C++/Python and IDE, Programming concepts of Raspberry Pi Pico with C/MicroPython.

Textbooks

1 Arm Microprocessor Systems Cortex-M Architecture Programming and Interfacing, Muhammad Tahir, T&F India.
2 The Definitive Guide to ARM Cortex-M3 and Cortex-M4 Processors, Joseph Yiu.
3 ARM A32 Assembly Language, Bruce Smith.
3 8051 Microcontrollers, Satish Shah, Oxford University Press.
4 Microprocessors and Interfacing, N.Senthil Kumar, M.Saravanan, Oxford University Press
5 Programming and Interfacing Atmel AVR Microcontrollers, Grace, Cengage Learning.
6 Practical AVR Microcontrollers, Alan Trevennor, Technology In Action.
7 Getting Started with Arduino, Massimo Banzi and Michael Shiloh.
8 Getting Started with Raspberry Pi, Matt Richardson and Shawn Wallace.

Reference Books

1 The Definitive Guide to ARM Cortex-M3 Processors, Stellaris, Texas Instruments.
2 ARM System-on-Chip Architecture, Steve Furber.
3 ARM processor, Santul Bisht, Lambert Publications
4 Modern Assembly Language Programming with the ARM Processor, Larry D Pyeatt.
5 Programming and Customizing AVR Microcontroller, Dhananjay Gadre.
6 Arduino Cookbook 2nd Edition, Michael Margolis.
7 Raspberry Pi The Ultimate Guide, Geoff Adams.
8 Internet of Things with Raspberry Pi and Arduino, Anita Gehlot.

List of Laboratory Exercise

1) Programming Assignments based on ARM Processor (Minimum 3) using Assembly Language.
2) Programming Assignments on 8051 using C (Minimum 2).
3) Programming Assignments based on AVR Controller (Minimum 3) on AVR Assembly language or Embedded C.
4) Study of and Using VR Studio/Amtel Studio 6/ Visual Micro Lab.
   Simple programming assignments on Arduino and Raspberry Pi Controllers (1 each) :
   Arduino Uno programs on Arduino Desktop IDE or Web IDE using,
   a) Assembly/C/Python/Atmel Studio7
   b) Raspberry Pi, Pico programs on C/C++/MicroPython.

Project Based Learning Assignments

Note:- *Students in a group of 3 to 4 shall complete any one project from the following list

1. Design and Implementation of Drunken People Identification with Auto Ignition Disable Function using ARM (Processor/Controller)/AVR Controller/8051 Microcontroller
2. Design and Implementation of Automatic Turn off for Water Pump with Four Different Time Slots using ARM (Processor/Controller)/AVR Controller/8051 Microcontroller
3. Design and Implementation of Gas Leak Detector with Automatic Air Exhaust Using ARM Cortex
4. Design and Implementation of ARM Based Liquid Level Detection & Flow Control
5. Design and Implementation of Motion Based Door Opener (in malls, big shops) using ARM (Processor/Controller)/AVR Controller/8051 Microcontroller

6. Design and Implementation of Fire Detection and Alarm using ARM (Processor/Controller)/AVR Controller/8051 Microcontroller
7. Design and Implementation of Remote-Control Plant Watering System using ARM (Processor/Controller)/AVR Controller/8051 Microcontroller
8. Design and Implementation of Voice Controlled Air Purifier based on Arduino and Raspberry Pi
9. Design and Implementation of Face Recognition Door Lock System based on Arduino and Raspberry Pi
10. Design and Implementation of Vehicle Number Plate Recognition based on Arduino and Raspberry Pi

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit – VI

Applied Algorithms

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
Lecture:	**4 Hours/Week**	**End Semester Examination:**	**60 Marks**	**Theory**	**4**
Practical:	**2 Hours/Week**	**Internal Assessment:**	**40 Marks**		
		Term Work:	**25 Marks**	**Practical**	**1**
Total	**6 Hours/Week**		**125 Marks**		**5**

Course Objective:
Understand and compare important algorithmic design paradigms and analysis of algorithms. To choose and extend efficient algorithms required for designs.

Prerequisite:
Students should be well versed with algorithms and operations on basic data structures stacks, queues, linked lists, trees, graphs. Students should have knowledge of searching sorting algorithms.

Course Outcomes: On completion of the course, students will have the ability to:
1. Interpret the performance of algorithms using analysis techniques.
2. Examine the fundamental algorithmic strategies.
3. Compare the fundamental algorithmic strategies.
4. Implement graphs and trees algorithms.
5. Interpret the tractable or intractable problem.
6. Summarize the advance types of algorithms.

Unit I **08 Hours**

Introduction to Algorithm analysis: Characteristics of Algorithm. Analysis of Algorithm: Asymptotic analysis of Complexity Bounds – Best, Average and Worst-Case behavior, Performance Measurements of Algorithm, Time and Space Trade-Offs. Analysis of Recursive Algorithms through Recurrence Relations: Substitution Method, Recursion Tree Method and Masters' Theorem.

Unit II **08 Hours**

Algorithmic Strategies 1: Brute-Force technique, Heuristics, Greedy algorithms, Divide and Conquer, Illustrations of these techniques for Problem-Solving.

Unit III **08 Hours**

Algorithmic Strategies 2: Dynamic Programming, Branch and Bound algorithms, Backtracking, methodologies; Illustrations of these techniques for Problem-Solving.

Unit IV **08 Hours**

Graph and Tree Algorithms: Self-Balancing tress, B Trees, B+ Trees, Single source shortest path algorithms, all pair shortest path algorithms, Network Flow Algorithm

Unit V **08 Hours**

Tractable and Intractable Problems: Computability of Algorithms, Computability classes – P, NP, NP-complete andNP-hard. Cook's theorem, Standard NP-complete problems and Reduction techniques.

Unit VI **08 Hours**

Advanced Topics: Approximation algorithms, Randomized algorithms, Class of problems beyond NP – P SPACE, Introduction to Quantum Algorithms and parallel algorithms.

List of Internal Assignment will be framed by respective Course Coordinator.

Textbooks

1. "Fundamental of Computer Algorithms", E. Horowitz and S. Sahni, Orient Black.
2. "Introduction to Algorithms", T. H. Cormen, C. E. Leiserson and R. L. Rivest,PHI Learning Pvt. Ltd. (Originally MIT Press).
3. "The Design and Analysis of Computer Algorithms", A. Aho, J. Hopcroft and J. Ullman, Pearson Education India.
4. Computer Algorithms: Introduction to Design and Analysis, S. Baase, Pearson Education India.
5. "The Art of Computer Programming", D. E. Knuth, Addison Wesley.

Reference Books

1. M. Welss, "Data Structures and Algorithm Analysis in C++", Pearson Education, ISBN- 81-7808-670-0.
2. G. A.V, PAI , "Data Structures and Algorithms ", McGraw Hill, ISBN -13: 978-0-07-066726-6.

List of Laboratory Exercise

1. From a given vertex in a weighted connected graph, find shortest paths to other vertices using Dijkstra's algorithm.
2. Write a Code to find the shortest path using Bellman-Ford algorithm.
3. Write and analyze code to sort an array of integers using merge sort.
4. Write and analyze to sort an array of integers using divide and conquer quick sort Method.
5. Write a program to implement Longest Common Subsequence problem using Dynamic Programming.
6. Write a program to Implement 0/1 Knapsack problem using Dynamic Programming.
7. Write a program to Implement N Queen's problem using Back Tracking.
8. Write a program to implement quick sort using randomize algorithm.
9. Write a program to implement network flow algorithm.

Project Based Learning Assignments

Note:- *Students in a group of 3 to 4 shall complete any one project from the following list

1. Design and develop a project for Search engine using data structures
2. Design and develop a project for Google form like application
3. Design and develop a project for shortest path calculation for travelling salesman problem
4. Design and develop a project for finding keywords from the paragraph
5. Design and develop a project for Customer Billing system
6. Design and develop a project for word dictionary using search tree concept
7. Design and develop a project for salary calculation of employees based on performance
8. Design and develop a project for password recovery system
9. Create a mini project to construct game: Create Sudoku

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit - VI

Operating System

TEACHING SCHEME		EXAMINATION SCHEME		CREDIT SCHEME	
Lecture:	**3 Hours/Week**	**End Semester Examination:**	**60 Marks**	**Theory:**	**3**
Practical:	**2 Hours/Week**	**Internal Assessment:**	**40 Marks**	**Practical:**	**1**
		Term Work and Practical:	**50 Marks**		
Total	**5 Hours/Week**		**150 Marks**		**4**

Course Objective:
The learning objective of this course is to introduce the internal operation of modern operating systems. The course will cover processes and threads, mutual exclusion, CPU scheduling, deadlock, memory management, and file systems.

Prerequisite:
Programming skills, elementary data structures, algorithms, and computer architecture.

Course Outcomes: On completion of the course, students will have the ability to:

1. Explain the services provided by the system calls.
2. Implement the scheduling algorithms like FCFS, SJF and priority scheduling.
3. Implement the memory allocation techniques like first fit, best fit and worst fit.
4. Explain practical implementation of the inter-process communication of the processes.
5. Implement the file system.
6. Explain the concept of the deadlock occurrence, avoidance and implementation of deadlock free condition.

Unit I **06 Hours**

Computer System Overview: Basic Elements, Instruction Execution, Interrupts, Memory Hierarchy, Cache Memory, Direct Memory Access, Multiprocessor and Multicore Organization. Operating system overview- objectives and functions, Evolution of Operating System. - Computer System Organization Operating System Structure and Operations- System Calls, System Programs, OS Generation and System Boot.

Unit II **06 Hours**

Process Management: Process Concept, Process states, Process control, Threads, Uni-processor Scheduling: Types of scheduling: Preemptive, Non preemptive, Scheduling algorithms: FCFS, SJF, RR, Priority, Thread Scheduling, Real Time Scheduling. System calls like ps, fork, join, exec family, wait.

Unit III **06 Hours**

Memory Management: Memory Management requirements, Memory partitioning: Fixed and Variable Partitioning, Memory Allocation: Allocation Strategies (First Fit, Best Fit, and Worst Fit), Fragmentation, Swapping, and Paging. Segmentation, Demand paging Virtual Memory: Concepts, management of VM, Page Replacement Policies (FIFO, LRU, Optimal, Other Strategies), Thrashing.

Unit IV **06 Hours**

Inter Process Communication: Basic Concepts of Concurrency, Cooperating process, Advantage of Cooperating process, Bounded- Buffer - Shared-Memory Solution, Inter-process Communication (IPC), Basic Concepts of Inter-process Communication and Synchronization.

Unit V **06 Hours**

File Systems and I/O Systems : Mass Storage system – Overview of Mass Storage Structure, Disk Structure, Disk Scheduling and Management, swap space management; File-System Interface – File concept, Access methods, Directory Structure, Directory organization, File system mounting, File Sharing and Protection; File System Implementation- File System Structure, Directory implementation, Allocation Methods, Free Space Management, Efficiency and Performance, Recovery; I/O Systems – I/O Hardware, Application I/O interface, Kernel I/O subsystem, Streams, Performance.

Unit VI **06 Hours**

Concurrency control: Concurrency: Principles of Concurrency, Mutual Exclusion: S/W approaches, H/W Support, Semaphores, pipes, Message Passing, signals, Monitors, Classical Problems of Synchronization: Readers-Writers, Producer Consumer, and Dining Philosopher problem. Deadlock: Principles of deadlock, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, System calls like signal, kill.

Textbooks

1.Abraham Silberschatz, Peter B. Galvin, Greg Gagne, Operating System Concepts. Sixth edition. Addison-Wesley (2003).
2.Modern Operating Systems -By Andrew S. Tanenbaum (PHI).
3.Operating Systems 5th Edition, William Stallings, Pearson Education India.
4.Peterson and Silberschatz, Modern Operating Systems.
5. Harvey M. Deitel, An introduction to operating systems. Addison-Wesley.

List of Internal Assignment will be framed by respective Course Coordinator.

Reference Books

1. A.M. Lister, Fundamentals of Operating Systems. Macmillan (1979).
2 Andrew Tanenbaum & Albert Woodhull, Operating Systems: Design and Implementation. Prentice-Hall.

List of Laboratory Exercise

1. Basic Linux Commands and Overview.
2. Write Shell Script for finding the global complete path for any file.
3. Write Shell Script to broadcast a message to a specified user or a group of users logged on any terminal.
4. Write Shell Script to copy the file system from two directories to a new directory in such a way that only the latest file is copied in case there are common files in both the directories.
5. Write Shell Script to compare identically named files in two different directories and if they are same, copy one of them in a third directory.
6. Write Shell Script to delete zero sized files from a given directory (and all its sub-directories).
7. Implementation of FCFS (First Come First Serve) CPU Scheduling.
8. Implementation of SJF (Shortest Job First) CPU Scheduling.
9. Implementation of FIFO Replacement Algorithm.
10. Implementation of Optimal Page Replacement Algorithm.

Project Based Learning Assignments

Note:- *Students in a group of 3 to 4 shall complete any one project from the following list

1. Design of Intranet mail system project.
2. Design pf First-fit, worst-fit and best-fit for given allocation memory requirements

3. Simulation of the behavior of the multiprogramming operating system and use CPU scheduler, and CPU Execution.
4. Design the FCFS, SSTF, and SCAN disk-scheduling algorithms to simulate a simple disk drive, which has a specified number of logical blocks numbered from 0 onwards.
5. A Java simulator program to analyze the dependency of Page Faults on the Page Frames for incoming page requests.
6. CPU Scheduling Algorithm to calculate Throughput, Utilization, Turn Around time, Waiting Time. Gantt chart displayed for all n processes.
7. To simulate Round Robin algorithm.
8. A multi-threaded TCP server application, which allows multiple users to be registered and login.
9. Write a simple manual describing how to use the shell. The manual should contain enough detail for a beginner to UNIX to use it.
10. To simulate the dispatcher for allocating the process to CPU.

Syllabus for Unit Tests:

Unit Test -1	Unit – I, Unit – II, Unit - III
Unit Test -2	Unit – IV, Unit – V, Unit – VI

Information Technology Laboratory II

TEACHING SCHEME		EXAMINATION SCHEME	CREDIT SCHEME	
Practical:	**4 Hours/Week**	**Term Work and Practical: 50 Marks**	**Theory**	**Credits**
Tutorial:	**1 Hour/Week**	**Term Work and Oral: 25 Marks**	**Practical:**	**2**
			Tutorial	**1**
Total	**5 Hour/Week**	**75 Marks**		**3**

Course Objectives:

1) Understand web environment for building the application.
2) Implement web application.
3) Implement Servlet.
4) Implement Java Messaging Services.
5) Implement Java Mail API.

Prerequisite:

1) Core Java 2) Scripting languages.

Course Outcomes: On completion of the course, students will have the ability to:

1. Understand the lifecycle of web application.
2. Implement session management using servlet.
3. Apply standard and custom tags of JSP.
4. Design competitive web application which will work real web environment.
5. Implement Java Messaging Services.
6. Apply Java Mail API.

Unit I **06 Hours**

Introduction to Servlet: Web Application Basics, Architecture and challenges of Web, application. Introduction to servlet, Servlet life cycle, Developing and Deploying Servlets, Exploring Deployment Descriptor (web.xml).

Unit II **06 Hours**

Session Management and Servlet Chaining: 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 III **06 Hours**

Java Server Pages: Standard Tags: 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.

Unit IV **06 Hours**

Java Server Pages: Custom Tags: Tag Libraries, JSP Expression Language (EL), Using Custom Tag, JSP Capabilities Exception Handling Session Management Directives JSP with Java. Introduction to struts.

Unit V **06 Hours**

Java Messaging Services: JMS Architecture, Point-to-Point Messaging Domain, Publisher/Subscriber, Messaging Domain, JMS API, JMS Queue.

Unit VI **06 Hours**

Java Mail API: SMTP, POP, IMAP, MIME, NNTP, sending mail, receiving mail, mail with attachment, forward email, delete email.

Textbooks

1. Web Technologies: HTML, JAVASCRIPT, PHP, JAVA, JSP, ASP.NET, XML and Ajax, Black Book: HTML, JavaScript, PHP, Java, Jsp, XML and Ajax, Black Book Paperback – 1 January 2009, Kogent Learning Solutions Inc.
2. Java EE 8 Cookbook: Build reliable applications with the most robust and mature technology for enterprise development, Packt Publication, Elder Moraes.
3. Headfirst Servlets and JSP: Passing the Sun Certified Web Component Developer Exam 2nd Edition, Bryan Basham, Kathy Sierra, Bert Bates.

Reference Books

1. Beginning Java EE 7, Appress Publication, Antonio Goncalves.
2. Java EE 7 Essentials: Enterprise Developer Handbook 1st Edition, Headfirst Publication, Arun Gupta.
3. J2EE: The complete Reference Paperback, Jim Keogh.

List of Laboratory Exercise:

1. Maintain record of students and perform CRUD functionality.
2. Write a program to redirect a request using a dynamic approach.
3. Write a program to pass the data using session.
4. Write a servlet to remove spam.
5. Maintain the record of faculty member using jsp action tags and directives.
6. Design a tag to perform the necessary editing in a given report.
7. Design reusable components of the form using taglib.
8. Implement sending and receiving mail utility using Java Mail API.
9. Implement Java Message Service queue.
10. Understand working of framework – struts- case study.

Project Based Learning Assignments

Note:- *Students in a group of 3 to 4 shall complete any one project from the following list

1. Implement assignment and project submission system.
2. Implement a program to issue Leaving Certificate, Transcripts and Bonafede certificate to student
3. Implement a program to assign problem statement for practical examination in secured environment.
4. Design a template for NBA report.
5. Design an application for Feedback Management System.
6. Design application to maintain track of research paper with indexing per year.
7. Create message and mail communication of given message.
8. Design a post customized as per social media platform.
9. Design an interface to collect job opportunities and disseminate to eligible student
10. Design a project to track details of Industrial Training.