SRIT R20
COURSE STRUCTURE
&
SYLLABUS

B. Tech Regular Four Year Degree Program
(Applied for the Batches admitted from 2020-2021)
&
B. Tech (LES) for the batches admitted from 2021-2022

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

[AUTONOMOUS]
Affiliated to JNTUA & Approved by AICTE
Accredited by NAAC with 'A' Grade & Accredited by NBA (CSE, ECE & EEE)Rotarypuram Village, B K Samudram Mandal,
Ananthapuramu - 515701

COURSE STRUCTURE AND SYLLABI (Based on AICTE Model Curriculum)SRIT-R20

Bachelor of Technology In
Computer Science and Engineering

B. Tech (Regular- Full time)
(Effective for the students admitted into I Year from the Academic year **2020- 2021**)
&
B. Tech (Lateral Entry Scheme)
(Effective for the students admitted into II Year from the Academic year **2021- 2022**)

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY
(Autonomous)
Affiliated to JNTUA & Approved by AICTE Accredited by NAACwith 'A' Grade & NBA (CSE, ECE & EEE)
Rotarypuram Village, B K SamudramMandal,Ananthapuramu - 515701.

I Semester: I BTech I Semester (5 Theory + 3 Labs)

B. Tech Course Structure

Semester 0

(Common for all branches of Engineering)

S.No.	Course Name	L – T – P - C
1.	Physical Activities -- Sports, Yoga and Meditation, Plantation	0 – 0 – 6 - 0
2.	Career Counseling	2 – 0 – 2 - 0
3.	Orientation to all branches -- career options, tools, etc.	3 – 0 – 0 - 0
4.	Orientation on admitted Branch --corresponding labs, tools and platforms	2 – 0 – 3 - 0
5.	Proficiency Modules & Productivity Tools	2 – 1 – 2 - 0
6.	Assessment on basic aptitude and mathematical skills	2 – 0 – 3 - 0
7.	Remedial Training in Foundation Courses	2 – 1 – 2 - 0
8.	Human Values & Professional Ethics	3- 0 – 0 - 0
9.	Communication Skills -- focus on Listening, Speaking, Reading, Writing skills	2 – 1 – 2 - 0
10.	Concepts of Programming	2 – 0 – 2 - 0

I Semester: I BTech I Semester (5 Theory + 3 Labs)

Course Code	Course Name	Subject Area	Periods per week			Credits	Scheme of Examination Max. Marks		
			L	T	P		CIA	SEE	Total
R204GA54101	Linear Algebra and Calculus	BSC	2	1	0	3	40	60	100
R204GA56101	Applied Physics	BSC	2	1	0	3	40	60	100
R204GA52101	Communicative English-I	HSMC	3	0	0	3	40	60	100
R204GA05101	Problem Solving & Programming	ESC	3	0	0	3	40	60	100
R204GA03101	Engineering Graphics	ESC	1	0	4	3	40	60	100
R204GA52102	Communicative English-I Lab	HSMC	0	0	3	1.5	40	60	100
R204GA56102	Applied Physics Lab	BSC	0	0	3	1.5	40	60	100
R204GA05102	Problem Solving & Programming Lab	ESC	0	0	3	1.5	40	60	100
Total						19.5	320	480	800

II Semester: I B. Tech II Semester (4 Theory + 4 Labs)

Course Code	Course Name	subject Area	Periods per week			Credits	Scheme of Examination Max. Marks		
			L	T	P		CIA	SEE	Total
R204GA54201	Transforms & Partial Differential Equations	BSC	2	1	0	3	40	60	100
R204GA51102	Applied Chemistry	BSC	3	0	0	3	40	60	100
R204GA02101	Basic Electrical & Electronics Engineering	ESC	2	1	0	3	40	60	100
R204GA05201	Data Structures	ESC	3	0	0	3	40	60	100
R204GA03104	Engineering WorkshopPractice	ESC	1	0	4	3	40	60	100
R204GA02102	Basic Electrical& Electronics Engineering Lab	ESC	0	0	3	1.5	40	60	100
R204GA51104	Applied Chemistry Lab	BSC	0	0	3	1.5	40	60	100
R204GA05202	Data Structures Lab	ESC	0	0	3	1.5	40	60	100
Total						19.5	320	480	800

III Semester: II B. Tech I Semester (5 Theory + 3 Labs+1 NCMC+SOC)

Course Code	Course Name	subject Area	Periods per week			Credits	Scheme of Examination Max. Marks		
			L	T	P		CIA	SEE	Total
R204GA54302	Probability & Statistics	BSC	3	0	0	3	40	60	100
R204GA05301	Database Management Systems	PCC	3	0	0	3	40	60	100
R204GA52301	English Language & Employment Skills for Engineers	HSS	3	0	0	3	40	60	100
R204GA05302	Object Oriented Programming	PCC	3	0	0	3	40	60	100
R204GA05303	Software Engineering	PCC	3	0	0	3	40	60	100
R204GA05304	Data Base Management Systems Lab	PCC	0	0	3	1.5	40	60	100
R204GA05305	Object Oriented Programming Lab	PCC	0	0	3	1.5	40	60	100
R204GA05306	Software Engineering Lab	PCC	0	0	3	1.5	40	60	100
R204GA5MC01	Environmental Science	NCMC	2	0	0	0	40	-	40
R204GA05307	Skill Oriented Course-I	SOC	1	0	2	2	100	-	100
Total						21.5	460	480	940

IV Semester: II B. Tech II Semester (5 Theory + 3 Labs+1 NCMC+SOC)

Course Code	Course Name	subject Area	Periods per week			Credits	Scheme of Examination Max. Marks		
			L	T	P		CIA	SEE	Total
R204GA05401	Discrete Mathematics	BSC	3	0	0	3	40	60	100
R204GA05402	Formal Languages and Automata Theory	PCC	3	0	0	3	40	60	100
R204GA05403	Python Programming	PCC	3	0	0	3	40	60	100
R204GA05404	Design and Analysis of Algorithms	PCC	3	0	0	3	40	60	100
R204GA04407	Digital Logic Design	ESC	3	0	0	3	40	60	100
R204GA05405	Python Programming Lab	PCC	0	0	3	1.5	40	60	100
R204GA05406	Design and Analysis of Algorithms Lab	PCC	0	0	3	1.5	40	60	100
R204GA05407	Linux Programming Lab	PCC	0	0	3	1.5	40	60	100
R204GA5MC02	Indian Constitution	NCMC	2	0	0	0	40	-	40
R204GA05408	Skill Oriented Course-II	SOC	1	0	2	2	100	-	100
Total						21.5	460	480	940
Internship2 Months (Mandatory) during summer vacation									
Honors/Minor courses (The hours distribution can be 3-0-2 or 3-1-0 also)						4	0	0	4

V Semester: III B. Tech I Semester (5 Theory + 2 Labs+1NCMC+SAC+Internship)

Course Code	Course Name	subject Area	Periods per week			Credits	Scheme of Examination Max. Marks		
			L	T	P		CIA	SEE	Total
R204GA05501	Web Development Technologies	PCC	3	0	0	3	40	60	100
R204GA05502	Computer Networks	PCC	3	0	0	3	40	60	100
R204GA05503	Operating Systems	PCC	3	0	0	3	40	60	100
Professional Elective courses – I									
R204GA05504	1.Data Warehousing and Data Mining	PEC	3	0	0	3	40	60	100
R204GA05505	2. Wireless Sensor Networks								
R204GA05506	3. Software Testing Methodologies								
Open Elective-I		OEC	2	0	2	3	40	60	100
R204GA05509	Web Development Technologies Lab	PCC	0	0	3	1.5	40	60	100
R204GA05510	Computer Networks and Operating Systems Lab	PCC	0	0	3	1.5	40	60	100
R204GA05511	Skill Oriented Course-III	SOC	1	0	2	2	100	-	100
R204GA5MC03	Essence of Indian Traditional Knowledge	NCMC	2	0	0	0	40	-	40
R204GA05512	Summer Internship-I	SI	0	0	0	1.5	100	-	100
Total						21.5	520	420	940
Honors/Minor courses (The hours distribution can be 3-0-2 or 3-1-0 also)						4	0	0	4

VI Semester: III B. Tech II Semester (5 Theory + 3 Labs+1 NCMC+SAC)

Course Code	Course Name	subject Area	Periods per week			Credits	Scheme of Examination Max. Marks		
			L	T	P		CIA	SEE	Total
R204GA05513	Compiler Design	PCC	3	1	0	3	40	60	100
R204GA05601	Artificial Intelligence & Machine Learning	PCC	3	0	0	3	40	60	100
R204GA05602	Android Application Development	PCC	3	0	0	3	40	60	100
Professional Elective Courses –II									
R204GA05603	1.Introduction to Big data	PEC	3	0	0	3	40	60	100
R204GA05604	2. Cyber Security								
R204GA05605	3. Software Project Management								
Open Elective – II		OEC	2	0	2	3	40	60	100
R204GA05608	Compiler Design Lab	PCC	0	0	3	1.5	40	60	100
R204GA05609	Artificial Intelligence & Machine Learning Lab	PCC	0	0	3	1.5	40	60	100
R204GA05610	Android Application Development Lab	PCC	0	0	3	1.5	40	60	100
R204GA05611	Skill Oriented Course-IV	SOC	1	0	2	2	100	-	100
R204GA5MC04	Life Sciences for Engineers	NCMC	2	0	0	0	40	-	40
Total						21.5	460	480	940
Honors/Minor courses (The hours distribution can be3-0-2 or 3-1-0 also)						4	0	0	4
Industrial/Research Internship (Mandatory) 2 Months during summer vacation									

VII Semester: IV B. Tech I Semester (5 Theory + SAC+ HSS+Internship)

Course Code	Course Name	subject Area	Periods per week			Credits	Scheme of Examination Max. Marks		
			L	T	P		CIA	SEE	Total
Professional Elective Courses – III									
R204GA05701	1.Data Analytics	PEC	3	0	0	3	40	60	100
R204GA05702	2. Mobile Computing								
R204GA05703	3. Software Requirements & Estimation								
Professional Elective Courses – IV									
R204GA05704	1. Artificial Neural Networks	PEC	3	0	0	3	40	60	100
R204GA05705	2. Internet of Things and Its Applications								
R204GA05706	3. Software Quality Assurance								
Professional Elective Courses – V									
R204GA32702	1. Deep Learning	PEC	3	0	0	3	40	60	100
R204GA05707	2. Block Chain Fundamentals								
R204GA05708	3. Agile Methodologies								
Open Elective – III		OEC	2	0	2	3	40	60	100
Open Elective – IV		OEC	2	0	2	3	40	60	100
	*Humanities and Social Science Elective	HSS	3	0	0	3	40	60	100
R204GA05713	Skill Oriented Course-V	SOC	1	0	2	2	100	-	100
R204GA05714	Summer Internship-II	SI	0	0	0	3	100	-	100
Total						23	440	360	800

VIII Semester: IV B. Tech II Semester (Project work, seminar and internship in industry)

Course Code	Course Name	subject Area	Periods per week			Credits	Scheme of Examination Max. Marks		
			L	T	P		CIA	SEE	Total
R204GA05801	Project work, seminar and internship in industry	Major Project	0	0	0	12	80	120	200
Total						12	80	120	200

Open Elective-I (V Semester, III B. Tech, I-Semester)

Course Code	Course Name	Subject Area	Periods per week			Credits	Scheme of Examination Max. Marks		
			L	T	P		CIA	SEE	Total
R204GA01504	Air Pollution and Control	OEC	3	0	0	3	40	60	100
R204GA01505	Construction Technology	OEC	3	0	0	3	40	60	100
R204GA02504	System Reliability Concepts	OEC	3	0	0	3	40	60	100
R204GA02505	Design of PV Systems	OEC	3	0	0	3	40	60	100
R204GA03508	Entrepreneurship	OEC	3	0	0	3	40	60	100
R204GA03509	Additive Manufacturing	OEC	3	0	0	3	40	60	100
R204GA04507	Digital Electronics	OEC	2	1	0	3	40	60	100
R204GA04508	Principles of Communication	OEC	2	1	0	3	40	60	100
R204GA05507	Essentials of Python Programming	OEC	3	0	0	3	40	60	100
R204GA05508	Computer Organization & Operating System	OEC	3	0	0	3	40	60	100
R204GA52501	Business Environment & Policies	OEC	3	0	0	3	40	60	100
R204GA52502	Managerial Economics and	OEC	3	0	0	3	40	60	100

Open Elective-II (VI Semester, III B. Tech, II-Semester)

Course Code	Course Name	Subject Area	Periods per week			Credits	Scheme of Examination Max. Marks		
			L	T	P		CIA	SEE	Total
R204GA01608	Architecture and Town Planning Measurements	OEC	3	0	0	3	40	60	100
R204GA01609	Sustainable Development And Environment Management	OEC	3	0	0	3	40	60	100
R204GA02606	Energy Storage Systems	OEC	3	0	0	3	40	60	100
R204GA02607	Electrical Safety Measures	OEC	3	0	0	3	40	60	100
R204GA03608	Non Destructive Testing And Evaluation	OEC	3	0	0	3	40	60	100
R204GA03609	Total Quality Management	OEC	3	0	0	3	40	60	100
R204GA04607	Basics of VLSI	OEC	2	1	0	3	40	60	100
R204GA04608	Principles of Digital Signal Processing	OEC	2	1	0	3	40	60	100
R204GA05606	Mean Stack Technology	OEC	3	0	0	3	40	60	100
R204GA05607	Introduction to Artificial Intelligence	OEC	3	0	0	3	40	60	100
R204GA56601	Optical Physics and Its Applications	OEC	3	0	0	3	40	60	100
R204GA52503	Management Science	OEC	3	0	0	3	40	60	100

Open Elective-III (VII Semester, IV B. Tech, I-Semester)

Course Code	Course Name	Subject Area	Periods per week			Credits	Scheme of ExaminationMax. Marks		
			L	T	P		CIA	SEE	Total
R204GA01713	Disaster Managementand Mitigation	OEC	3	0	0	3	40	60	100
R204GA01714	Sustainable Energy Efficient Building Materials & Technologies	OEC	3	0	0	3	40	60	100
R204GA02709	Electrical Engineering Materials	OEC	3	0	0	3	40	60	100
R204GA02710	Solar Energy Conversion Systems	OEC	3	0	0	3	40	60	100
R204GA03713	Basics of Electric Vehicles	OEC	3	0	0	3	40	60	100
R204GA03714	Supply Chain Management	OEC	3	0	0	3	40	60	100
R204GA04710	Principles of Microcontrollers &	OEC	2	1	0	3	40	60	100
R204GA04711	Basics of Image Processing	OEC	2	1	0	3	40	60	100
R204GA05709	Data Science	OEC	3	0	0	3	40	60	100
R204GA05710	Fundamentals of Security in Computing	OEC	3	0	0	3	40	60	100
R204GA54701	Mathematical Modelling	OEC	3	0	0	3	40	60	100
R204GA56701	Thin Film Technologyand Its Applications	OEC	3	0	0	3	40	60	100

Open Elective-IV (VII Semester, IV B. Tech, I-Semester)

Course Code	Course Name	Subject Area	Periods per week			Credits	Scheme of ExaminationMax. Marks		
			L	T	P		CIA	SEE	Total
R204GA01715	Low Cost Housing Techniques	OEC	3	0	0	3	40	60	100
R204GA01716	Green Buildings	OEC	3	0	0	3	40	60	100
R204GA02711	Wind Energy Conversion Systems	OEC	3	0	0	3	40	60	100
R204GA02712	Soft Computing Techniques	OEC	3	0	0	3	40	60	100
R204GA03715	Industrial Automation and Robotics	OEC	3	0	0	3	40	60	100
R204GA03716	Alternative Sources of Energy	OEC	3	0	0	3	40	60	100
R204GA04712	Principles of Embedded Systems	OEC	2	1	0	3	40	60	100
R204GA04713	Design Thinking	OEC	2	1	0	3	40	60	100
R204GA05711	Virtualization and Cloud Computing	OEC	3	0	0	3	40	60	100
R204GA05712	Blockchain Technologyand Applications	OEC	3	0	0	3	40	60	100
R204GA54702	OptimizationTechniques	OEC	3	0	0	3	40	60	100
R204GA51701	Global Warming andClimate Changes	OEC	3	0	0	3	40	60	100
R204GA05715	Programming In Java	OEC	3	0	0	3	40	60	100

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Linear Algebra & Calculus

(Common to all Branches)

I B.Tech - I Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
R204GA54101	**BSC**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		2	1	0	3	40	60	100

Objectives

- This course will illustrate the students in the concepts of Linear Algebra and calculus.
- To equip the students with standard concepts like differentiation, integration at an intermediate to advanced level of mathematics to develop the confidence and ability among the students to handle various applications in their respective core subjects.

Unit I - Linear Algebra (Matrices and Solution of equations)

Basic definitions of matrices, Rank of a matrix by echelon form, Normal form; Normal form of PAQ, solving system of homogeneous and non-homogeneous linear equations, Eigen values and Eigen vectors and their properties.

Learning Outcomes:

At the end of this unit, the students will be able to

1. Determine the rank of a matrix.
2. Solving system of non-homogeneous linear equations.
3. Solving system of homogeneous linear equations.
4. Determine Eigen values and Eigen vectors.

Unit II - Linear Algebra (Inverse of Matrix and reduction into various forms)

Cayley-Hamilton theorem (without proof), finding inverse and power of a matrix by Cayley-Hamilton theorem, Diagonalization of a matrix, quadratic forms and nature of the quadratic forms, reduction of quadratic form to canonical forms by orthogonal transformation.

Learning Outcomes:

At the end of this unit, the students will be able to

1. Diagonal form and different factorizations of a matrix.
2. Identify special properties of a matrix, such as positive definite…etc.
3. Use this information to facilitate the calculation of matrix characteristics.
4. Understand quadratic forms and their conversions by orthogonal transformation.

Unit III - Differential Equations (First order and First degree)

Formation of differential equations, Differential equations of 1^{st} order and 1^{st} degree-variable separable, Homogeneous equations and equations reducible to homogeneous, Exact, Non-exact, Linear and Bernoulli's equations.

Applications of Differential equations of 1^{st} order and 1^{st} degree- Newton's law of cooling, Law of Natural growth and decay, Orthogonal trajectories.

Learning Outcomes:

At the end of this unit, the students will be able to

1. Form the differential equations from the given situations.
2. Solve the differential equations of first order and first degree.
3. Understand the applications of differential equations.

Unit IV - Differential Equations (Second and higher order)

Non-Homogeneous linear differential equations of second and higher order with constant co-efficient with RHS term of the type e^{ax}, sin ax, cos ax, polynomials in x, $e^{ax}V(x)$ and$xV(x)$. Method of variation of parameters, Cauchy's and Legendre's linear equations, Applications to L- C-R Circuit problems and Deflection of Beams.

Learning Outcomes:

At the end of this unit, the students will be able to

1. Identify the essential characteristics of linear differential equations with constant coefficients.
2. Solve the linear differential equations with constant coefficients by appropriate method
3. Classify and interpret the solutions of linear differential equations.
4. Formulate and solve the higher order differential equation by analyzing physical situations.

Unit V - Multiple Integrals

Double integrals, change of order of integration, double integration in polar coordinates, areas enclosed by plane curves.

Evaluation of triple integrals, change of variables between Cartesian, cylindrical and spherical polar co-ordinates.

Learning Outcomes:

At the end of this unit, the students will be able to

1. Evaluate double integrals of functions of several variables in two dimensions using Cartesian and polar co-ordinates.
2. Apply double integration techniques in evaluating areas bounded by region.
3. Evaluate multiple integrals in Cartesian, cylindrical and spherical geometries.

Text Books:

1. Higher Engineering Mathematics, B.S.Grewal, 44th edition, 1965, Khanna publishers.
2. Engineering Mathematics-I & Mathematical methods, E. Rukmangadachari & E. Keshava Reddy, 1st January 2013 and April 4th 2009, Pearson publishers.
3. Engineering Mathematics Volume-I & Mathematical methods, T. K.V. Iyengar, B. Krishna Gandhi, S. Ranganatham and M.V.S.S.N. Prasad, 16th edition, 2014 and 2006, S. Chand publications.

Reference Books:

1. Advanced Engineering Mathematics, Erwin Kreyszig, 10th edition reprint 2016, Wiley India.
2. Higher Engineering Mathematics, B.V. Ramana, 32nd reprint 2018, Mc Graw Hill publishers.
3. Advanced Engineering Mathematics, Alan Jeffrey, 1st edition, 19th June 2001, Elsevier.

Course Outcomes:

At the end of the course, students will be able to

1. Solve the system of linear and differential equations and evaluate multiple integrals.
2. Understand various operations on matrices, Eigen values and Eigen vectors.
3. Determine the diagonalization of a matrix and quadratic forms.

4. Solve the Differential equations of first order Exact, Linear, Bernoulli.
5. Solve the second and higher order differential equations by finding particular integral methods such as e^{ax}, sin ax, polynomials in x, e^{ax} V(x) and xV(x).
6. Determine the surface and volume integrals of cardioids, lemniscates by using Double and Triple integrals.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Applied Physics

(Common to ECE, EEE, CSE, CSD & CSM)

I B.Tech - I Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
R204GA56101	**BSC**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		2	1	0	3	40	60	100

Objectives

- To identify the importance of the optical phenomenon i.e., interference, diffraction and polarization related to its engineering applications.
- To impart knowledge in basic concepts of LASERs and optical fibres along with its engineering applications.
- To explain the significant concepts of magnetic and dielectric materials with their potential applications in the emerging micro-devices.
- To understand the basic knowledge of band theory of solids and identify the importance of semiconductors in the functioning of electronic devices.
- To familiarize the concepts related to super conductors and nano materials with their fascinating applications.

Unit I - Physical Optics

Principle of Superposition–Interference of light–Interference in thin films by reflection, Newton's Rings–determination of the wavelength of a given monochromatic light and refractive index. Applications of interference.

Diffraction–Fresnel and Fraunhofer Diffraction, Fraunhofer Diffraction due to Single-slit and Double-slit, Diffraction Grating, Applications of diffraction.

Polarization–Polarization by double refraction, Nicol's Prism, Half wave and Quarter wave plates, Applications of Polarization.

Learning Outcomes:

At the end of unit**,** students will be able to

1. Explain about coherent sources and the conditions for sustained interference.
2. Identify engineering applications of interference and diffraction.
3. Analyze the differences between interference and diffraction with applications.
4. Illustrate the concept of polarization of light and its applications.
5. Classify ordinary polarized light and extraordinary polarized light.

Unit II - Lasers and Fibre Optics

Introduction–Characteristics of LASER, Spontaneous and Stimulated emission of radiation, Einstein's coefficients, Population Inversion, Pumping Mechanisms, He-NeLASER, Semiconductor LASER, Applications of the LASER.

Introduction to Optical Fibres–Total Internal Reflection, Construction of optical fibres, Critical angle of propagation, Acceptance angle, Numerical Aperture, Classification of fibres based on refractive index profile & modes, Propagation of electromagnetic wave through optical fibre importance of V number, Block Diagram of Fibre, Optic Communication system, Fibre Optics sensors, Medical Applications.

Learning Outcomes:

At the end of this Unit, the students will be

1. Understand the basic concepts of LASER light.
2. Apply the concepts to learn the types of LASERs.
3. Identify the Engineering applications of LASERs.
4. Explain the working principle of optical fibres.
5. Classify optical fibres based on refractive index profile and mode of propagation.
6. Identify the applications of optical fibres in medical, communication and other fields.

Unit III - Band Theory of Solids and Semiconductors

De-Broglie Hypothesis, Schrodinger's Time Independent equation, Particle in one-dimensional potential well, Classical Free Electron Theory, Quantum Free Electron Theory, Fermi-Dirac Distribution, Sources of Electrical Resistance, Band Theory of Solids, Classification of Solids into Conductors, Semiconductors and Insulators.

Semiconductors: Intrinsic semiconductors, Extrinsic semiconductors: P-type & N-type, Drift and Diffusion currents, Einstein Relation, Hall effect, Applications of Hall effect, Direct and Indirect band gap semiconductors, Working principle of P-N Junction Diode, Applications of Semiconductors.

Learning Outcomes:

At the end of Unit, the students will be able to

1. Describe the Schrodinger's wave equations of a particle.
2. Explain classical and quantum free electron theory.
3. Classify the solids into conductors, semiconductors and insulators.
4. Explain the direct and indirect band gap semiconductors.
5. Identify the type of semiconductor using Hall effect.
6. Describe the working principle of semiconductor diode.

Unit IV - Magnetic Materials and Dielectric Materials

Introduction–Magnetic dipole moment, Magnetization, Magnetic susceptibility and permeability, Origin of magnetic moments, Classification of Magnetic materials, Weiss theory of ferromagnetism(qualitative), Hysteresis–soft and hard magnetic materials, Magnetic Device Applications (Magnetic bubble memory).

Introduction–Dielectric polarization, Dielectric polarizability, Susceptibility and Dielectric constant, Types of polarizations: Electronic, Ionic (Quantitative) and Orientation (Qualitative) Polarizations, Lorentz field, Clausius-Mossotti equation, Applications of Dielectrics: Ferro Electricity.

Learning Outcomes:

At the end of Unit, the students will be able to

1. Classify the magnetic materials based on susceptibility and their temperature dependence.
2. Apply the concept of magnetism to magnetic devices.
3. Explain the concepts of dielectric constant and polarization in dielectric materials.
4. Describe various types of polarizabilities of dielectric materials.
5. Discuss Lorentz field and Claussius-Mossotti relation in dielectrics.
6. Explain the applications of dielectric and magnetic materials.

Unit V - Superconductors and Nano materials

Superconductors - Properties, Meissner's effect, Penetration depth, Type-I and Type-II Superconductors, DC and AC Josephson Effects, BCS Theory, High TC superconductors, Applications of superconductors.

Nano materials–surface area to volume ratio and quantum confinement, Properties of nano materials: Optical, Electrical, Magnetic and Mechanical properties, Synthesis of nano materials: Top-down–Ball Milling, Bottom-up–Chemical Vapour Deposition, Applications of Nano materials.

Learning Outcomes:

At the end of Unit, the students will be able to

1. Explain how electrical resistivity of solids changes with temperature.
2. Classify superconductors based on Meissner's effect.
3. Explain BCS theory & Josephson effect in superconductors.
4. Identify the nano size dependent properties of nano materials.
5. Illustrate the methods for the synthesis and characterization of nano materials.
6. Apply the basic properties of nano materials in various Engineering branches.

Text Books:

1. A Textbook of Engineering Physics, M. N. Avadhanulu, P.G. Kshirsagar & TVS Arun Murthy, 11th Edition, 2019, S. Chand Publications.
2. Engineering Physics, Shatendra Sharma and Jyotsna Sharma, 1st Edition, 2018, Pearson Education.

Reference Books:

1. Engineering Physics, B. K. Pandey and S. Chaturvedi, 1st Edition, 2012,Cengage Learning.
2. A Textbook of Nano Science and Nano Technology, T Pradeep, 1st Edition, 2013, Tata McGraw Hill.
3. Optical Fiber Communications, Gerd Keiser, 4thEdition, 2008, Tata McGraw Hill.
4. Fundamentals of Physics, Halliday, Resnick and Walker, 4thEdition, 2013, Wiley India Pvt. Ltd.

Course Outcomes:

At the end of the course, students will be able to

1. Calculate various physical parameters related to applied physics.
2. Explain the phenomenon of interference, diffraction and polarization and its applications.
3. Describe the construction and working of LASERs and Optical fibre and its applications in different fields.
4. Explain the classification of solids and conductivity in semiconductors.

5. Describe the classification of magnetic materials and types of polarization in dielectric materials.
6. Explain the theory of superconductivity and synthesis of nano materials.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Communicative English-I

(Common to all Branches)

I B.Tech - I Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
R204GA52101	**HSMC**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		3	0	0	3	40	60	100

Objectives

- To enable students, improve their lexical, grammatical and communicative competence.
- To comprehend the given texts and respond appropriately.
- Focus on appropriate reading strategies and reinforce language skills for comprehension of various academic texts and authentic material.
- To provide them an intensive training in writing skills.

Unit - I

Lesson: The Death Trap: *Saki*

Grammar: Noun-Pronoun Agreement, Subject-Verb Agreement

Vocabulary Building: Word Formation III–Prefixes and Suffixes from Other Languages Writing: Principles of good writing, paragraph writing

Speaking: Making Request and Responding to Them Reading:

Time Management: On Saving Time by *Seneca*

Learning Outcomes:

At the end of Unit, students will be able to:

1. Recognize the relation between noun and pronoun and construct meaningful sentences using subject-verb agreement rules to reduce errors in writing.
2. Construct new words using prefixes and suffixes to enlarge the lexical knowledge and connect ideas using the principles, techniques of paragraph writing and paraphrasing.
3. Apply the useful phrases for making requests and responding to them in day-to-day conversations.

Unit - II

Lesson: Self-Improvement: How I Became a Public Speaker by *George Bernard Shaw*

Grammar: Articles, Tenses

Vocabulary Building: Word Formation II–Root words from Other Languages Writing: Punctuation and Spelling

Speaking: Getting Someone's Attention and Interrupting

Reading: The Brook: *Alfred Tennyson*

Learning Outcomes:

At the end of Unit, the students will be able to

1. Identify the use of articles and tenses in a context and recognize the purposes of punctuation and correct spelling for better writing skills.
2. Memorize the root words and other word formation strategies to enrich the vocabulary.
3. Express to others using appropriate phrases for improving speaking skills.

Unit - III

Lesson: Values and Ethics: If (poem) by *Rudyard Kipling*

Grammar: Prepositions and Question tags

Vocabulary Building: Word Formation–Introduction to Word formation, Homophones, homographs and homonyms

Writing: If-Clauses and Sentences, Letter writing

Speaking: Introducing Oneself and others

Reading: On the Conduct of Life: *William Hazlitt*

Learning outcomes:

At the end of Unit, the students will be able to

1. Understand the text to make effective comprehension and recall the use of prepositions and question tags.
2. Define the uses of affixes and other word formation strategies to enhance vocabulary.
3. Recognize the differences between clauses and sentences to excel in writing skills and practice the techniques of introducing to improve better speaking skills.

Unit - IV

Lesson: Innovation: Muhammad Yunus

Grammar: Misplaced Modifiers, Modals Vocabulary Building: Synonyms and antonyms

Writing: Essay writing

Speaking: Role Plays for practice of Conversations

Reading: ChinduYellama

Learning Outcomes:

At the end of Unit, the students will be able to

1. Identify the use of modifiers for producing meaningful and grammatically correct sentences.
2. Reproduce and rewrite the words in sentences by choosing similar and opposite words.
3. Apply the methods and techniques to write coherent and meaningful essays.
4. Practice the conversations using situations and apply the knowledge in enhancing their conversational skills.

Unit - V

Lesson: Politics and the English Language: *George Orwell*

Grammar: Redundancies and Clichés and conjunctions

Vocabulary Building: Common Abbreviations

Writing: Writing a Summary Speaking:

Formal Oral Presentations

Reading: Motivation: The Dancer with a White Parasol by *Ranjana Dave*

Learning outcomes:

At the end of Unit, the students will be able to

1. Analyze the roles of redundancies and clichés in producing meaningful and grammatically correct sentences.
2. Discuss various commonly used abbreviations and explain their full forms for better understanding.
3. Develop a paragraph by applying the strategies of writing summaries to improve writing skills.
4. Design and develop oral presentations to perform in front of the audience.

Text Books:

1. Language and Life: A Skills Approach, Vol. I,First Edition, 2018, Orient Blackswan Private Limited.

Reference Books:

1. Practical English Usage, Michael Swan, 4th Edition, 2017, Oxford University Press.
2. Remedial English Grammar, F.T. Wood, 2007, Macmillan.
3. On Writing Well, William Zinsser, 2012, Harper Resource Book.
4. Study Writing, Liz Hamp-Lyons and Ben Heasly, 2006, Cambridge University Press.
5. Communication Skills, Sanjay Kumar and PushpLata, Second Edition, 2015, Oxford University Press.
6. Exercises in Spoken English, Parts, I-III. CIEFL, Hyderabad. Oxford University Press.

Course Outcomes:

At the end of the course, the students will be able to

1. Remember explicit and implicit meaning of a text.
2. Understand and produce academic vocabulary appropriately orally and in writing.
3. Apply relevant formats of letter writing, essay writing and writing summary.
4. Analyze the use of grammar to construct meaningful sentences with suitable word choices.
5. Evaluate comprehensive skills through listening and reading texts on Life Skills.
6. Create situations in order to enhance speaking skills in introducing, requesting and responding.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Problem Solving & Programming

(Common to all Branches)

I B.Tech - I Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA05101	**ESC**	3	0	0	3	40	60	100

Objectives

- Introduce the concept of Algorithm and use it to solve computational problems.
- Teach the syntax and semantics of a C Programming language.
- Demonstrate the use of Control structures of C Programming language.
- Illustrate the methodology for solving computational problems.

Unit I - Introduction to Programming, Algorithms, Flowcharts and C Programming

Programs and Programming, Programming languages, Compiler, Interpreter, Loader, Linker, Program execution, Fourth generation languages. Fifth generation languages, Classification of Programming languages, Structured Programming concept, Algorithms, Pseudo-code, Flowcharts, Strategy for designing algorithms, tracing an algorithm to depict logic and specification for converting algorithms into programs.

Introduction to programming languages-standardization of C Language, developing programs in C, a simple C program and parts of C Program revisited structure of a C Program.

Learning Outcomes:

At the end of unit, students will be able to

1. Understand the generations and classification of programming languages.
2. Solve complex problems using language independent notations.
3. Illustrate the steps in compilation of a C Program.

Unit II - Basics of C Programming, Input and Output and Control Flow

Variable names, data types and sizes, constants, declarations, arithmetic operators, relational and logical operators, type conversions, increment and decrement operators, bitwise operators, assignment operators and expressions, conditional expressions, precedence and order of evaluation.

Standard input and output, formatted output-printf, formatted input-scanf.

Statements and blocks, if else, else-if, switch, Loops – while, for, do-while, break and continue, Goto and Labels.

Learning Outcomes:

At the end of unit**,** students will be able to

1. Recognize the elements of C Programming language.
2. Recognize the formatted input/output functions for I/O operations.
3. Select the control flow structure for solving the problem.

Unit III - Functions and Program Structure

Basics of functions, functions returning non-integers, external variables, scope variables, header variables, register variables, block structure, initialization, recursion, the C processor.

Finding the square root of a number, the smallest divisor of a number, the greatest common divisor of two integers, generating prime numbers.

Learning Outcomes:

At the end of unit, students will be able to

1. Apply modular approach for solving the problem.
2. Recognize the C pre-processor functions for pre-processing.
3. Solve mathematical problems using C Programming language.

Unit IV - Arrays, Strings & Pointers

Introduction to Arrays- One-dimensional arrays, strings: one-dimensional character arrays, multi-dimensional arrays, array of strings.

Introduction to pointers: understanding memory addresses, address operator, pointer, void pointer, null pointer, use of pointers, arrays and pointers, pointers and strings, pointers to pointers, pointers to functions, dynamic memory allocation.

Learning Outcomes:

At the end of unit, students will be able to

1. Apply String manipulation functions.
2. Structure the individual date elements to simplify the solutions.
3. Organize homogenous data.
4. Facilitate efficient memory utilization.

Unit V - User defined data types, Variables, Sorting, Searching and Some other Features

Structures, unions, enumeration types, bit- fields.

Sorting by selection, sorting by exchange, sorting by insertion, sorting by partitioning, linear search, binary search.

Command line arguments, Variable-length argument list, formatted input- scanf, file access, Error handling- stderr and exit, Line Input and Output, Miscellaneous Functions.

Learning Outcomes:

At the end of unit**,** students will be able to

1. Organize heterogeneous data.
2. Illustrate searching and sorting algorithms based on the type of the data.
3. Understand standard Input/output and Error operations.

Text Books:

1. PradipDey, and Manas Ghosh, "Programming in C", 2018, Oxford University Press.
2. R.G. Dromey, "How to Solve it by Computer", 2014, Pearson.
3. Brian W. Kernighan, Dennis M, Ritchie, "The C Programming Language", 2nd Edition,Pearson.

Web Resources:

https://www.learn-c.org/

https://www.cprogramming.com/tutorial/

Reference Books:

1. RS Bichkar "Programming with C", 2012, Universities Press.
2. PelinAksoy, and Laura Denardis, "Information Technology in Theory", 2017, C engage Learning.
3. Byron Gottfried and Jitender Kumar Chhabra, "Programming with C", 4th Edition, 2019, McGraw Hill Education.

Course Outcomes:

At the end of the course, students will be able to

1. Recognize the importance of programming language independent constructs.
2. Know the basic understanding of computers, language translators and algorithmic thinking.
3. Demonstrate the identifiers, operators, I/O functions and control statements for branching and looping.
4. Illustrate the modular programming using functions for code reuse.
5. Organize the data using arrays and memory management using pointers.
6. Learn user defined data types, files, searching and sorting.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Engineering Graphics

(Common to ECE, CSE, CSD & CSM)

I B.Tech - I Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
R204GA03101	**ESC**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		1	0	4	3	40	60	100

Objectives

- Bring awareness that Engineering Graphics is the Language of Engineers.
- Familiarize how industry communicates technical information.
- Teach the practices for accuracy and clarity in presenting the technical information.
- Develop the engineering imagination essential for successful design.
- Instruct the utility of drafting &modelling by using AutoCAD.

Unit I - Introduction to drawing and Projections of points and Straight lines

Introduction to drawing- Principles of orthographic projection – Convention – First angle projections, projections of points, Projection of Straight lines in simple position, inclined to one plane and inclined to both the planes.

Learning Outcomes:

At the end of this unit, the students will be able to

1. Explain the significance of engineering graphics.
2. Sketch the projections of points with respect to different quadrants.
3. Sketch the projections of lines inclined to one or both the planes.

Unit II - Projection of planes

Projection of planes in simple position, inclined to one plane and inclined to both the planes.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Sketch the projections of planes in simple position.
2. Sketch the projections of planes inclined to one plane.
3. Sketch the projections of planes inclined to both the planes.

Unit III - Projection of solids

Projection of solids, Axis perpendicular to HP, VP, axis inclined to HP, VP and axis inclined to both HP & VP.

Learning Outcomes:

At the end of this unit, the students will be able to

1. Sketch the projections of solids in simple position.
2. Sketch the projections of solids with axis inclined to one plane.
3. Sketch the projections of solids with axis inclined to both the planes.

Unit IV - Isometric projection and Ortho-ISO conversion

Principles of Isometric projection-Isometric Scale-Conventions, Isometric Views of lines, Planes, Simple solids (cube, cylinder and cone), Conversion of Ortho graphic views to Isometric views.

Learning Outcomes:

At the end of this unit, the students will be able to

1. Explain about isometric projections, isometric scale and conventions used.
2. Sketch the isometric views of lines, planes, simple solids.
3. Sketch Conversion of Ortho graphic views to isometric views.

Unit V - ISO-Ortho conversion and Introduction to AutoCAD

Conversion of isometric Views to Orthographic Views and Introduction to AutoCAD.

Learning Outcomes:

At the end of this unit, the students will be able to

1. Sketch conversion of isometric Views to orthographic Views.
2. Use AutoCAD tools in developing 2D and 3Dmodels.

Text Books:

1. Engineering Drawing, N.D. Bhatt, Fifty Third Edition, 2014, Charotar Publishing House Pvt. Ltd.
2. Engineering Drawing, K.L. Narayana& P. Kannaih, Third Edition, 2011, Scitech Publishers, Chennai.

Reference Books:

1. Engineering Drawing,DhananjayJohle, First Edition, 2017, McGrawHill education.
2. A textbook of Engineering Drawing, Prof.P. J. Shah, 2013, S. Chand Publishers.
3. Engineering Drawing + AutoCAD, K. Venugopal, Fifth Edition, 2011, New age Publishers.
4. Engineering Graphics for Degree, K.C. John, 2014, Prentice Hall of India.
5. Engineering Drawing, with AutoCAD, B.V.R.Guptha& M. Raja Roy, 2016,I.K. International Publishing House Pvt. Ltd.
6. Engineering Drawing and Graphics using AutoCAD, T. Jeyapoovan, Third Edition, 2010, Vikas publishing house.

Course Outcomes:

At the end of the course, the students will be able to

1. Construct orthographic views of points and lines with reference to inclined to one plane and inclined to both the planes.
2. Construct orthographic views of planes with reference to inclined to one plane and inclined to both the planes.
3. Construct orthographic views of solids with reference to axis inclined to HP and VP.
4. Construct orthographic views of solids with reference to axis inclined to both HP and VP.
5. Develop 3D Isometric View in relation with 2D orthographic view and sketch the conversion of orthographic views to isometric views.
6. Sketch the ISO-ortho conversion and Construct 2D (orthographic) and 3D (isometric) views in CAD environment.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Communicative English-I Lab

(Common to all Branches)

I B.Tech - I Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
R204GA52102	**HSMC**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		0	0	3	1.5	40	60	100

Objectives

- Students will be exposed to a variety of self-instructional, learner friendly modes of language learning.
- Students will cultivate the habit of reading passages from the computer monitor. Thus, providing them with the required facility to face computer based competitive exams like GRE, TOEFL, and GMAT etc.
- Students will learn better pronunciation through stress, intonation and rhythm.
- Students will be trained to use language effectively to face interviews, group discussions, public speaking.

List of Sessions

Session-I

1. Introduction to Phonetics
2. Vowels and Consonants sounds

Session-II

1. Syllables
2. Syllabification

Session-III

1. Stress: Word & Sentence stress
2. Rules for stress

Session-IV

1. Intonation
2. Falling and Rising tone

Session-V

1. Listening skills: Types of listening
   - Listening to small talks
   - Listening to vocabulary

Session-VI

1. Impromptu speech
2. Describing person/objects/situations

Session-VII

1. Giving directions
2. Telephonic skills

Session-VIII

1. Oral Presentations
2. Situational dialogues/Role plays

Session-IX

1. Debates
2. Group Discussions

Session-X

1. Reading Skills- Reading comprehension

Reference Books:

1. A Textbook of English Phonetics for Indian Students by T. Balasubramanian, 2012, Macmillan Publishers.
2. Exercises in Spoken English, Part 1,2& 3, 1997, CIEFL Hyderabad, Oxford, India.
3. Communication Skills for Engineers by Sunita Mishra & C. Muralikrishna, 2011, Pearson Education India.

Course Outcomes:

At the end of the course, the students will be able to

1. Phonemic sounds and reading comprehension skills (LR).
2. Understand write-up skills through describing objects/places/persons(W).
3. Apply an enrich language to enhance for a better speaking skill through JAM, Small talks and Debates.
4. Analyze different ways of greetings in introducing oneself/others, to a well nit summarizing while listening lectures, using enrich vocabulary to reduce errors in speech and writing.
5. Evaluate various ways of communication- verbal and non-verbal cues.

Create awareness on mother tongue influence and neutralize it in order to improve fluency in spoken English.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Applied Physics Lab

(Common to ECE, EEE, CSE, CSD & CSM)

I B.Tech - I Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
R204GA56102	**BSC**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		0	0	3	1.5	40	60	100

Objectives

- Understand the concepts of interference and diffraction and their applications.
- Understand the role of optical fibre parameters in communication.
- Recognize the importance of the energy gap in the study of conductivity and Hall-effect in a semiconductor.
- Illustrates the magnetic and dielectric materials applications.
- Apply the principles of semiconductors in various electronic devices.

List of Experiments

1. Determine the thickness of the wire using wedge shape method.
2. Determination of the radius of curvature of the lens by Newton's ring method.
3. Determination of wavelength of LASER source by plane diffraction grating method.
4. Determination of dispersive power of a diffraction grating.
5. Determination of dielectric constant by charging and discharging method.
6. Determination of magnetic field along the axis of a circular coil carrying current.
7. Determination of the resistivity of semiconductor by Four probe method.
8. Determination of the energy gap of a semiconductor.
9. Study the variation of B versus H by magnetizing the magnetic material (B – Hcurve).
10. Determination of the numerical aperture of a given optical fibre and hence to find its acceptance angle
11. Determination of dispersive power of prism using the spectrometer.
12. Determination of Hall voltage and Hall coefficient of a given semiconductor using Hall-effect.

Note: In the above list, out of 12experiments, any 10 experiments must be performed in a semester

Reference Books:

1. A Textbook of Practical Physics, S. Balasubramanian, M.N. Srinivasan 2nd Edition, 2017, S. Chand Publishers.

Course Outcomes:

At the end of the course, the students will be able to

1. Recognize the importance of optical phenomena like interference, diffraction, dispersion and their applications.
2. Calculate the variation of magnetic field by passing current through the circular coil.
3. Determine the energy gap of a p-n junction diode and dielectric constant of a given material.
4. Determine the wavelengths of different colours from mercury Vapour lamp by using diffraction grating.
5. Determine the acceptance angle and numerical aperture of an optical fibre.

6. Determine the energy loss in magnetic materials from B-H curve.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Problem Solving & Programming Lab

(Common to all Branches)

I B.Tech - I Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
R204GA05102	**ESC**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		0	0	3	1.5	40	60	100

Objectives

- To learn C Programming Language.
- To make the students solve problems, implement algorithms using C language.
- Identify methods appropriate for solving problems in C
- Apply user defined data types to solve specific problems
- Understand command line arguments.

List of Experiments

1. Write a C program which reverses the number.
2. Write a C program which finds the second maximum number among the given list of numbers.
3. Write aprogramwhichfindsthekthsmallestnumberamongthegivenlistofnumbers.
4. Write an algorithm and implement using C language the following exchanges
   a ←b←c←d.
5. Write aCProgramwhichcountsthenumberofpositiveandnegativenumbersseparately and also compute the sum of them.
6. Implement the C program which computes the sum of the first n terms of the series Sum = 1 – 3 + 5 -7 +9 and so on.
7. Write a C program which determines the numbers whose factorial values are between 5000 and 32565.
8. Write an algorithm and implement using a C program which finds the sum of the infinite series $1 - x^2/2! + x^4/4! - x^6/6! + ...$
9. Write a C program to print the sequence of numbers in which each number is the sum of the three most recent predecessors. Assume first three numbers as 0, 1 and1.
10. Implement a C program which converts a hexadecimal, octal and binary number to decimal number and vice versa.
11. Write an algorithm which computes all the factors between 1 to 100 for a given number and implement it using C.
12. Write an algorithm which computes the sum of the factorials of numbers between m and n.
13. Write a C program which reverses the elements of the array.
14. Implement a C program to perform the following operations on Matrices:
    a. Addition b. Subtraction c. Multiplication
15. Implement C program to perform string manipulations.
16. Given a list of n numbers, design an algorithm which prints the number of stars equivalent to the value of the number. The start for each number should be printed horizontally.
17. Implement the sorting algorithms: a. Insertion sort b. Exchange sort c. Selection sort

Partitioning sort.

18. Illustrate the use of auto, static, register and external variables.
19. Write a C program which takes two numbers as command line arguments and finds all the common factors of those two numbers.
20. Write a C program which sorts the strings using array of pointers. # The above list is not exhaustive. Instructors may add some experiments to the above list. Moreover, 50% of the experiments are to be changed every academic year. Instructors can choose the experiments, provided those experiments are not repetitions.

Reference Books:

1. YashavantKanetkar, "Let Us C", 16th edition, BPB Publications.
2. R.G. Dromey, "How to Solve it by Computer". 2014, Pearson.

Course Outcomes:

At the end of the course, the students will be able to

1. Select the right control structure for solving the problem.
2. Analyze different sorting algorithms.
3. Find solutions for computational problems.
4. Develop C programs which utilize the memory efficiently using programming constructs like pointers.
5. Develop logic for solving different problems.
6. Understand the command line arguments and its applications.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Transforms & Partial Differential Equations

(Common to all Branches)

I B.Tech - II Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
R204GA54201	**BSC**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		2	1	0	3	40	60	100

Objectives

- This course aims at providing the student with knowledge on different types of transforms and Partial differential equations.

Unit I - Laplace Transforms

Laplace transform of standard functions – Inverse transform – First shifting Theorem, Transforms of derivatives and integrals – Unit step function – Second shifting theorem – Dirac's delta function – Convolution theorem – Laplace transform of Periodic function. Differentiation and integration of transform – Application of Laplace transforms to ordinary differential equations of first and second order.

Learning Outcomes:

At the end of this unit, students will be able to

1. Understand Laplace transforms.
2. Apply properties of Laplace transforms.
3. Understand Laplace transforms of special functions.
4. Apply Inverse Laplace transforms.
5. Apply Laplace transforms to solve Differential Equations.

Unit II - Fourier series

Fourier series: Determination of Fourier coefficients – Fourier series – Even and Odd functions – Fourier series in an arbitrary interval – Even and odd periodic continuation – Half-range Fourier sine and cosine expansions- Parseval's formula- Complex form of Fourier series.

Learning Outcomes:

At the end of this unit, the students will be able to

1. Understand Fourier Series.
2. Determine Fourier Coefficients.
3. Understand Even and Odd functions.
4. Apply Fourier Series in arbitrary intervals.
5. Find Half range Fourier Sine and Cosine expansions.

Unit III – Fourier transforms

Fourier integral theorem (only statement) – Fourier sine and cosine integrals. Fourier transforms – Fourier sine and cosine transforms – Properties – Inverse transforms – Finite Fourier transforms.

Learning Outcomes:

At the end of this unit, the students will be able to

1. Calculate Fourier Sine and Cosine integrals.
2. Understand Fourier transforms.

3. Apply Fourier Sine and Cosine transforms in evaluating integrals.
4. Apply properties of Fourier transforms.

Understand Finite Fourier transforms.

Unit IV - Z-transforms

Z-transform – Inverse Z-transform – Properties – Damping rule – Shifting rule – Initial and final value theorems. Convolution theorem – Solution of difference equations by Z-transforms.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Understand Z-transforms.
2. Apply properties of Z-transforms.
3. Understand Initial and Final value theorems.
4. Prove Convolution theorem in Z-transforms.
5. Apply Z transforms to solve Difference Equations.

Unit V – Partial Differential Equations & Applications

Formation of partial differential equations by elimination of arbitrary constants and arbitrary functions – Method of separation of variables – Solutions of one-dimensional wave equation, heat equation and two-dimensional Laplace's equation under initial and boundary conditions.

Learning Outcomes:

At the end of this unit, the students will be able to

1. Form Partial Differential Equations.
2. Solve Partial Differential Equations.
3. Understand Initial and boundary condition problems.
4. Solve applications of Partial Differential Equations.

Text Books:

1. Higher Engineering Mathematics, B.S.Grewal, Khanna publishers, 1965, 44thedition.
2. Engineering Mathematics-II, E. Rukmangadachari& E. Keshava Reddy, Pearson publishers, 1st January2013.
3. 3. Engineering Mathematics Volume-II, by T.K.V. Iyengar, B. Krishna Gandhi, S. Ranganatham and M.V.S.S.N. Prasad, S. Chand publications, 2014, 16thedition.

Reference Books:

1. Advanced Engineering Mathematics by Erwin Kreyszig, Wiley India, 10th edition reprint 2016.
2. Higher Engineering Mathematics, by B.V. Ramana, Mc Graw Hill publishers, 32nd reprint2018.
3. Advanced Engineering Mathematics, by Alan Jeffrey, Elsevier, 19th June2001.

Course Outcomes:

At the end of the course, students will be able to

1. Solve different kinds of transforms and partial differential equations
2. Solve the differential equations of first and second order by applying Laplace transform method.
3. Apply Fourier sine and cosine series expansions to evaluate and interpret the given functions.
4. Apply Fourier sine and cosine transform to evaluate the given functions.

5. Solve the difference equations by applying Z-transform.
6. Solve one dimensional wave, heat and two-dimensional Laplace's equation under initial and boundary conditions.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Applied Chemistry

(Common to ECE, EEE, CSE, CSD & CSM)

I B.Tech - II Semester **SRIT R20**

Course Code	Category	Hours/Week			Credits	Maximum Marks		
		L	T	P	C	CIA	SEE	Total
R204GA51102	BSC	3	0	0	3	40	60	100

Objectives

- To familiarize engineering chemistry and its applications.
- To train the students on the principles and applications of electro chemistry and polymers.
- To impart the concept of soft and hard waters, softening methods of hard water.

Unit I - Water Quality and Treatment

Impurities in water, Hardness of water and its Units, Disadvantages of hard water, Estimation of hardness by EDTA method, Numerical problems on hardness, Estimation of dissolved oxygen, alkalinity, acidity, Water treatment for domestic purpose (Chlorination, Bleaching powder, Ozonization).

Industrial use of water: For steam generation, troubles of Boilers - Scale & Sludge, Priming and Foaming, Caustic Embrittlement and Boiler Corrosion.

External Treatment: Ion-Exchange and Permutit processes. Demineralisation of brackish water, Reverse Osmosis and Electro dialysis.

Learning Outcomes:

At the end of this unit, students will be able to

1. List the differences between temporary and permanent hardness of water.
2. Explain the principles of reverse osmosis and electro dialysis.
3. Illustrate problems associated with hard water - scale and sludge.
4. Explain the working principles of different Industrial water treatment processes.

Unit II - Electrochemistry and Applications

Electrodes – concepts, reference electrodes (Calomel electrode, Ag/AgCl electrode and glass electrode) electro chemical cell, Nernst equation, cell potential calculations, numerical problems, Potentiometry- potentiometric titrations (redox titrations), concept of conductivity, conductivity cell, conductometric titrations (acid-base titrations) ,photo voltaic cell–working and applications, photo galvanic cells with specific examples.

Primary cells – Zinc-air battery, Fuel cells -Hydrogen-Oxygen, methanol fuel cells – working of the cells. Secondary cells–lead acid and lithium-ion batteries-working of the batteries including cell reactions.

Learning Outcomes:

At the end of this unit, students will be able to

1. Apply Nernst equation for calculating electrode and cell potentials.
2. Differentiate between pH metry, potentiometric and conductometric titrations.
3. Explain the theory of construction of battery and fuel cells.
4. Solve problems based on cell potential.

Unit III - Fuels and Combustion

Classifications of Fuels – Characteristics of Fuels, Calorific Value– Units, Numerical Problems.

Solid Fuels: Coal-Classification and Analysis (proximate), Coke: Characteristics of metallurgical coke, Manufacture of Metallurgical Coke by Otto Hoffmann's by product oven process.

Liquid Fuels: Petroleum - Refining of Petroleum, Gasoline- Octane Number, Diesel -Cetane Number, Synthetic Petrol: Bergius Processes, Fischer Troph's synthesis Power Alcohol: Manufacture, Advantages and Disadvantages of Power Alcohol Gaseous Fuels: Natural gas, Producer gas, Water gas, Coal gas and Biogas.

Combustion: Basic principle and numerical problems.

Learning Outcomes:

At the end of this unit, the students will be able to

1. Solve the numerical problems based on Calorific value.
2. Select suitable fuels for IC engines.
3. Explain Calorific values, Octane Number, Refining of petroleum.

Unit IV - Polymer Chemistry

Introduction to polymers, functionality of monomers, chain growth and step growth polymerization, co-ordination polymerization with specific examples and mechanisms of polymer formation.

Plastics - Thermoplastics and Thermosetting, Preparation, Properties and applications of Bakelite, Nylons. Processing and vulcanisation of natural rubber. Elastomers–Buna-S, Buna-N:preparation, properties and applications. Conducting polymers – polyacetylene, polyaniline – mechanism of conduction and applications. Inorganic polymers (Polysilicon's)-preparation, properties and applications.

Learning Outcomes:

At the end of this unit, the students will be able to

1. Explain the different types of polymers and their applications.
2. Explain the preparation, properties and applications of Bakelite, Nylon-66 and Carbon fibres.
3. Describe the mechanism of conduction in conducting polymers.
4. Discuss Buna-S and Buna-N elastomers and their applications.

Unit V - Instrumental Methods and Applications

Electromagnetic spectrum. Absorption of radiation: Beer-Lambert's law. Principle and applications of pH metry, potentiometry, conductometry, UV-Visible, IR and NMR Spectroscopies. Principles of Gas Chromatography (GC) and High-Performance Liquid Chromatography (HPLC), separation of gaseous mixtures and liquid mixtures.

Learning Outcomes:

At the end of this unit, students will be able to

1. Explain the different types of spectral series in electromagnetic spectrum.
2. Understand the principles of different analytical instruments.
3. Explain the different applications of analytical instruments.

Text Books:

1. Engineering Chemistry, Jain and Jain, 16th Edition, 2013, Dhanpatrai.
2. Engineering Chemistry, K. N. Jayaveera, G. V. SubbaReddyand C. Ramachandraiah, First Edition, 2013, McGraw Hill Higher Education.

Reference Books:

1. Principles of Instrumental Analysis, Skoog and West, 6th Edition, 2007, Thomson.
2. A Text book of Engineering Chemistry,S. S. Dhara and Uma, 12th Edition, 2010, S. Chand Publication.
3. Engineering Chemistry, K. B. Chandra Sekhar, U. N. Das and Sujatha Mishra,
4. First Edition, 2010, SCITECH Publications India Pvt Limited.

Course Outcomes:

At the end of course, the students will be able to

1. Explain the analysis of water, boiler troubles and demineralization techniques.
2. Calculate the emf of the cells and batteries by Nernst equation.
3. Explain the construction and working of batteries and fuel cells, redox and conductometric titrations.
4. Discuss the types, preparation, properties and applications of solid, liquid and gaseous fuels.
5. Discuss the types, preparation and properties of polymers.
6. Discuss the principle and applications of pHmetry, potentiometry, conductometry, gas and HPLC Chromatography.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Basic Electrical & Electronics Engineering

(Common to MEC, CSE, CSD & CSM)

I B.Tech - II Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA02101	**ESC**	2	1	0	3	40	60	100

PART-A: Basic Electrical Engineering

Objectives

- To learn about basics of electrical circuits.
- To understand DC and AC electrical circuit analysis.
- To understand working principle of transformers and electrical machines.
- To learn about Measuring Instruments.

Unit I –DC & AC Circuits

Electrical circuit elements (R, L and C) – Kirchhoff's laws - Series and parallel connection of resistances with DC excitation. Superposition Theorem, Representation of sinusoidal waveforms - peak and rms values - phasor representation - real power - reactive power - apparent power - power factor - Analysis of single-phase AC circuits consisting of RL, RC& RLC series circuits.

Learning Outcomes:

1. To learn about Kirchhoff's laws.
2. Able to analyze simple electric circuits with DC excitation.
3. Able to apply network theorems to simple circuits.
4. Able to analyze single phase AC circuits consisting of series RL, RC & RLC combinations.

Unit II - DC & AC Machines

Principle and operation of DC Generator - EMF equations - OCC characteristics of DC generator, principle and operation of DC Motor – Performance Characteristics of DC Motor - Speed control of DC Motor, Principle and operation of Single-Phase Transformer - OC and SC test on transformer, principle and operation of Induction Motor [Elementary treatment only].

Learning Outcomes:

1. To distinguish between principle and operation of DC Generator & DC Motor.
2. To Learn about the speed control of DC Motor.
3. To distinguish between operation of Transformer and Induction motor.

Unit III - Measuring Instruments

Introduction, classification of instruments, operating principles, essential features of measuring instruments, Moving coil permanent magnet (PMMC) and Moving Iron instruments(Voltmeters and Ammeters)- Extension of range of the meters.

Learning Outcomes:

1. To distinguish between principle and operation of Moving Iron and Moving Coil Instruments.
2. To learn about the need to Extend the range of the meters.

Text Books:

1. D. P. Kothari and I. J. Nagrath - "Basic Electrical Engineering" - Tata McGraw Hill - 2010.
2. V.K. Mehta &Rohit Mehta, "Principles of Power System" – S. Chand –2018.

Reference Books:

1. L. S. Bobrow - "Fundamentals of Electrical Engineering" - Oxford University Press - 2011.
2. E. Hughes - "Electrical and Electronics Technology" - Pearson -2010.

Course Outcomes:

After the completion of the course, the students will be able to

1. Calculate the electrical quantities in DC and AC circuits.
2. Determine the performance characteristics of DC and AC machines.
3. Explain the working principle and operation of measuring instruments.

PART-B : Basic Electronics Engineering

Objectives:

- To know about comprehensive idea of working principle, operation and applications of PN junction &Zener diodes, BJT, FET, MOSFET and Operational amplifier.
- To learn about fundamentals of Digital electronics.
- To teach efficacy of electronic principles which are pervasive in engineering applications.

Unit I – Analog Electronics-I

Overview of Semiconductors, PN junction Diode, Characteristics of a Diode, Applications of Diode as switch and rectifier, Zener diode, Zener diode act as regulator.

Learning Outcomes:

1. Able to understand operation and characteristics of Diodes.
2. Able to make use of diodes in simple, typical circuit applications.

Unit II - Analog Electronics-II

BJT construction, operation, configuration and characteristics, Operational Amplifiers: Introduction, block diagram, basic op-amp circuits, Application of Operational Amplifiers: Inverting, Non Inverting, Adder, subtractor and voltage follower.

Learning Outcomes:

1. Able to distinguish operation and characteristics of BJT.
2. Able to make use of transistors in simple, typical circuit applications.
3. Able to understand operation of basic op-amp circuits.

Unit III – Digital Electronics

Introduction, Switching and Logic Levels, Digital Waveform, characteristics of digital ICs, logic gates: AND, OR, NOT, NOR and NAND gates, number system: Binary, Octal, Decimal and Hexa decimal and their conversions.

Learning Outcomes:

1. Able to understand different logic gates using truth table.
2. Able to understand the concepts of number system.

Text Books:

1. D.P. Kothari, I.J. Nagrath, "Basic Electronics", 2nd edition, McGraw Hill Education (India) Private Limited.

2. S.K. Bhattacharya, "Basic Electrical and Electronics Engineering", 2nd edition, Pearson India Private Limited.

Reference Books:

1. R. Muthu Subramanian, S. Salivahanan, "Basic Electrical and Electronics Engineering", Tata McGraw-Hill Education, Reprint2012.
2. David Bell, "Electronic Devices and Circuits", Oxford University Press, 5th Edn.,2008.

Course Outcomes:

After the completion of the course, the students will be able to

1. Explain the concepts of analog and digital electronic devices.
2. Determine the performance characteristics of diode, transistor and op-amp circuit.
3. Understand the concepts of logic gates and number system.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Data Structures

(Common to EEE, MEC, ECE & CSE, CSD &CSM)

I B.Tech - II Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
R204GA05201	**ESC**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		3	0	0	3	40	60	100

Objectives

- To teach the representation of solution to the problem using algorithm.
- To explain the approach to algorithm analysis.
- To introduce different data structures for solving the problems.
- To demonstrate modeling of the given problem as a graph.
- To elucidate the existing hashing techniques.

Unit I – Introduction to Data Structures

Algorithm Specification, Performance Analysis, Performance Measurement. Arrays: Arrays, Dynamically Allocated Arrays, Structures and Unions.

Sorting: Motivation, Quick sort, How fast can we sort, Merge sort, Heap sort.

Learning Outcomes:

At the end of this unit, the students will be able to

1. Analyze the given algorithm to find the time and space complexities.
2. Select appropriate sorting algorithm.
3. Design a sorting algorithm.

Unit II - Stack, Queue and Linked lists

Stacks, Stacks using Dynamic Arrays, Queues, Circular Queues Using Dynamic Arrays, Evaluation of Expressions, Multiple Stacks and Queues. Linked lists: Singly Linked Lists and Chains, Representing Chains in C, Linked Stacks and Queues, Additional List Operations, Doubly Linked Lists.

Learning Outcomes:

At the end of this unit, the students will be able to

1. Evaluate expressions.
2. Develop the applications using Stacks and Queues.
3. Construct the Linked lists for various applications.

Unit III –Trees

Introduction to Non-linear Data structures: Binary Trees, Binary Tree Traversals, Additional Binary Tree Operations, Binary Search Trees, Counting Binary Trees, Optimal Binary Search Trees, AVL Trees. B-Trees: B Trees, B+ Trees.

Learning Outcomes:

At the end of this unit, the students will be able to

1. Explain the concept of a Tree.
2. Compare different Tree structures.
3. Apply Trees for indexing.

Unit IV - Graphs and Hashing

Graphs: The Graph Abstract Data Type, Elementary Graph Operations, Minimum Cost Spanning Trees, Shortest Paths and Transitive Closure.
Hashing: Introduction to Hash Table, Static Hashing and Dynamic Hashing.

Learning Outcomes:

At the end of this unit, the students will be able to

1. Recognize the importance of Graphs in solving real world problems.
2. Apply various graph traversal methods to applications.
3. Design a minimum cost solution for a problem using Spanning trees.
4. Select the appropriate hashing technique for a given application.
5. Design a hashing technique.

Unit V - Files and Advanced Sorting

File Organization: Sequential File Organization, Direct File Organization, Indexed Sequential File Organization.
Advanced sorting: Sorting on Several keys, List and Table sorts, Summary of Internal sorting and External sorting.

Learning Outcomes:

At the end of this unit, the students will be able to

1. Organize data in the form of Files.
2. Apply sorting on large amount of data.

Text Books:

1. Ellis Horowitz and SartajSahni, "Fundamentals of Data Structures in C", 2nd Edition, Galgotia Book Source, Pvt. Ltd.,2004.
2. Alan L. Tharp, "File Organization and Processing", Wiley and Sons,1988.

Web Resources:

https://www.greatlearning.in/academy/learn-for-free/courses/data-structures-in-c
https://www.tutorialspoint.com/data_structures_algorithms/

Reference Books:

1. D. Samanta, "Classic Data Structures", 2nd Edition, Prentice-Hall of India, Pvt. Ltd., India,2012.
2. Peter Bras, "Advanced Data Structures", Cambridge University Press,2016.
3. Richard F.Gilberg, BehrouzA.Forouzan, "Data Structures A Pseudo code Approach with C", Second Edition, Cengage Learning2005.

Course Outcomes:

At the end of the course, the students will be able to

1. Select appropriate Data Structure for solving a problem by organizing the data.
2. Find the performance of an algorithm.
3. Organize the data using line
4.
5. ar data structures.
6. Use non-linear data structures to maintain the data.
7. Illustrate hash tables and graphs for quick data manipulation.

8. Develop Algorithm for sorting large files of data.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Engineering Workshop Practice

(Common to EEE, ECE, CSE, CSD & CSM)

I B.Tech - II Semester **SRIT R20**

Course Code	Category	Hours/Week			Credits	Maximum Marks		
		L	T	P	C	CIA	SEE	Total
R204GA03104	ESC	1	0	4	3	40	60	100

Objectives

- To familiarize students with wood working, sheet metal operations, electrical house wiring, Arc welding skills, CNC machines and pick and place robot.

List of Experiments

Trade 1: Wood Working

Familiarity with different types of woods and tools used in wood working and make following joints.

a) Half – Lap joint
b) Mortise and Tenon joint
c) Corner Dovetail joint or Bridle joint

Trade 2: Sheet Metal Working

Familiarity with different types of tools used in sheet metal working, Developments of following sheet metal job from GI sheets.

a) Tapered tray b) Conical funnel c) Elbowpipe d) Brazing

Trade 3: Electrical Wiring

Familiarities with different types of basic electrical circuits and make the following connections

a) Parallel and series b) Two-way switch c) God own lighting
d) Tube light e) Three phase motor f) Soldering of wires

Trade 4: Arc Welding

Familiarity with different types of tools used in Arc welding, and make the following joints

a) Lap joint b) Butt joint.

Trade 5: CNC demonstration

Demonstration of CNC lathe, CNC Milling and 5 Axis robot arm.

Text Books:

1. K. Venkata Reddy Workshop Mannual 6th Ed., B.S. Publishers,2013.
2. B.L. Juneja Workshop practice 1stEd., Cengage,2015.

Course Outcomes:

At the end of the course, the students will be to

1. Use tools for various workshop operations.
2. Apply wood working skills in real world applications.
3. Build different parts with metal sheets in real world applications.
4. Apply different types of basic electric circuit connections.
5. Demonstrate soldering.
6. Demonstrate CNC machine and industrial manipulator.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Basic Electrical & Electronics Engineering Lab

(Common to ME, CSE, CSD& CSM)

I B.Tech - II Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA02102	**ESC**	0	0	3	1.5	40	60	100

PART-A: Electrical Engineering Lab

Objectives:

- To verify Kirchhoff's laws.
- To verify Superposition theorem.
- To learn performance characteristics of DC Machines.
- To perform Open Circuit & Short Circuit test on 1- Phase Transformer.
- To Study the I – V Characteristics of Solar PV Cell.

List of Experiments

1. Verification of Kirchhoff laws.
2. Verification of Superposition Theorem.
3. Open Circuit characteristics of a DC Shunt Generator.
4. Speed control of DC Shunt Motor.
5. OC & SC test on1 – Phase Transformer.
6. Brake test on 3 - Phase Induction Motor.
7. I – V Characteristics of Solar PV cell.
8. Brake test on DC Shunt Motor.

Note: At least 5 of the above experiments are to be conducted.

Course Outcomes:

After the completion of this course, the students will be to

1. Verify Kirchhoff's Laws & Superposition theorem.
2. Perform testing on AC and DC Machines.
3. Study I – V Characteristics of PV Cell.

PART-B: Electronics Engineering Lab

Objectives:

- Describe construction, working and characteristics of Diodes, Transistors and Operational Amplifiers.
- Demonstrate how electronic devices are used for applications such as rectification, switching and amplification.
- Build different building blocks in Digital electronics using logic gates.
- Explain functionality of flip-flops, shift registers and counters for data processing applications.
- Explain functioning of various communication systems.

List of Experiments:

1. Draw and study the characteristics of Semi-conductor diode and Zener Diode.
2. Draw and study the input and output characteristics of Transistor in Common Emitter configuration.

3. Draw and study the static and transfer characteristics of FET in Common Source Configuration.
4. Construct half wave and full wave rectifier circuits. Find ripple factor and plot their output waveforms with and without filter.
5. Study the application of Op-amp as an Inverting amplifier, Non-inverting amplifier, Voltage follower, Summer and Subtractor.
6. Realization of logic gates, AND, OR, NOT, NAND, NOR and XOR.
7. Realization of Adders, Multiplexers and Decoders using logic gates.
8. Realization of flip-flops using logic gates.

Note: At least 5 of the above experiments are to be conducted.

Course Outcomes:

At the end of the course, the students will be to

1. Describe construction, working and characteristics of Diodes, Transistors and Operational Amplifiers.
2. Build different building blocks in digital electronics using logic gates.
3. Understand electronic devices to be used in rectifier circuits.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Applied Chemistry Lab

(Common to EEE, ECE, CSE, CSD & CSM)

I B.Tech - II Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
R204GA51104	**BSC**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		0	0	3	1.5	40	60	100

Objectives

- To Verify the fundamental concepts with experiments

List of Experiments

1. Determination of total hardness of water by EDTA method.
2. Determination of Copper by EDTA method.
3. Determination of Alkalinity of Water.
4. Estimation of Dissolved Oxygen by Winkler's method.
5. Conduct metric estimation of strong acid using standard sodium hydroxide solution
6. pH metric titration of strong acid vs strong base.
7. Estimation of Ferrous Iron by Dichrometry.
8. Potentiometry - determination of redox potentials and emfs.
9. Determination of Strength of an acid in Pb-Acid battery.
10. Preparation of a polymer.

Reference Books:

1. Vogel's Text book of Quantitative Chemical Analysis, J Mendham *et al*, Sixth Edition, 2012, Pearson Education.
2. Chemistry Practical– Lab Manual, K. B. Chandra Sekhar, G. V. Subba Reddy and K. N. Jayaveera, First edition, 2014, SM Enterprises.

Course Outcomes:

At the end of the course, the students will be to

1. Determine the redox potentials and emf of solutions by potentiometry.
2. Determine the weight of prepared organic polymers.
3. Estimate the hardness, alkalinity and dissolved oxygen of water.
4. Estimate the amount of Cu+2, Fe+2 in given samples.
5. Determine the strength of acids by conductometer & PHmeter.
6. Determine the strength of acid in Pb acid battery.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Data Structures Lab

(Common to EEE, MEC, ECE, CSE, CSD and CSM)

I B.Tech - II Semester **SRIT R20**

Course Code	Category	Hours/Week			Credits	Maximum Marks		
		L	T	P	C	CIA	SEE	Total
R204GA05202	ESC	0	0	3	1.5	40	60	100

Objectives

- To introduce the different data structures.
- To elucidate how the data structure selection influences the algorithm complexity.
- To explain the different operations that can be performed on different data structures.
- To introduce the different search and sorting algorithms.

List of Experiments

1. Write a program on string operations using array of pointers.
2. Write a program Searching Algorithms (With the Number of Key Comparisons) Sequential, Binary and Fibonacci Search Algorithms.
3. Write a program Sorting Algorithms: Insertion Sort, Selection Sort, Shell Sort, Bubble Sort, Quick Sort, Heap Sort, Merge Sort, and Radix Sort. Using the system clock, compute the time taken for sorting of elements. The time for other operations like I/O etc. should not be considered while computing time.
4. Write a program Singly Linked List, Doubly Linked List and Circular Linked List.
5. Write a program stack implementation using arrays.
6. Write a program Stack implementation using Linked lists.
7. Write a program queue using arrays. Implement different forms of queue. While implementing you should be able to store elements equal to the size of the queue. No positions should be left blank.
8. Write a program queue using Linked lists.
9. Write a program binary search tree, performing operations insertion, deletion and traversal.
10. Write a program breadth first search.
11. Write a program depth first search.
12. Write a program traveling salesman problem.
13. Write a program file operations.
14. Write a program indexing of a file.
15. Write a program reversing the links (not just displaying) of a Linked list.
16. Consider a linked list consisting of name of a person and gender as a node. Arrange the linked list using 'Ladies first' principle. You may create new Linked lists if necessary.
17. An expression can be represented in three ways: infix, prefix and postfix. All the form are necessary in different contexts. Write modules to convert from one form to another form.
18. Table can be defined as a collection of rows and columns. Each row and column may have a label. Different values are stored in the cells of the table. The values can be of different data types. Numerical operations like summation, average etc., can be performed on rows/columns which contain numerical data. Such operations are to be prevented on data which is not numeric. User may like to insert row/columns in the already existing table. User may like to remove row/column. Create table data type and support different operations on it.

Reference Books:

1. Data Structures using C and C++, Yedidyah Langsam. Moshe J. Augenstein Aaron M. Tenenbaum, 2nd Edition, PHI.
2. Data Structures using C & C++, Rajesh K.Shukla,Wiley-India.
3. ADTs, Data Structures and Problem Solving with C++, Larry Nyhoff, Pearson.

Course Outcomes:

At the end of the course, the students will be able to

1. Select the data structure appropriate for solving the problem.
2. Implement searching and sorting algorithms.
3. Design new data types.
4. Illustrate the working of stack and queue.
5. Organize the data in the form of files.
6. Implement applications of graphs and trees.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Probability & Statistics

(Common to CSE, CSM, CSD, ME & CIV)

II B.Tech - I Semester **SRIT R20**

Course Code	Category	Hours/Week			Credits	Maximum Marks		
		L	T	P	C	CIA	SEE	Total
R204GA54302	BSC	3	0	0	3	40	60	100

Objectives

This course aims at providing the student with the knowledge on

- The theory of Probability and random variables.
- Usage of statistical techniques like testing of hypothesis, testing of significance, chi-square test and basic concepts of Queuing theory.

Unit I - Elementary Statistics

Basics to statistics- definition-advantages-limitations-frequency distribution tables-Arithmetic mean, median, mode for grouped and ungrouped data-variance, standard deviation, co-efficient of variation. Correlation –properties, correlation co-efficient-Regression-properties-Regression co- efficient- relation between correlation co-efficient and Regression co-efficient.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Understand the fundamentals of Statistical techniques.
2. Evaluate the measures of central tendency for grouped and ungrouped data.
3. Understand the Correlation & regression and their properties.

Unit II - Probability theory

Probability, probability axioms, addition law and multiplicative law of probability, conditional probability, Baye's theorem, random variables (discrete and continuous), probability density functions, properties, mathematical expectation.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Define the terms trial, events, sample space, probability, and laws of probability.
2. Apply Baye's theorem to real time problems.
3. Explain the notion of random variable, distribution functions and expected value.

Unit III -Random variables & Distributions

Probability distribution - Binomial, Poisson approximation to the binomial distribution and normal distribution-their properties-Uniform distribution-exponential distribution.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Understand the concept of Probability Distributions.
2. Apply Binomial and Poisson distributions for real data to compute probabilities, theoretical frequencies.
3. Understand the concept of Normal Distributions and their properties.

Unit IV - Testing of Hypothesis

Estimation-parameters, statistics, sampling distribution, point estimation, Formulation of null hypothesis, alternative hypothesis, the critical and acceptance regions, level of significance, two types of errors and power of the test. Large Sample Tests: Test for single proportion, difference of proportions, test for single mean and difference of means. Confidence interval for parameters in one sample and two sample problems.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Explain the concept of estimation, interval estimation and confidence intervals.
2. Apply the concept of hypothesis testing for large samples.

Unit V - Testing of significance

Student t-distribution (test for single mean, two means and paired t-test), testing of equality of variances (F-test), $\chi 2$ - test for goodness of fit, $\chi 2$ - test for independence of attributes. Queuing Theory: Pure Birth and Death process, M/M/1 & M/M/S & related problems.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Apply the concept of testing hypothesis for small samples.
2. Estimate the goodness of fit.
3. Explain the Concept of Queuing theory & apply in real time problems.

Text Books:

1. Grewal B.S., Higher Engineering Mathematics, Khanna publishers, 44thEdition,2018.
2. Erwin Kreyszig, Advanced Engineering Mathematics, Wiley India, 9thEdition,2007.

Reference Books:

1. Ramana B.V., Higher Engineering Mathematics, Mc Graw Hill publishers,2008.
2. Alan Jeffrey, Advanced Engineering Mathematics, Elsevier,2002.

Course Outcomes:

At the end of the course, student will be able to

1. Understand the concepts of probability, sampling distributions, test of hypothesis and Queuing theory.
2. Explain the characteristics through correlation and regression tools.
3. Apply Probability theory to find the chances of happening of events.
4. Understand various probability distributions and calculate their statistical constants.
5. Solve the problems on testing of hypothesis in large samples.
6. Solve the problems on testing of hypothesis in small samples and Queuing models.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Database Management Systems

(Common to CSE, CSM & CSD)

II B.Tech - I Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA05301	**PCC**	3	0	0	3	40	60	100

Objectives

- To understand design and implementation of a database system.
- To understand database designs, database modeling.
- To understand the management of a database.
- To emphasize the importance of normalization in databases.
- To facilitate students in Database design

Unit I – Introduction to Databases

Database System Applications, Purpose of Database Systems, Views of Data, Database Languages, Relational Databases. Database Design, Database Storage and Querying, Database Architecture, Database Users and Administrator and History of Databases.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Determine the levels of data abstractions and data independence.
2. Illustrate the process of Database Design.
3. Perform Data manipulation operations.

Unit II – Relational databases

Relational Model: Structure of Relational Databases, Database Schema, Keys, Relational Query languages, Relational Algebra, Tuple Relational Calculus and Domain Relational calculus, Relational Operations.

SQL: SQL data definition, Basic Structure of SQL Queries, Additional Basic operations, Set Operations, Null Values, Aggregate Functions, Nested Queries, Modification of databases, Join Expressions, views, Transactions, Integrity Constraints, SQL data types and schemas, Authorization, Functions and procedures, Triggers, Recursive Queries.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Identify the keys for a database schema.
2. Construct an SQL query Using DML and DDL Statements and views.
3. Implement and construct functions, procedures and triggers for a Database.

Unit III – Database Design

Database Design with E-R Model: Overview of the Design Process, The Entity-Relational Model, Constraints, Removing Redundant Attributes in Entity set, Reduction to Relational Schema, Entity-Relationship Design issues, Extended E-R features, Alternative Notions for Modeling, Other accepts of Database Design.

Relational Database Design: Features of Good Relational Designs, Atomic Domains and First Normal Form, Decomposition using Functional Dependencies, Decomposition using multi valued Dependencies. Database Design Process, Modeling Temporal Data.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Construct E-R Model for a given specifications of an organization.
2. Apply Database Refinement using Normal Forms.

Unit IV – Transaction Management

Fundamentals of Transaction: Transaction Concept, A simple Transaction Model, storage Structure, Transaction Atomicity and Durability, Transaction Isolation, Serializability, Transactions as SQL Statements.

Concurrency Control in Transactions: Lock-Based Protocols, Deadlock Handling, Multiple Granularity, Timestamp-Based Protocols, Validation Protocols, Multi version Schemes. **Recovery system:** Failure Classification, Storage, Recovery and Atomicity, Recovery Algorithm.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Illustrate properties of a transaction and transaction model.
2. Determine whether the given schedule is a concurrent schedule or not.
3. Understand the importance of Database recovery Mechanisms.

Unit V – Data Storage and Querying

Storage and File Structure: Overview of physical storage media, magnetic disk and flash storage, RAID, tertiary storage, File Organization.

Indexing and hashing: Ordered Indices, B+ Tree Index Files, Extensions, Multiple- Key Access, Static hashing, Dynamic Hashing, Comparison of Ordered Indexing and Hashing, Bitmap Indices, Index Definition in SQL, Query processing and Optimization.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Demonstrate physical media storage devices.
2. Illustrate various file Organization methods.
3. Perform operations of indexing and hashing.

Text Books:

1. Abraham Silberschatz, Henry F. Korth,Sudarshan S., "Database System Concepts", McGraw-Hill,6th Edition 2011.
2. Raghurama Krishan, "Database Management Systems", McGraw Hill,2003.

Reference Books:

1. Ullman J. D., "Principles of Database and Knowledge – Base Systems", Vol 2 Computer Science Press,1989.
2. Elmasri R., and Navathe S., "Fundamentals of Database Systems", Pearson Education, 5th Edition, 2007.
3. Serge Abiteboul, Richard Hull, VictorVianu, "Foundations of Databases", Addison-Wesley, Reprint 1996.

Course Outcomes:

At the end of the course, student will be able to

1. Describe the design concepts of database management systems.
2. Demonstrate the fundamentals of database systems.
3. Design a database using Relational Model and SQL queries.

4. Design a database with E-R and Relational Model using normalization.
5. Illustrate a transaction model with various concurrency control Protocols and Recovery Mechanisms.
6. Illustrate File Organization methods and indexing techniques.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

English Language & Employability Skills for Engineers

(Common to EEE, ECE, CSE, CSM & CSD)

II B.Tech - I Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA52301	**HSMC**	3	0	0	3	40	60	100

Objectives

- Focus on appropriate reading strategies to become more competent, efficient and perceptive academic reader who is able to comprehend the contents and main ideas of what is read.
- Help to speak comprehensibly at the advanced level like group discussions, technical presentations, small talks.
- Facilitate necessary listening skills in order to follow and comprehend discourse such as lectures, conversations, interviews and discussions.
- Provide knowledge of grammatical structures and vocabulary to use meaningfully and appropriately in written and spoken forms.
- Impart effective writing strategies and demonstrate the same in preparing resume, E-mail, project proposal useful for both academic and professional careers.

UNIT I – Lexical Competence and Effective Business Writing skills

Listening: Comprehend the information by listening to phone messages, orders and lectures.

Vocabulary: Academic and Technical vocabulary Usage.

Grammar: Framing questions open and closed type and Cause and Effect.

Writing: Writing Agenda and Minutes; Writing Notices and Memos.

Speaking: Small talks.

Learning Outcomes:

At the end of the Unit, the student will be able to

1. Improve the comprehensive ability through listening various audio and visual aids.
2. Use academic and technical vocabulary to face competitive exams and improve lexical competence.
3. Demonstrate the ability to use cause and effect phrases and frame open and closed end questions for effective written communication.

UNIT II – Professional Email Writing Skills and Video Conferencing

Listening: Predicting the content by listening to short audio clips.

Vocabulary: Academic and Technical Collocations Usage.

Grammar: Report speech.

Writing: Writing Professional E-mails in English.

Speaking: Conducting Meetings and Video Conferencing.

Learning Outcomes:

At the end of the Unit, the student will be able to

1. Comprehend and predict the content by listening to short audios and videos.
2. Use technical collocations; improve lexical competence to communicate ideas more effectively.
3. Recognize the importance of using reported speech.

UNIT III – Resume Preparation and Non-verbal Cues

Reading: Summarizing, Paraphrasing, and Note making

Vocabulary: One word substitutions.

Grammar: Degrees of comparison and simple, compound and complex sentences.

Writing: Writing Winning Resumes and Cover letters.

Speaking: Advanced Conversation techniques and Facial Expressions.

Learning outcomes:

At the end of the Unit, students will be able to

1. Evaluate research and technical articles by using summarizing, paraphrasing and note making strategies.
2. Apply appropriate substitutions while writing technical texts.
3. Identify the components and use of adjectives in the sentence.

UNIT IV – Academic Writing and Presentation Skills

Reading: Critical Reading.

Vocabulary: Idiomatic expressions.

Grammar: Compare and contrast using connectives.

Writing: Effective Technical Paper Writing.

Speaking: Technical presentations-Oral/Power Point Presentations and Kinesics.

Learning outcomes:

At the end of the Unit, students will be able to

1. Enhance logical thinking by comprehending technical texts through critical reading.
2. Use Idiomatic expressions to exhibit their lexical competence and get thorough knowledge in the usage of different forms of sentences.

UNIT V – Employability Skills: Group Discussions and Interview Etiquettes

Vocabulary: Academic Vocabulary Assessment.

Grammar: Common errors in a sentence.

Writing: Project Proposal Writing.

Speaking: Group discussion etiquette, Interview etiquette and Proximics.

Learning outcomes:

At the end of the Unit, students will be able to

1. Take assessment in academic vocabulary by exhibiting his lexical competence.
2. Identify and rectify the common errors in sentences for better communication.
3. Write proposals for various funding projects to get accomplish their research ideas.

Text Books:

1. Ashrif Rizvi, Effective Technical Communication, Tata Mc Graw Hill, 2019.
2. Meenakshi Raman and Sangeetha Sarma, Technical Communication, Oxford University Press, Third Edition, 2015.

Reference Books:

1. Wood F.T., Remedial English Grammar, Macmillan, 2007.
2. Aysha Vishwamohan, English for Technical Communication for Engineering students, Tata Mc Graw Hill, 2009.
3. Sanjay Kumar and Pushpalatha, Communication Skills, Oxford University Press, 2011.
4. Exercises in Spoken English Parts I-III, CIEFL, Oxford University Press, Hyderabad.
5. Michael Swan, Practical English Usage, OUP, 1995.
6. E. Suresh Kumar & P. Sreehari, Communicative English, Orient Blackswan, 2009.

Course Outcomes:

At the end of the course, student will be able to

1. Remember the topic, the content and piece of information by listening.
2. Understand and produce appropriate academic vocabulary in oral and written communication.
3. Apply relevant formats of resume writing, E-mails and project proposals.
4. Analyze the use of grammar to construct meaningful sentences with suitable word choices.
5. Evaluate comprehensive skills through listening and reading prescribed text book.
6. Create opportunities by enhancing speaking skills to perform in small talks, group discussions and Interviews.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Object Oriented Programming

(Common to CSE, CSM & CSD)

II B.Tech - I Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA05302	**PCC**	3	0	0	3	40	60	100

Objectives

- Study the syntax, semantics and features of Java Programming Language.
- Learn the method of creating Multi - threaded programs and handle exceptions.
- The model of object-oriented programming: abstract data types, encapsulation, inheritance and polymorphism
- Fundamental features of an object-oriented language like Java: object classes and interfaces, exceptions and libraries of object collections
- Learn Java features to create GUI applications & perform event handling.

Unit I – Introduction to Java, Data types, Arrays and Variables, Operators, Control Statements, Introducing Classes

Introduction to Java: Object Oriented Programming, History and Evolution of java, Java's magic: The byte code, Java Buzzwords, Java Keywords, The Java class Libraries.

Data Types, Operators and Control Statements: Java Data Types, Variables and Constants, Naming Conventions, Type conversion and casting, Arrays, Operators & Expressions, Java Control Statements.

Introducing Classes and Methods: Classes and Objects, Introducing Methods, Constructors, this Keyword, Garbage Collection. Overloading Methods and Constructors, Argument passing, Recursion, Introducing Access Control, understanding static, Command Line Arguments, Exploring the String class.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Classify suitable data type for the given problem scenario.
2. Use control statements to affect the program execution flow.
3. Use language constructs for efficient problem solving.

Unit II – Inheritance, Exception Handling

Inheritance: Basics, super keyword, method overriding, dynamic method dispatch, Abstract classes, using final with inheritance, Introducing Nested and Inner classes.

Exception Handling: Fundamentals, Exception Types, Using try and catch, Multiple catch clauses, Nested try statements, throw, throws, finally, Java Built-in Exceptions, Creating user-defined exceptions.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Implement concepts of inheritance to find solution to the given problem.
2. Use exception handling mechanism to handle runtime errors.

Unit III – Packages, Interfaces and Multithreading

Packages: Basics, Access protection, Importing Packages, Creating and Importing User-defined Packages.

Interfaces: Declaring, Implementing and Extending Interfaces, using static methods in an Interface, using final keyword in interfaces.

Multithreaded Programming: Multithreading in Java, The Java Thread Model, Life Cycle of a Thread, The main thread, Creating Thread, Creating Multiple Threads, Thread Priorities, Synchronization, Inter Thread Communication, Suspending, resuming and stopping threads, Obtaining a thread state, The finalize () method.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Understand interface and its methods.
2. Analyze the suitable packages for the given problem scenario.
3. Explain thread and its methods.

Unit IV – Collections Framework & Introduction to AWT

Collections Framework: Overview, Collection Interfaces, Collection Classes. Working with Maps, Comparators.

Introduction to AWT: Windows, Graphics and Text

AWT classes, window fundamentals, frame windows, creating and displaying information within a window, Graphics, Color, Fonts, Managing text output using Font Metrics.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Use the prepackaged data structure to design collection framework programs.
2. Understand AWT fundamentals.
3. Use AWT framework to sketch a window.

Unit V – Introduction to AWT, Event Handling, Swings

Event Handling in Java: The Delegation Event Model, Event Classes and Event Listener Interfaces.

AWT Controls, Layout Managers, and Menus: AWT Control Fundamentals, Labels, Buttons, Check Boxes, Checkbox Group, Choice Controls, Lists, Scroll Bars, TextField and TextArea, Layout Managers, Menu Bars and Menus, Dialog Boxes, File Dialog.

Swings: Swing Features, MVC Connection, Components and Containers, JLabel, ImageIcon, JTextField, Swing Buttons, Check Boxes, Radio Buttons, JTabbedPane, JScrollPane, JList, JComboBox, JTree, and JTable.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Analyze AWT controls for the given problem.
2. Use Swing framework to sketch a window.

Text Books:

1. "The Complete Reference -Java", Herbert Schildt, Mc GRAW HILL Edition, 9th Edition, 2016.
2. "Java – How to Program", Paul Deitel, Harvey Deitel, PHI, 8thEdition, 2009.

Reference Books:

1. "A Programmers Guide to Java SCJP", Third Edition, Mughal, Rasmussen, Pearson,2009.
2. "Programming with Java" T.V.Suresh Kumar, B.Eswara Reddy, P.Raghavan Pearson Edition, 2011.
3. "Java Fundamentals - A Comprehensive Introduction", Herbert Schildt and DaleSkrien, Special Indian Edition, McGrawHill,2013.

Course Outcomes:

At the end of the course, student will be able to

1. Describe elements of Java, arrays and overloading.
2. Develop programs using type casting, type promotion and control statements for efficient problem solving.
3. Implement inheritance and exception handling for problem solving.
4. Implement threaded programming and usage of interfaces & packages.
5. Develop programs using collection framework and AWT frame work.
6. Develop programs using layout manager, Swing frame work and AWT controls suitable for the given problem scenario.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Software Engineering

(Common to CSE, CSM & CSD)

II B.Tech - I Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA05303	**PCC**	3	0	0	3	40	60	100

Objectives

- To understand the concepts of software engineering, requirement models, design models, SCM, different kinds of risks, project estimations and software testing techniques.
- To choose suitable software process model for a given problem scenario.
- To choose software requirement model in various scenarios.
- To apply architectural and component-level designs in various software applications.
- To use software testing techniques in order to debug developed software product.

Unit I: Basics of Software models and development

Evolving role of software, Nature of Software in web apps, Software a crisis on the Horizon, Software Myths.

Software models: Software process, Software process models: The linear sequential model, The prototyping model, The RAD model.

Evolutionary models: Specialized Process Models, Process product.

Agile Development: Agility, Agility and the cost of change, Agile process, Extreme Programming, Other Agile Process Models.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Learn definition, importance of software engineering and role of software engineering in development of web apps.
2. Discuss software myths.
3. Classify software process models.

Unit II: Software Process and project Planning

Measures, Metrics, and Indicators, Metrics in the Process and Project Domains ,Software Measurement, Reconciling Different Metrics Approaches, Metrics for software quality, Managing validation.

Project Planning objectives Resources, Software project estimation, Empirical estimation models, Automated estimation tools.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Learn the basics of Measures, Metrics, and Indicators and their approaches.
2. Discuss the importance of project planning in software development lifecycle.
3. Use automated estimation tools in order to find estimation time to complete a project.

Unit III: Risk analysis and Software quality assurance

Software risks, Risk identification, Risk projection, Risk refinement, safety risks and hazard,

RMMM plans.

Project scheduling and tracking**:** Defining task set, Defining task network, scheduling, earned value analysis, Error tracking, project plan.

Quality concepts, The quality movement, software quality assurance, Reviews, Reliability.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Understand different kinds of risks that a software engineer comes across while developing a project or product(s).
2. Predict project scheduling and tracking of errors
3. Explain various technical reviews in order to get quality software.

Unit IV : Software configuration management and Analysis

Identification of objects in the software configuration, configuration audit, SCM standards. Analysis concepts and principles**:** Requirement analysis, prototyping, Specification Review.

Analysis modeling**:** Data modeling, functional modeling, Behavioral modeling, Data dictionary.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Identify the changes in the software development and controlling them by programming team with SCM standards.
2. Explain requirements gathering and analysis phase of Software Development Lifecycle.
3. Distinguish different types of analysis models.
4. Draw UML diagrams for the scenario given.

Unit V: Design concepts, principles and Software Testing

Effective modular design, design heuristics, Design model, Documentation, Software design**,** Software architecture, Data designing, Architectural styles, Transform mapping, Transaction mapping, Refining architectural design. User interface design.

Software Testing Techniques: White box and black box testing, Testing for specialized environment, architectures and applications.

Software testing Strategies: Unit testing, integrating testing, validation technique, System testing, debugging.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Understand basics of Design concepts of Software Development Lifecycle.
2. Classify various software testing techniques.
3. Use testing techniques in software project development.
4. Develop User interface for any given Web applications.

Text Books:

1. Roger S. Pressman, Software Engineering - A Practitioner's Approach, 6th Edition, MGH, 2005.
2. Ian Sommerville, Software Engineering, 9th Edition, Pearson Publishers,2010

Reference Books:

1. Rajib Mall, Fundamentals of Software Engineering, PHI, 3rd Edition, 2009.
2. Waman S Jawadekar, Software Engineering: A Primer, Tata McGraw-Hill,2008.
3. Pankaj Jalote, Software Engineering, A Precise Approach, WileyIndia,2010.

Course Outcomes:

At the end of the course, student will be able to

1. Classify Software development life cycle models and have knowledge on different phases of software process models.
2. Demonstrate the basics of Measures, Metrics, Indicators and their approaches in project planning, estimation to have a quality product.
3. Recognize different kinds of risks and track the errors while developing a project or product.
4. Explain software quality assurance methods in order to get quality software.
5. Describe requirements gathering, different types of analysis models and controlling changes in software development with SCM standards.
6. Understand basics of Design concepts, various software testing techniques and strategies

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Database Management Systems Lab

(Common to CSE, CSM & CSD)

II B.Tech - I Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA05304	**PCC**	0	0	3	1.5	40	60	100

Objectives

- To understand the fundamentals of SQL and PL/SQL.
- To use DDL, DML statements for design of database Schemas.
- To use features of Query language like aggregate functions, group-by clause.
- To use set operations and set comparison operators for evaluating complex queries.
- To use PL/SQL language constructs to implement triggers, stored procedures, stored functions and cursors.

List of Experiments:

1. Write SQL queries to CREATE TABLES for various databases using DDL commands (i.e. CREATE, ALTER, DROP,TRUNCATE).
2. Write SQL queries to MANIPULATE TABLES for various databases using DML commands (i.e. INSERT, SELECT, UPDATE,DELETE,).
3. Write SQL queries to create VIEWS for various databases (i.e. CREATEVIEW, UPDATE VIEW, ALTER VIEW, and DELETEVIEW).
4. Write SQL queries to perform RELATIONAL SET OPERATIONS (i.e. UNION, UNION ALL, INTERSECT, MINUS, CROSS JOIN, NATURALJOIN).
5. Write SQL queries to perform SPECIAL OPERATIONS (i.e. ISNULL,BETWEEN, LIKE, IN,EXISTS).
6. Write SQL queries to perform JOIN OPERATIONS (i.e. CONDITIONALJOIN, EQUI JOIN, LEFT OUTER JOIN, RIGHT OUTER JOIN, FULL OUTERJOIN).
7. Write SQL queries to perform AGGREGATE OPERATIONS (i.e. SUM,COUNT, AVG, MIN,MAX).
8. Write SQL queries to perform ORACLE BUILT-IN FUNCTIONS (i.e. DATE,TIME).
9. Write SQL queries to perform KEY CONSTRAINTS (i.e. PRIMARYKEY, FOREIGN KEY, UNIQUE NOT NULL, CHECK,DEFAULT).
10. Write a PL/SQL program for calculating the factorial of a given number.
11. Write a PL/SQL program for finding the given number is prime number or not.
12. Write a PL/SQL program for displaying the Fibonacci series up to an integer.
13. Write PL/SQL program to implement Stored Procedure on table.
14. Write PL/SQL program to implement Stored Function on table.
15. Write PL/SQL program to implement Trigger on table.
16. Write PL/SQL program to implement Cursor on table.

Reference Books:

1. "Database System Concepts", 6th Edition by Abraham Silberschatz, Henry F. Korth,S. Sudarshan, McGraw-Hill,2011.
2. Steven Feuerstein. OraclePL/SQL Programming,2014.

Course Outcomes:

At the end of the lab, students will be able to

1. Implement a database schema for a given specifications.
2. Insert, alter and modify the database schema and its instances.
3. Write SQL query for a given requirement.
4. Evaluate equivalent SQL queries for a given requirement.
5. Perform join, aggregate operation and use oracle built in functions for a given query.
6. Develop PL/SQL triggers, stored procedures, stored functions for a database.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Object Oriented Programming Lab

(Common to CSE, CSM & CSD)

II B.Tech - I Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA05305	PCC	0	0	3	1.5	40	60	100

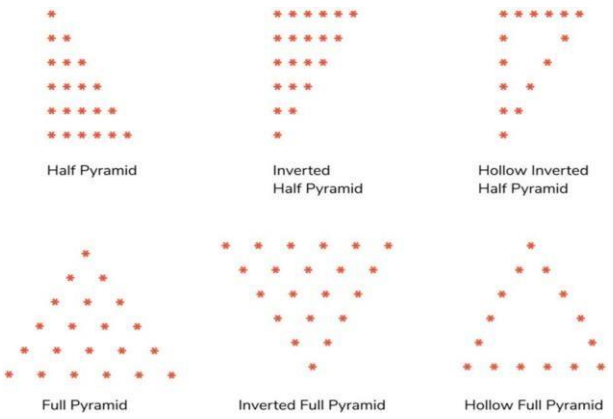
Objectives

- Learn to use object orientation to solve problems and use java language to implement them.
- To experiment with the syntax and semantics of java language and gain experience with java programming.
- Discuss JDK Java environment to create, debug and run simple Java programs
- Demonstrate java compiler and eclipse platform and learn how to use Netbeans IDE to create Java Application.
- Discuss database connectivity with java programming.

List of Experiments:

1) Preparing and practice – Installation of Java software, study of any Integrated development environment, sample programs on operator precedence and associativity, class and package concept, scope concept, control structures, constructors and destructors. Learn to compile, debug and execute java programs.

2) Write Java program(s) that print following pyramid patterns.



3) Write Java program(s) on use of inheritance, preventing inheritance using final, abstract classes.
4) Write Java program(s) on dynamic binding, method overloading and overriding.
5) Write Java program(s) on ways of implementing interface.
6) Write Java program(s) which uses the exception handling features of the language, creates exceptions and handles them properly, uses the predefined exceptions, and create own exceptions
7) Write java program that inputs 5 numbers, each between 10 and 100 inclusive. As each number is read display it only if it's not a duplicate of any number already read.
   Display the complete set of unique values input after the user enters each new value.
8) Write Java program(s) on creating multiple threads, assigning priority to threads, synchronizing threads, suspend and resume threads

9) Write a java program to split a given text file into n parts. Name each part as the name of the original file followed by .part<n> where n is the sequence number of the part file.

10) Write a java program to create a super class called Figure that receives the dimensions of two dimensional objects. It also defines a method called area that computes the area of an object. The program derives two subclasses from Figure. The first is Rectangle and second is Triangle. Each of the sub classes override area() so that it returns the area of a rectangle and triangle respectively.

11) Write a Java program that creates three threads. First thread displays "Good Morning" every one second, the second thread displays "Hello" every two seconds and the third thread displays "Welcome" every three seconds.

12) Design a simple calculator which performs all arithmetic operations. The interface should look like the calculator application of the operating system. Handle the exceptions if any.

13) Write a java program to handle mouse events and keyboard events.

14) Write a java program that allows conduction of object type examination containing multiple choice questions, and true/false questions. At the end of the examination when the user clicks a button the total marks have to be displayed in the form of the message.

15) Write a java program that creates menu which appears similar to the menu of notepad application of the Microsoft windows or any editor of your choice.

16) Write a java program that creates dialog box which is similar to the save dialog box of the Microsoft windows or any word processor of your choice.

17) Write a java program to find and replace pattern in a given file.

18) Use inheritance to create an exception super class called Exception A and exception sub classes Exception B and Exception C, where Exception B inherits from Exception A and Exception C inherits from Exception B. Write a java program to demonstrate that the catch block for type Exception A catches exception of type Exception B and Exception C.

19) Write a Java program which opens a connection to standard port on well-known server, sends the data using socket and prints the returned data.

20) Create an interface for stack with push and pop operations. Implement the stack in two ways: fixed size stack and Dynamic stack (stack size is increased when stack is full).

21) Create multiple threads to access the contents of a stack. Synchronize thread to prevent simultaneous access to push and pop operations.

22) Write java program(s) that use collection framework classes Array List, Linked List, Hash Map, Linked Hash Map, Tree Map, Tree Set, Hash Table, Iterator, ListIterator.

23) Write a Java program to connect with any Database by using JDBC(Java Database Connectivity)specification.

Reference Books:

1. Deitel P.J., and Deitel H.M., "Java: How to Program", PHI, 8th Edition.
2. Radha Krishna P., "Object Oriented Programming through Java", Universities Press,2007.
3. Bruce Eckel, "Thinking in Java", Pearson Education,2006.

Course Outcomes:

At the end of the lab, students will be able to

1. Implement data types, variables and control structures to solve problems.
2. Use object-oriented concepts to solve problems including generating series primes, searching a pattern in a file.
3. Write and execute programs using inheritance and interfaces.
4. Develop programs using threads and Exception handling.
5. Implement programs using collection framework.
6. Design layouts using AWT and swing concepts.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Software Engineering Lab

(Computer Science and Engineering)

II B.Tech - I Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
R204GA05306	**PCC**	**L**	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
		0	0	3	1.5	40	60	100

Objectives:

- To understand how to develop a software project plan and SRS document using tools.
- To design a software using UML tool.
- To test the software, website, security and system using tools.
- To analyze the quality and Object- Oriented metrics to ensure the quality of software.

List of Experiments:

Do the following 7 exercises for any two projects given in the list of sample projects or any other projects:

1) Development of problem statement.
2) Preparation of Software Requirement Specification Document, Design Documents and Testing Phase related documents.
3) Preparation of Software Configuration Management and Risk Management related documents.
4) Study and usage of any Design phase CASE tool
5) Performing the Design by using any Design phase CASE tools.
6) Develop test cases for unit testing and integration testing
7) Develop test cases for various white box and black box testing techniques.

Reference Books:

1. Booch, Jackobson and Rambaugh, "UML Guide", Pearson Edu,1999.
2. Chapman and Hall,Software Metrics:A Rigorous Approach, IEEE Standards for SRS Documents, IEEE Std. 840. 4. Fenton NE,1991.

Course Outcomes:

At the end of the lab, student will be able to

1. Prepare software project plan.
2. Prepare Software Requirement Specification document.
3. Prepare design document and compute effort estimates for a software project.
4. Design UML diagram for a case study.
5. Design and Develop test cases for a software.
6. Perform website, security and system testing.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Environmental Science

(Common to all Branches)

II B.Tech - I Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA5MC01	**NCMC**	2	0	0	0	40	-	40

Objectives

- To understand the importance of protecting natural resources for future generations.
- To understand the importance of protecting of ecosystem and biodiversity for future generations.
- To understand the importance pollution causes due to the day-to-day activities of human life to save earth from the inventions by the engineers.
- To Discuss the Water conservation, rain water harvesting, global issues and possible solutions.
- To Discuss the Population growth, variation among nations and the Role of IT in environment and human health.

Unit I - Multidisciplinary Nature of Environmental Studies

Definition, Scope and Importance – Need for Public Awareness.

NATURAL RESOURCES: Renewable and non-renewable resources –Forest resources – Use and over – exploitation, deforestation– Mineral resources: Use and exploitation, environmental effects of extracting and using mineral resources, – Food resources: World food problems, changes caused by agriculture and overgrazing, effects of modern agriculture, fertilizer-pesticide problems, – Energy resources.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Explain the scope and importance of Environment.
2. Discuss the forest and mineral resources
3. Explain the Energy resources.

Unit II - Ecosystems Biodiversity and Its Conservation

Concept of an ecosystem. – Structure and function of an ecosystem – Producers, consumers and decomposes – Energy flow in the ecosystem – Ecological succession – Food chains, food webs and ecological pyramids.

Definition of biodiversity- genetic, species and ecosystem diversity – Value of biodiversity, consumptive use, Productive use, social, ethical, aesthetic and option values – Hot-sports of biodiversity – Threats to biodiversity: habitat loss, poaching of wildlife, man-wildlife conflicts-Conservation of biodiversity: In-situ and Ex-situ conservation of biodiversity.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Discuss the concept of ecosystem and biodiversity
2. Explain the values, threats and conservation of biodiversity.
3. Explain the Structure and function of an ecosystem.

Unit III - Environmental Pollution

Definition, Cause, effects and control measures of: Air pollution, water pollution, Mariene pollution, noise pollution, Nuclear pollution.

SOLID WASTE MANAGEMENT: Causes, effects and control measures of urban and industrial wastes – Role of an individual in prevention of pollution – Pollution case studies.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Explain the Cause, effects of environmental pollution.
2. Discuss the Causes and control measures urban and industrial wastes.

Unit IV - Social Issues and the Environment

From Unsustainable to Sustainable development – Water conservation, rain water harvesting– Resettlement and rehabilitation of people; its problems and concerns. Issues and possible solutions – Climate change, global warming, acid rain, ozone layer depletion, nuclear accidents. Wasteland reclamation, Environment Protection Act, Forest Conservation Act–Public awareness.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Discuss the Water conservation, rain water harvesting.
2. Explain the global issues and possible solutions.
3. Explain the Wildlife and forest conservation act.

Unit V - Human Population and the Environment

Population growth, variation among nations. Population explosion – Family Welfare Programmed. Environment and human health – Human Rights – Value Education – Women and Child Welfare – Role of information Technology in Environment and human health.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Discuss the Population growth, variation among nations.
2. Explain the Role of IT in environment and human health.

Text Books:

1. ErachBharucha, "Text Book of Environmental Studies for Undergraduate Courses," Universities Press Pvt Ltd, Hyderabad, 2nd Edition 2013.
2. Kaushik, "Environmental Studies," New Age Publishers.

Reference Books:

1. Rajagopalan, "Environmental Studies," Oxford Publishers.
2. Sharma J. P., "Comprehensive Environmental studies," Laxmi publications.
3. Gilbert M. Masters, Wendell. P., "Introduction to Environmental engineering and science" Ela - Printice hall of India Private limited.

Course Outcomes:

At the end of the course student will be able to

1. Explain the importance of environment, natural resources, biodiversity, environmental pollution, global issues, effects of population and environment.
2. Explain the scope and importance of environment, uses& exploitation of forest, mineral, food resources, renewable and non-renewable energy resources.
3. Discuss the structure and functions of ecosystem, Ecological pyramids, Food chains, Ecological succession, values, threats, Conservation and Hot-spots of biodiversity.

4. Explain the causes, effects and control methods of air, water, marine, noise and nuclear pollution, causes, effects and control methods of solid wastage.
5. Discuss the resettlement and rehabilitation issues, waste land reclamation, water conversation, causes, effects and control methods of acid rain, ozone layer depletion, global warming and environmental protection act.
6. Explain the factors effecting population, women and child welfare programmers, human rights, value.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Discrete Mathematics

(Common to CSE, CSM & CSD)

II B.Tech - II Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA05401	**BSC**	3	0	0	3	40	60	100

Objectives

- This course will introduce and illustrate in the elementary discrete mathematics for computer science and engineering students.
- To equip the students with standard concepts like formal logic notation, methods of proof, induction, sets, relations, graph theory, permutations and combinations, counting principles.

Unit I – Mathematical Logic

Propositional Calculus: Statements and Notations, Connectives, Well Formed Formulas, Truth Tables, Tautologies, Equivalence of Formulas, Duality Law, Tautological Implications, Normal Forms, Theory of Inference for Statement Calculus, Consistency of Premises, Indirect Method of Proof.

Predicate Calculus: Predicative Logic, Statement Functions, Variables and Quantifiers, Free and Bound Variables, Inference Theory for Predicate Calculus.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Simplify and evaluate basic logic statements using truth tables.
2. Express a logic sentence in terms of predicates, quantifiers and logical connectives.
3. Apply rules of inference and methods of proof including direct and indirect proof forms.
4. Understand the inference theory for predicate calculus.

Unit II – Set Theory

Introduction, Operations on Binary Sets, Principle of Inclusion and Exclusion.

Relations: Properties of Binary Relations, Relation Matrix and Digraph, Operations on Relations, Partition and Covering, Transitive Closure, Equivalence, Compatibility and Partial Ordering Relations, Hasse Diagrams.

Functions: Bijective Functions, Composition of Functions, Inverse Functions, Permutation Functions, Recursive Functions, Lattice and its Properties.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Understand the operations of sets and use Venn diagrams to solve applied problems.
2. Determine the domain and range of a function and apply the properties of functions to application problems.
3. Identify the types of functions, finding the inverse of function and perform the composition of functions.
4. Understand about lattice and its properties.

Unit III - Algebraic Structures and Number Theory

Algebraic Structures: Algebraic Systems, Examples, General Properties, Semi Groups and Monoids, Homomorphism of Semi Groups and Monoids, Group, Subgroup, Abelian Group, Homomorphism, Isomorphism.

Number Theory: Properties of Integers, Division Theorem, The Greatest Common Divisor, Euclidean Algorithm, Least Common Multiple, Testing for Prime Numbers, The Fundamental Theorem of Arithmetic, Modular Arithmetic (Fermat's Theorem and Euler's Theorem).

Learning Outcomes:

At the end of this unit, the student will be able to

1. Understand the various algebraic structures and their properties.
2. Use elementary number theory including divisibility properties, prime numbers, GCD and perform modulo arithmetic.
3. Apply algorithm such as Euclidean and theorems such as Fermat's and Euler's for solving the problems.

Unit IV – Combinatorics

Basic of Counting, Permutations, Permutations with Repetitions, Circular Permutations, Restricted Permutations, Combinations, Restricted Combinations, Generating Functions of Permutations and Combinations, Binomial and Multinomial Coefficients, Binomial and Multinomial Theorems. The Principles of Inclusion, Exclusion, Pigeonhole Principle and its Application.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Know the fundamentals of counting and understanding the difference between permutation and combination.
2. Solve the counting problems by applying product and sum rules, permutations and combinations.
3. Understand the pigeonhole principle and its applications.
4. Apply Binomial Theorem for solving problems.

Unit V – Graph Theory

Basic Concepts of Graphs, Sub graphs, Matrix Representation of Graphs: Adjacency Matrices, Incidence Matrices, Isomorphic Graphs, Paths and Circuits, Eulerian and Hamiltonian Graphs, Multigraphs, Planar Graphs, Euler's Formula, Graph Coloring and Covering, Chromatic Number, Spanning Trees, Algorithms for Spanning Trees.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Determine if a graph is simple / multi-graph, directed / undirected,cyclic/acyclic.
2. Represent a graph using adjacency list and adjacency matrix and apply graph theory to problems in computer networks.
3. Determine if a graph has Euler or Hamilton path /circuit.
4. Understand about spanning tree and apply the algorithms for spanning trees in solving the problems.

Text Books:

1. Tremblay J. P.,and Manohar P., Discrete Mathematical Structures with Applications to Computer Science, Tata McGraw Hill,2015.
2. Liu C. L. and Mohapatra D.P., Elements of Discrete Mathematics-A Computer Oriented Approach, Tata McGraw Hill, 3rd Edition,2008.

Reference Books:

1. Erwin Kreyszig, Advanced Engineering Mathematics, Wiley India,10th Edition, 2014.
2. Ramana B.V., Higher Engineering Mathematics, Mc Graw Hill publishers, Sixth Reprint, 2008.
3. Alan Jeffrey, Advanced Engineering Mathematics, Elsevier, 1st Edition, 2010.

Course Outcomes:

At the end of the course, student will be able to

1. Illustrate discrete mathematic components like logic, sets, structures, numbers and combinatorics.
2. Evaluate and simplify propositional and predicate calculus using inference theory.
3. Perform the operations on Sets, Relations and functions and their properties.
4. Identify algebraic systems and use general properties on number theory.
5. Implement the concepts of permutations & combinations for solving the counting problems.
6. Compute operations on Graphs and spanning trees.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Formal Languages and Automata Theory

(Common to CSE, CSM & CSD)

II B.Tech - II Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA05402	PCC	3	0	0	3	40	60	100

Objectives

- To understand the properties of Formal Languages, Deterministic and Non - Deterministic Finite Automata.
- To construct finite automata for regular expressions.
- To Illustrate the Context free languages and grammars, Normalizing CFG.
- To differentiate the deterministic and non-deterministic PDA.
- To apply the properties of Turing machines to solve the real time problems.

Unit I – Introduction

Finite Automata preliminaries: Strings, Alphabet, Language Operations, Finite State Machine: definitions, Applications. Finite Automation Model, Acceptance of strings and languages, Non-deterministic Finite Automation, NFA with ε moves. Equivalence between NFA and DFA, conversion of NFA into DFA, Equivalence between two FSM's, Minimization of FSM, Moore and Mealy machines.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Understand the basics of formal languages and its operations.
2. Illustrate conversions and equivalences of finite automata.
3. Know the importance of finite automata in compiler design.

Unit II – Regular Sets

Regular sets, Regular Expressions, Identity Rules, Finite Automata and Regular Expressions, Pumping lemma of regular sets, Closure Properties of Regular Sets (proofs not required), Chomsky hierarchy of Languages.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Understand the relationship between Finite automata and Regular languages.
2. Know the pumping lemma of regular sets.
3. Use pumping lemma for show that a language is not regular.

Unit III - Context Free Grammars and Languages

Context Free Grammar, Regular Grammar, Right linear grammar and Left linear grammar, Equivalence between regular linear grammar and FA, inter-conversion between RE and RG. Definition of CFG, derivation trees, and sentential forms, Right Most and Left Most derivation of Strings, Ambiguity in Context Free Grammars. Simplification of Context Free Grammars, Chomsky Normal Form, Greibach Normal Form, Pumping Lemma for Context Free Languages. Decision properties of CFL (proofs omitted).

Learning Outcomes:

At the end of this unit, the student will be able to

1. Understand Regular Grammars and Context Free Grammars.
2. Apply minimization of Context Free Grammars.
3. Use pumping lemma to prove that the given language is CFL or not.

Unit IV - Push down Automata

Push Down Automata: Definition of the Pushdown Automaton, A Graphical Notation for PDA's, Instantaneous Descriptions of a PDA, The Languages of a PDA, Acceptance by Final State, Acceptance by Empty Stack, Equivalence of PDA's and CFG's, Properties of Context Free Languages.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Understand the acceptance of Push Down Automata.
2. Apply equivalence of CFL and PDA.

Unit V - Turing Machine, Decidability and REL

Turing Machine: Turing Machine Model, Representation of TM, Design of TM, Types of Turing machines (proofs not required).

Decidability and Recursively Enumerable Languages: Decidability of problems, Universal Turing Machine, Undecidability of Post Correspondence Problem, Rice's theorem.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Understand the basics of Turing Machine.
2. Illustrate concepts of Computability Theory.
3. Generalize Turing Machines into universal TMs.

Text Books:

1. Mishra and Chandrashekaran, "Theory of Computer Science and Automata languages and computation", PHI,3rd edition, 2011.
2. John E.Hopcroft, Rajeev Motwani and Jeffrey D. Ullman, "Introduction to Automata Theory Languages and Computation", Pearson, 3rd Edition,2006.

Reference Books:

1. Daniel I.A. Cohen, "Introduction to Computer Theory", John Wiley,2013.
2. Sipser, "Introduction to Theory of Computation" Thomson,2nd edition, 2018.
3. Shyamalendu Kandar, "Introduction to Automata Theory, Formal Languages and Computation",Pearson,2013.

Course Outcomes:

At the end of the course, student will be able to

1. Know the types of formal machines and languages.
2. Design finite and non-finite state machines for acceptance of strings.
3. Implement Regular Expressions in real time applications.
4. Design Context Free Grammars for formal languages.
5. Develop push down automata for a given language
6. Design Turing Machines and distinguish between Decidability and Undecidability.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Python Programming

(Common to CSE, CSM, CSD & ECE)

II B.Tech - II Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA05403	**PCC**	3	0	0	3	40	60	100

Objectives

- To understand the basics of Scripting Language.
- To get exposure on problems solving approaches of computer science.
- To use various packages in solving problems.
- To develop the skill of designing Graphical user Interfaces in Python.
- To develop the ability to write database applications in Python.

Unit I – Basics of Python

Introduction: History of Python, Need of Python Programming, Applications Basics of Python Programming using the REPL(Shell), Running Python Scripts, Variables, Assignment, Keywords, Input-Output, Indentation.

Types, Operators and Expressions: Types - Integers, Strings, Booleans; Operators- Arithmetic Operators, Comparison (Relational) Operators, Assignment Operators, Logical Operators, Bitwise Operators, Membership Operators, Identity Operators, Expressions and order of evaluations Control Flow- if, if-elif-else, for, while, break, continue, pass.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Understand the programming in Python using the REPL(Shell).
2. Categorize the types, operators and expressions in python programming.

Unit II – Data Structures & Functions

Data Structures - Lists - Operations, Slicing, Methods; Tuples, Sets, Dictionaries, Sequences. Comprehensions.

Functions - Defining Functions, Calling Functions, Passing Arguments, Keyword Arguments, Default Arguments, Variable-length arguments, Anonymous Functions, Fruitful Functions (Function Returning Values), Scope of the Variables in a Function - Global and Local Variables.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Classify the data into data structures.
2. Use the functions in solving python-based solutions.

Unit III – Modules, Python Packages & Brief Tour of the Standard Library

Modules: Creating modules, import statement, from import statement, name spacing.

Python packages: Introduction to PIP, Installing Packages via PIP, Using Python Packages Error and Exceptions: Difference between an error and Exception, Handling Exception, try except block, Raising Exceptions, User Defined Exceptions

Brief Tour of the Standard Library - Operating System Interface - String Pattern Matching, Mathematics, Internet Access, Dates and Times, Data Compression, Multithreading.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Understand the importing of modules and their importance.
2. Understand the PIP package in installing the python packages.
3. Use the error handling techniques in python.

Unit IV – Objects and their Use, Object Oriented Programming

Objects and Their Use: Software objects, Turtle graphics - Creating a turtle graphics window, The default turtle, Fundamental turtle attributes and behavior, Additional turtle attributes, Creating multiple turtles.

Object Oriented Programming**:** Encapsulation, Inheritance, and Polymorphism.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Use the software objects in Turtle graphics.
2. Understand the concepts of Object Oriented Programming in python.

Unit V – GUI Programming, Testing

GUI Programming - Tkinter Overview, Tkinter pragmatics, Documentation, Extensions, structure; Tkinter coding alternatives, adding buttons and callbacks-lambda, bound method, callable class object, Binding events; adding multiple widgets, Reusable GUI Components with classes, Dialog,

Entry, check buttons and Radio buttons, Scales, Menus.

Testing: Why testing is required? Basic concepts of testing, Unit testing in Python, Writing Test cases, Running Tests.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Understand the basics of GUI programming in python.
2. Use the Tkinter package in designing the Graphical User Interface.
3. Represent and run the testcases.

Text Books:

1. Mark Lutz, Learning Python, O'Reilly Publications, 5th edition,2013.
2. Charles Dierbach, Introduction to Computer Science using Python: A Computational Problem-Solving Focus,Wiley India Edition,2016.

Reference Books:

1. Vamsi Kurama, Python Programming: A Modern Approach, Pearson,2017.
2. Allen Downey, Think Python, Green Tea Press,2012.
3. Kenneth Lambert and Juneja B.L., Fundamentals of Python Cengage Learning,3rd Edition,2012.

Course Outcomes:

At the end of the course, student will be able to

1. Describe fundamentals of Python programming and its applications.
2. Implement Python programs using data types, Operators and Control statements.
3. Operate Python data structures for accessing data using functions.
4. Carry out modular programming using functions and packages and standard libraries.
5. Implement graphical methods and OOP concepts in Python Programming.
6. Implement Interfaces and testing.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Design and Analysis of Algorithms

(Computer Science and Engineering)

II B.Tech - II Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA05404	PCC	3	0	0	3	40	60	100

Objectives

- To know the importance of the complexity of a given algorithm.
- To study various algorithm design techniques.
- To utilize data structures and/or algorithmic design techniques in solving new problems.
- To illustrate clever and efficient ways to solve a given problem.
- To know and understand basic computability concepts and the complexity classes P,NP and NP-Complete.
- To study some techniques for solving hard problems.

Unit I – Algorithms and Searching Techniques

Algorithm: Definition, specification, Performance analysis, growth of functions. Elementary Data Structures: Stacks, queues, trees, heaps, sets and disjoint sets, union, graphs, hashing.

Basic Traversal and Search Techniques for binary trees and Graphs, Connected components and Spanning trees, Bi-connected components and DFS.

Learning Outcomes:

At the end of unit**,** students will be able to

1. Understand the elementary data structures.
2. Understand the importance of analyzing the complexity of an algorithm.
3. Apply traversal and search techniques for binary trees and various graphs.

Unit II – Divide and Conquer, Greedy Method

Divide and Conquer: General method, Binary Search, Finding the maximum and minimum, Merge sort, Quick Sort, Selection sort, Stassen's matrix multiplication.

Greedy Method: General method, Knapsack problem, Job Scheduling with Deadlines, Minimum cost Spanning Trees, Optimal storage on tapes, Single-Source shortest paths.

Learning Outcomes:

At the end of unit**,** students will be able to

1. Apply the divide and conquer technique to solve the new problems.
2. Apply the greedy method to solve the optimization problems.
3. Analyze the complexity of a given algorithm.

Unit III – Dynamic Programming and Backtracking

Dynamic programming: General Method, Multistage graphs, All-pairs shortest paths, Optimal binary search trees, 0/1 knapsack, The Traveling sales person problem.

Back Tracking: General Method, 8 – queens problem, Sum of subsets problem, Graph coloring and Hamiltonian cycles, Knapsack Problem.

Learning Outcomes:

At the end of unit**,** students will be able to

1. Apply dynamic programming technique to solve the optimization problems.
2. Apply Backtracking technique for solving constraint satisfaction problems.

Unit IV –Branch and Bound, Lower Bound Theory

Branch and Bound: The method, Travelling salesperson, 0/1 Knapsack problem, Efficiency considerations.

Lower Bound Theory: Comparison trees, Lower bounds through reductions – Multiplying triangular matrices, Inverting a lower triangular matrix, Computing the transitive closure.

Learning Outcomes:

At the end of unit**,** students will be able to

1. Understand the importance of lower bound theory concept.
2. ApplyBranchandBounddesigntechniquetosolvecombinatorialoptimizationproblems.

Unit V : NP Hard and NP Complete Problems

NP – Hard and NP – Complete Problems: NP Hardness, NP Completeness, Consequences of being in P, Cook's Theorem, Reduction Source Problems, Reductions: Reductions for some known problems.

Learning Outcomes:

At the end of unit**,** students will be able to

1. Classify the problems into NP hard and NP complete.
2. Understand the NP complete problem and cook's theorem.

Text Books:

1. Ellis Horowitz, Sartaj Sahani S., and Rajasekhran,"Fundamentals of Computer Algorithms", UniversityPress, 2nd edition, 2012.
2. Parag Himanshu Dave, Himanshu BhalchandraDave, "Design and Analysis of Algorithms", Pearson Education, 2nd Edition, 2009.

Reference Books:

1. Cormen T.H., Leiserson C.E., Rivest R.L.,and Stein C., "Introduction to Algorithms", PHI Pvt. Ltd./ Pearson Education,2nd Edition, 2001.
2. Lee R.C.T., Tseng S.S., Chang R.C., and Tsai T., "Introduction to Design and Analysis of Algorithms A strategic approach",Mc GrawHill,2021.
3. AllenWeiss, "DatastructuresandAlgorithmAnalysisinC++",Pearson education,2nd Edition, 2005.

Course Outcomes:

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

1. Find the solution for a given problem using different general methods.
2. Implement Divide-and-Conquer design approach to solve the problems like binary search, finding maximum and minimum.
3. Implement Greedy method and Dynamic Programming Techniques to solve the optimization problems.
4. Use Backtracking technique for solving constraint satisfaction problems.
5. Use Branch and Bound design technique to solve combinatorial optimization problems.
6. Implement reductions for known problems in NP Hard and NP Complete.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Digital Logic Design

(Computer Science and Engineering)

II B.Tech - II Semester **SRIT R20**

Course Code	Category	Hours/Week			Credits	Maximum Marks		
		L	T	P	C	CIA	SEE	Total
R204GA04407	ESC	3	0	0	3	40	60	100

Objectives

- Acquire the skills to manipulate and examine Boolean algebraic expressions, logical operations, Boolean functions and their simplifications.
- Understand the fundamental principles of digital design.
- Acquaint with classical hardware design for both combinational and sequential logic circuits
- Understand the basic software tools for the design and implementation of digital circuits and systems.
- Reinforce theory and techniques taught in the classroom through experiments and projects in the laboratory.

UNIT I- Number Systems and Switching Functions

Binary Systems: Digital Systems, Binary Numbers, Number Base Conversions, Octal and Hexadecimal Numbers, Compliments, Signed Binary Numbers, Binary Codes, Binary Storage And Registers, Binary Logic.

Boolean Algebra and Logic Gates: Basic Definitions, Axiomatic Definition of Boolean algebra, Basic Theorems and properties of Boolean Algebra, Boolean Functions, Canonical and Standard Forms, Other Logic Operations, Digital Logic Gates, Integrated Circuits.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Define Number systems and perform Number base conversions.
2. Carryout the Conversion from one number base system to another number base system.
3. Use basic Theorems & Properties of Boolean algebra for simplifying Boolean expressions.
4. Illustrate the logic gates and its logic operations.

UNIT II- Gate- Level Minimization

The Map Method, Four Variable Map, Five-Variable Map, Product of Sums Simplification, Don't-Care Conditions, NAND and NOR Implementation, Other Two Level Implementations, Ex-OR Function, other Minimization Methods.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Carryout boolean functions by using n-variable K-map(n=1, 2,3,4,5).
2. Construct NAND-NOR Implementation and other two level Implementation.
3. Describe the Simplification of boolean expressions using standard methods.
4. Use tabular method for simplification of Boolean functions.

UNIT III - COMBINATIONAL LOGIC

Combinational Circuits, Analysis Procedure, Design Procedure, Binary Adder- Subtractor, Decimal Adder, Binary Multiplier, Magnitude Comparator, Decoders, Encoders, Multiplexers.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Draw the combinational logic circuits for the given specifications
2. Draw the sequential logic circuits for the given specifications.

UNIT IV - SYNCHRONOUS SEQUENTIAL LOGIC

Sequential Circuits, Latches, Flips-Flops, Analysis of Clocked sequential circuits, State Reduction & Assignment, Design procedure, Registers, Shift Registers, Ripple Counters, Synchronous counters, other counters.

Learning Outcomes:

At the end of this unit, the student will be able to

1.Draw the sequential logic circuits for the given specifications

2. Describe asynchronous sequential circuits.

3. Analyze state equation for Mealy and Moore finite state machines.

4. Solve sequential logic circuits with the acquired knowledge of flip-flops.

UNIT V - MEMORY AND PROGRAMMABLE LOGIC:

Random access memory, memory decoding, Error Detection and Correction, Read-only Memory, Programmable Logic Array, Programmable Array Logic.

Digital Logic Circuits: RTL and DTL Circuits, Transistor-Transistor Logic (TTL), Emitter- Coupled Logic (ECL), MOS, CMOS Logic, Comparisons of Logic Families.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Classify semiconductor memories.
2. Carryout the design of PLDs for given Boolean functions.
3. Analyze digital logic circuits.

Text Books:

1. Switching and Finite Automata Theory- ZviKohavi & NirajK. Jha, 3rdEdition, Cambridge, 2010.
2. Digital Design-Morris Mano, PHI, 3rdEdition,2018.

Reference Books:

1. Digital Logic & State Machine Design, David J. Comer, Oxford University Press 3rd Reprinted Indian Edition,2012.
2. Digital Logic Design, R.D. Sudhakar Samuel, Elsevier,2013.
3. Fundamentals of Logic Design, 5/e, Roth, Cengage,2020.

Course Outcomes:

At the end of the course, student will be able to

1. Describe the basic concepts of binary numbers, Boolean functions, logic gates, combinational & sequential logic circuits, Programmable devices and digital logic circuits.
2. Use basic theorems and properties of Boolean algebra for simplifying Boolean expressions.
3. Carryout the Boolean functions by using n variable K-map.
4. Draw the combinational & sequential logic circuits for the given specifications or constraints.
5. Analyze sequential logic circuits with the acquired knowledge of flip-flops.
6. Draw the Programmable devices and digital circuits by using given design procedures.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Python Programming Lab

(Common to CSE, CSM & CSD)

II B.Tech - II Semester							SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**	
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA05405	**PCC**	0	0	3	1.5	40	60	100

Objectives

- To understand the ease of programming using python.
- To develop the graphics using graphics package.
- To develop interface of an application.
- To apply object-oriented concepts in programming.
- To understand a range of Object-Oriented Programming, as well as in-depth data and information processing techniques.

List of Experiments:

1. a. Write a python script to display a simple message.
   b Write a python script to perform basic arithmetic operations on two values which are accepted from the user.
2. a. Write a python script to calculate the factorial of a given number.
   b. Write a python script to calculate sum of individual digits of a given number.
   c. Write a python script to display the prime number series up to the given N Value.
3. a. Write a python script to find the largest number among three numbers and display them in ascending order using if-else construct.
   b. Write a python script to create a simple text file, write the contents into the created file and display the same on to the console screen.
4. Write a python script to remove all the occurrences of a given character from a text file, copy the resultant text into another text file. Find the total occurrences of the eliminated characters and display the count along with the contents of the text file on to the console.
5. a. Write a python script to display Fibonacci sequence of numbers using while loop, for loop and do-while loop constructs.
   b. Write a python script to demonstrate string methods.
6. a. Write a python script to create a list and add n number of user-defined values to the list and display the same on to the console screen.
   b. Write a python script to perform the following operations on Lists:
      (i) Matrix Addition.
      (ii) Matrix Multiplication.
7. a. Write a python script to search a key element in the given list of elements.
   b. Write a python script to arrange the given list of elements in ascending or descending order.
8. a. Write a python script to find GCD of two numbers using recursive and non recursive functions.
   b. Write a python script to convert the following using functions:
      (i) Fahrenheit to Celsius temperature.
      (ii) Celsius to Fahrenheit temperature.
9. a. Write a python script to draw a square using set position method in absolute positioning.

b. Write a python script to draw a triangle using left, right and Forward methods in relative positioning.

c. Write a python script using penup and pendown methods to draw "W" character using turtle graphics.

d. Write a python script to create your own polygon shape and create an interesting design with it.

10. a. Write a GUI Script for creating text label in a window.

b. Write a Python Script to create a command button. When the button is clicked the event should be handled and the message on the window should change from "Hello" to "Good Bye".

11. a. Write a python script to demonstrate the Exception Handling.

b. Write a Python script to demonstrate the Mouse and Key Event handling.

c. Write a python script to demonstrate menu driven applications.

By forming a group of 3 to 4 members develop a mini project.

Reference Books:

1. Charles Dierbach, Introduction to Computer Science using Python: A Computational Problem-Solving Focus, Wiley India Edition,2016.
2. Mark Lutz, Programming Python, O'Reilly Publications, 4th Edition,2011.
3. Allen B. Downey, Think Python: How to Think Like a Computer Scientist, O'Reilly Publications, 2nd edition, 2015.

Course Outcomes:

At the end of the lab, students will be able to:

1. Write a script based programs using python.
2. Write python programs using concepts of functions to improve efficiency of programming.
3. Design graphics using related packages.
4. Implement object oriented concepts in programming.
5. Design user interface using GUI packages for applications.
6. Write programs to handle exceptions a runtime.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Design and Analysis of Algorithms Lab

(Computer Science and Engineering)

II B.Tech - II Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA05406	**PCC**	0	0	3	1.5	40	60	100

Objectives

- Learn how to analyze a problem and design the solution for the problem.
- Design and implement efficient algorithms for a specified application.
- Strengthen the ability to identify and apply the suitable algorithm for the given real world problem.
- To write programs in java to solve problems using divide and conquer strategy.
- To write programs in java to solve problems using backtracking strategy.

List of Experiments:

1. a. Write a Java program to implement the Stack using arrays. Write push(), pop(), and display() methods to demonstrate its working.
   b. Write a Java program to implement the Queue using arrays. Write insert(), delete(), and display() methods to demonstrate its working.
2. a. Write a Java program to print all the nodes reachable from a given starting node in a digraph using BFS method.
   b.Write a Java program to check whether a given graph is connected or not using DFS method.
3. Write a Java program to perform various tree traversal algorithms for a given tree.
4. Sort a given set of n integer elements using Quick Sort method and compute its time complexity. Run the program for varied values of n > 5000 and record the time taken to sort. Plot a graph of the time taken versus n on graph sheet. The elements can be read from a file or can be generated using the random number generator. Demonstrate using Java how the divide -and- conquer method works along with its time complexity analysis: worst case, average case and best case.
5. Sort a given set of n integer elements using Merge Sort method and compute its time complexity. Run the program for varied values of n > 5000, and record the time taken to sort. Plot a graph of the time taken versus n on graph sheet. The elements can be read from a file or can be generated using the random number generator. Demonstrate using Java how the divide-and- conquer method works along with its time complexity analysis: worst case, average case and best case.
6. Implement in Java, the 0/1 Knapsack problem using (a) Greedy method (b) Dynamic Programming method.
7. From a given vertex in a weighted connected graph, find shortest paths to other vertices using Dijkstra's algorithm. Write the program in Java.
8. Find Minimum Cost Spanning Tree of a given undirected graph using (a) Kruskal's algorithm (b) Prim's algorithm. Implement the program in Java language.
9. Write Java programs to (a) Implement All-Pairs Shortest Paths problem using Floyd's algorithm. (b) Implement Travelling Sales Person problem using Dynamic programming.
10. Design and implement in Java to find a subset of a given set S = {Sl, S2,.....,Sn} of n positive integers whose SUM is equal to a given positive integer d. For example, if S ={1, 2,5,6, 8} and d= 9, there are two solutions {1,2,6} and {1,8}. Display a suitable message, if the given problem instance doesn't have a solution.
11. Write Java programs to (a) Implement the presence of Hamiltonian Cycle in an

undirected Graph G of n vertices (b) Implement N Queen's problem using Back Tracking.

Reference Books:

1. Ellis Horowitz, Sartaj Sahani S., and Rajasekhran, "Fundamentals of Computer Algorithms", University Press, 2nd edition, 2012.
2. Deitel P.J., and Deitel H.M., "Java: How to Program", PHI.
3. Bruce Eckel, "Thinking in Java", Pearson Education.

Course Outcomes:

At the end of the lab, students will be able to:

1. Implement stack and queues using arrays.
2. Implement traversals and searching on trees and graphs.
3. Implement Divide-and-Conquer design approach to sort the given elements.
4. Develop Greedy method and Dynamic Programming Techniques to solve the optimization problems.
5. Implement programs on minimum cost spanning trees.
6. Use Back tracking technique for solving constraint satisfaction problems.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Linux Programming Lab

(Common to CSE, CSM & CSD)

II B.Tech - II Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA05407	**PCC**	0	0	3	1.5	40	60	100

Objectives

- To write Linux shell script programs.
- To implement Linux utilities using supported programming language.
- To implement process management.
- To implement Inter Process Communication (IPC).

List of Experiments:

1. Write a shell script that accept a file name, starting and ending line numbers as arguments and display all the lines between given line numbers.
2. Write a shell script that delete all lines containing a specified word.
3. Write a shell script that displays a list of all the files in the current directory.
4. Write a shell script that receives any number of file names as arguments checks if every argument supplied is a file or a directory and reports accordingly. Whenever the argument is a file or directory.
5. Write a shell script that accept a list of file names as arguments count and report the occurrence of each word.
6. Write a shell script to find the factorial of given integer.
7. Write a shell script that list the all files in a directory.
8. Write a awk script to find the number of characters, words and lines in a file?
9. Write a C program that demonstrates command line arguments and displays linux operating system environment.
10. Write a C Program that makes a copy of a file using standard I/O and system calls?
11. Implement in C the following Unix commands using system calls

    A) cat B) mv
12. Write a C program to emulate the Unix ls-l command?
13. Write a C program to list for every file in a directory, its inode number and file name?
14. Write a C Program that demonstrates redirection of standard output to a file.
15. Write a C program that creates child process and displays “parent” in parent process and “child” in child process.
16. Write a C program to create a Zombie process?
17. Write a C program that illustrates how an orphan is created.
18. Write a program that illustrates how to execute two commands concurrently with a command pipe?
19. Write a C programs that illustrate communication between two unrelated processes using named pipe.
20. Write a C program to create a message queue with read and write permissions to write 3 messages to it with different priority numbers.?
21. Write a C program that receives the messages (From the above message queue as specified in (19) and display them?
22. Write a C program that implements the concept of shared memory
23. Write a C program that illustrates suspending and resuming processes using signals.
24. Write client and server programs (using c) for interaction between server and client processes using Unix Domain sockets.

25. Write a client and server programs (using c) for interaction between server and client processes using Internet Domain sockets?

Reference Books:

1. W. Richard. Stevens, Advanced Programming in the UNIX Environment, 3rd edition, Pearson Education, New Delhi, India 2005.
2. Behrouz A. Forouzan, Richard Gilberg F., Unix and shell Programming, Thomson,2002.
3. Robert Love, Linux System Programming, O'Reilly, SPD,2013.

Course Outcomes:

At the end of the lab, students will be able to

1. Demonstrate the basic knowledge of Linux commands and file handling utilities using Linux shell environment and shell scripting.
2. Evaluate the concept of shell scripting programs by using an AWK and SED commands.
3. Illustrate the directory related shell commands and system calls
4. Analyze the process management and process hierarchy.
5. Implement Inter Process Communication (IPC) mechanism.
6. Demonstrate the concept client-server programming using sockets.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Indian Constitution

(Common to all Branches)

II B.Tech - II Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA5MC02	**NCMC**	2	0	0	0	40	-	40

Objectives

- Know History of making of the Indian Constitution.
- Know the Fundamental Rights & Duties.
- Understand the organs of Governance.
- Understand Local Administration.
- Understand the Role & Functions of Election Commission.

UNIT I - History & Philosophy of the Indian Constitution

History of making of the Indian Constitution: History, Drafting Committee (Composition & Working) Philosophy of the Indian Constitution: Preamble, Salient Features.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Know the history of Indian Constitution.
2. Understand the Philosophy of Indian Constitution.

UNIT II - Contours of Constitutional Rights and Duties

Fundamental Rights, Right to Equality, Right to Freedom, Right against Exploitation, Right to Freedom of Religion, Cultural and Educational Rights, Right to Constitutional Remedies, Directive Principles of State Policy, Fundamental Duties.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Know the Fundamental Rights & Duties.
2. Understand the Directive Principles of State Policy.

UNIT III - Organs of Governance

Parliament, Composition, Qualifications and Disqualifications, Powers and Functions- Executive, President, Governor, Council of Ministers, Judiciary, Appointment and Transfer of Judges, Qualifications, Powers and Functions.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Understand Parliament, Composition, Qualifications and Disqualifications, Powers and Functions- Executive, President, Governor, Council of Ministers, Judiciary.
2. Know the Appointment and Transfer of Judges, Qualifications, Powers and Functions.

UNIT IV - Local Administration

District's Administration head: Role and Importance, Municipalities: Introduction, Mayor and role of Elected Representative CEO of Municipal Corporation, Panchayati Raj: Introduction, PRI: ZilaPanchayat, Elected officials and their roles, CEO ZilaPanchayat: Position and role,

Block Level: Organizational Hierarchy (Different departments), Village level: Role of Elected and Appointed officials, Importance of grass root democracy.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Understand Role and Importance, Municipalities: Introduction, Mayor and role of Elected Representative CEO of Municipal Corporation.
2. Understand PRI, Zilla Panchayat, Elected officials and their roles, CEO Zilla Panchayat, Position and role.
3. Know the importance of Organizational Hierarchy (Different departments), Village level: Role of Elected and Appointed officials, Importance of grass root democracy.

UNIT V - Election Commission

Role and Functioning, Chief Election Commissioner and Election Commissioners, State Election Commission: Role and Functioning, Institute and Bodies for the welfare of SC/ST/OBC and women.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Understand the Role and Functioning of election commissioners & State Election Commission.
2. Know The Institute and Bodies for the welfare of SC/ST/OBC and women.

Text Books:

1. The Constitution of India, 1950 (Bare Act), Government Publication.
2. The Constitution of India Prof. S.R. Bhansali.

Reference Books:

1. Dr. S. N. Busi and Dr. B. R. Ambedkar framing of Indian Constitution, First Edition, 2015.
2. M.P. Jain, Indian Constitution Law, 7thEdition, Lexis Nexis,2014.
3. M.P. Jain, Indian Constitution Law, 6thEdition, Lexis Nexis,2010.
4. V. N Shukla's, "Constitution of India" Mahendrapaul Singh, Eastern book Complany, 13th edition, 2012.

Course Outcomes:

At the end of the course, student will be able to

1. Discuss the history, philosophy, fundamental rights, duties, organs of Governance, Local administration & Election commission in Indian constitution.
2. Explain History, Drafting Committee, Philosophy Preamble, Salient Features of Indian constitution.
3. Discuss fundamental rights& duties of a citizen & Directive Principles of State Policy.
4. Explain the organs of governance and their functions for to establish the regulatory framework for political activity and the governance process in a country.
5. Explain the importance of local administration and its organizational Hierarchy.
6. Discuss the functioning of election commission and the Institute and Bodies for the welfare of SC/ST/OBC and women.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Web Development Technologies

(Computer Science and Engineering)

III B.Tech - I Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA05501	**PCC**	3	0	0	3	40	60	100

Objectives

- To explore the fundamental concepts of web, internet protocols, client-server model, and applications.
- To acquire knowledge in design and implement web applications.
- To demonstrates the uses of scripting languages.
- To explore the fundamental concepts of AngularJS, React JS and Node.js.

Unit I – Introduction to Web and Internet

Introduction to Networks, Internet, Web Protocols, URL, Domain Name.

Web Browsers and Web Servers: Apache HTTP Server, Apache Tomcat Server, XAMPP Server, Installations of above Servers, Web System Architecture, Client-Side and Server-Side Scripting Technologies.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Explore fundamentals of Internet and Web.
2. Illustrate Client-Server Model.
3. Demonstrate the installation of Web Servers.
4. Explain the Scripting Technologies.

Unit II – Web Designing

HTML5: Basics of HTML5 Elements, Form Elements, Input Types and Media Elements.

CSS3: Properties, Selectors, Types of CSS. Introduction to Bootstrap & its Components.

XML: Document Type Definition (DTD), XML Schema XML-DOM, XSLT.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Create HTML forms, tables, Lists.
2. Apply various styles using CSS.
3. Create DTD, XML based web applications.
4. Explain the concepts of bootstrap components.

Unit III – JavaScript

Basics: Introduction to JavaScript, Data Types and Variables, Expressions, Operators, Statements, Objects, Array, Functions, Regular Expressions, Windows Object, Scripting Documents, Handling Event.

Advanced JavaScript: Introduction to JSON – JSON Structure, The jQuery Library, Introduction to AJAX.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Explain the basic concepts of JavaScript.
2. Demonstrate web page using jQuery.
3. Create AJAX based applications

Unit IV – PHP & DATABASE CONNECTIVITY

Introduction to PHP: Introduction, Download, Install and Configure of PHP, Anatomy of A PHP.

Overview of PHP Data Types and Concepts: Variables and Data Types, Operators, Expressions and Control Statements, Strings, Arrays and Functions, Regular Expressions.

PHP advanced concepts: Using Cookies, Using HTTP Headers, Using Sessions, Authenticating Users:

MySQL Basics: Introduction to MySQL, Querying Single and Multiple MySQL databases with PHP – PHP data objects.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Learn how to configure PHP.
2. Demonstrate the need of Regular Expressions.
3. Explain PHP advanced concepts like Cookies, HTTP Headers, Sessions.
4. The basic concepts of Database Connectivity using MySQL.

Unit V – Application Development Using Node.js

Overview of Node.js: Introduction to Node.js, Installing Node.js, Understanding the node.js event model, Introduction to Mongodb, Accessing Mongodb (CRUD) from Node.js.

Learning Outcomes: At the end of this unit, the student will be able to

1. Explain the fundamentals of Node.js.
2. Explain the importance of mongodb.
3. Demonstrate various advance concepts of Node,js.

Text Books:

1. Chris Bates, Web Programming: Building Internet Applications, 3rd Edition.
2. Brad Dayley, Brendan Dayley, and Caleb Dayley, Node.js, MongoDB and Angular Web Development: The definitive guide to using the MEAN stack to build web applications, 2nd Edition, Pearson Education, 2018.

Reference Books:

1. Paul Deitel, Harvey Deitel, Abbey Deitel, Internet & World Wide Web - How to Program, 5th edition, Pearson Education, 2012.
2. Beginning PHP and MySQL, 3rd Edition, Jason Gilmore, A press Publications (Dream tech.).
3. David Flanagan, "JavaScript: The Definitive Guide, Sixth Edition", O'Reilly Media, 2011.

Course Outcomes:

At the end of the course, student will be able to

1. Describe fundamentals of web and internet technologies to design web pages.
2. Demonstrate installation of different web servers.
3. Implement dynamic web pages effectively by using HTML5, CSS3 and XML.
4. Develop an application using JS objects, JQuery, JSON and AJAX.
5. Implement server-side programming using Cookies, HTTP Headers, Sessions in PHP.
6. Develop application using Node.js, React JS and MongoDB.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Computer Networks

(Common to CSE, CSM & CSD)

III B.Tech - I Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA05502	**PCC**	3	0	0	3	40	60	100

Objectives

- Explore the fundamental concepts of computer networking, protocols, architectures, and applications.
- Acquire knowledge in design, implement and analyze performance of layers in networking.
- Explore the fundamental concepts of network security.

Unit I – Introduction to Computer Networks

Introduction: Uses of Computer Networks, Network Hardware, Network Software, Reference Models.

Physical Layer: Theoretical basis for Data Communications, Transmission media, Circuit Switching and Packet Switching.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Explore fundamentals of Network Hardware and Software.
2. Compare and Contrast TCP/IP Reference Model with OSI Model.
3. Illustrate various transmissions medium and its communication technologies.

Unit II – Link Layer

Data Link Layer: Design issues, Error Detection and Correction Layer, Elementary Data Link Protocols, Sliding Window Protocols, Example Data Link Protocols.

Medium Access Control Layer: Channel Allocation Problem, Multiple Access Protocols, Ethernet, Wireless LAN.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Illustrate design issues of Data Link Layer.
2. Explore various types of elementary and sliding window protocols.
3. Explore Multiple Access protocol and switching.

Unit III – Communication Layer

Network Layer: Design issues, Routing Algorithms, Congestion Control Algorithms, Quality of Service, Internetworking, and Network Layer in the internet.

Transport Layer: Transport Services, Elements of Transport Protocols, Congestion Control, UDP, TCP, Performance issues, Delay-Tolerant Networking.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Explain the design issues, algorithms and QoS of network layer.
2. Illustrate transport services in the transport layer.
3. Illustrate Delay-Tolerant Networking.

Unit IV – Application Layer

Application Layer: DNS, Remote Logging, File Transfer, Electronic-Mail, WWW, HTTP, Network Management Systems, SNMP, Streaming Audio and Video, Content Delivery.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Illustrate various types of Applications in Application Layer.
2. Explore various Application technologies like DNS, Remote Logging, File Transfer, E-mail.
3. Describe the fundamentals of Network Management Systems.

Unit V – Network Security

Network Security: Cryptography, Symmetric-key Algorithms, Public-Key Algorithms, Digital Signatures, Management of Public keys, Communication Security, Authentication Protocols, Email Security, Web Security and Social issues.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Explain the fundamentals and principles of Cryptography.
2. To demonstrate Symmetric-key Algorithms, Public-Key Algorithms and Digital Signatures
3. To demonstrate various security concepts like Authentication Protocols, Communication, Email and Web security in social media.

Text Books:

1. "Computer Networks", Andrew S. Tanenbaum and David J. Wetharall, Prentice Hall, fifth Edition, 2011.
2. "Data Communications and Networking", Behrouz A. Forouzon, Sophia Chung Fegan, McGraw Hill Higher Education fifth Edition, 2012.

Reference Books:

1. "Computer Networks: A Systems Approach", Larry Peterson and Bruce Davie, 5th Ed, The Morgan Kaufmann Series, Elsevier, 2011.
2. "Computer Networking: A Top-Down Approach Featuring the Internet", J.F. Kurose and K.W.Ross, 6th Ed., Pearson Education, 2012.
3. "Data and Computer Communications", William Stallings, Pearson Education, 10th Ed, 2013.

Course Outcomes:

At the end of the course, student will be able to

1. Examine the different building blocks of Communication network and its architecture.
2. Contrast different types of networks and analyze the performance of network.
3. Identify and examine the various solutions in design issues of Data Link Layer.
4. Compare and Contrast various types of Routing Algorithms and Congestion Control Algorithms, Transport services, protocols.
5. Illustrate Fundamental applications of Application Layer.
6. Explain Network Security concepts.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Operating Systems

(Computer Science and Engineering)

III B.Tech - I Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA05503	**PCC**	3	0	0	3	40	60	100

Objectives

- To introduce the operating system concepts, designs and provide skills required to implement the services.
- To describe the trade-offs between conflicting objectives in large-scale system design.
- To develop the knowledge for application of the various design issues and services.

Unit I – Introduction to Operating System

Introduction: Operating System Operations, Resource Management, Security and Protection, Virtualization, Distributed Systems, Computing Environments.

Operating-System Structures: Operating-System Services, User and Operating-System Interface, System Calls, System Services.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Determine the functionality of operating systems
2. Illustrate the process of virtualization, computing environments.
3. Illustrate the role of services, interfaces and system calls.

Unit II – PROCESS MANAGEMENT

Processes: Process Concept, Scheduling, Operations on Processes, Inter process Communication: Shared-Memory Systems, Message-Passing Systems, Examples, Communication in Client–Server Systems. CPU Scheduling: Scheduling Criteria, Scheduling Algorithms, Threads.

Process Synchronization: The critical-section problem, Petersons Solution, Synchronization Hardware, Mutex Locks, Semaphores, Classic problems of synchronization, Monitors.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Demonstrate the process features like scheduling, creation and termination.
2. Explore IPC in Shared-Memory Systems and Message-Passing Systems.
3. Demonstrate Client-Server Communication Systems.

Unit III – MEMORY MANAGEMENT

Contiguous Memory Allocation, Swapping, Paging, Page Replacement algorithms, Thrashing, Memory Compression.

Deadlocks: System Model, Deadlock Characterization, Methods of handling Deadlocks, Deadlock prevention, Detection and Avoidance, Recovery from deadlock.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Investigate various techniques of allocating memory to processes.
2. Illustrate concepts of demand-paging, page-replacement algorithms.
3. Illustrate different methods for preventing or avoiding deadlocks.

Unit IV – STORAGE MANAGEMENT & FILE SYSTEM

Mass-Storage Structure: Overview of Mass-Storage Structure, Disk Scheduling, Storage Attachment, RAID Structure.

I/O Systems: I/O Hardware**,** Application I/O Interface, Kernel I/O Subsystem, Transforming I/O Requests to Hardware Operations.

File-System: File Concept, Access Methods, Directory Structure, Protection, Memory-Mapped Files, File system structure and Implementation.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Evaluate disk scheduling algorithms.
2. Explore file-system protection.
3. Illustrate local file systems, directory structures and remote file systems.

Unit V – Security and Protection

Protection: Goals, Principles and domain, Access Matrix, Implementation of Access Matrix, Access control, Revocation of Access Rights.

Security: The Security problem, Program threats, System and Network threats, Cryptography as a security tool.

Learning Outcomes: At the end of this unit, the student will be able to

1. Illustrate goals and principles of protection in modern operating system.
2. Demonstrate security threats and attacks.
3. Demonstrate various virtual machine technologies.

Text Books:

1. Operating System Concepts, Abraham Silberchatz, Peter B. Galvin, Greg Gagne, Wiley, Tenth Edition, 2018.
2. Operating Systems: Design And Implementation, Andrew S. Tanenbaum, Albert S. Woodhull, Pearson, 3rd Edition, 2015.

Reference Books:

1. Operating Systems, A Spiral Approach, Ramez Elmasri, A.Gil Carrick, David Levine, McGrawHill Higher Education, 2010.
2. Operating Systems, Three Easy Pieces, Remzi H. Arpaci-Dusseau, Andrea C. Arpaci-Dusseau, Arpaci-Dusseau Books, 2015.
3. Operating Systems: and design Principles,5th Edition,William Stallings,PHI.

Course Outcomes:

At the end of the course, student will be able to

1. Explain the fundamentals of operating systems like process, memory, storage, file system, security and protection.
2. Illustrate various operating System services, interfaces and system calls.
3. Demonstrate critics of process management and IPC.
4. Implement page replacement algorithms, memory management techniques and deadlock issues.
5. Illustrate architecture of file systems and I/O systems for mass storage structures.
6. Utilize the methods of operating system security and protection.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Data Warehousing and Data Mining

(Professional Elective-I)

III B.Tech - I Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA05504	**PEC**	3	0	0	3	40	60	100

Objectives

- Familiarize with mathematical foundations of data mining tools.
- Introduce classical models and algorithms in data warehouses and data mining.
- Investigate the kinds of patterns that can be discovered by association rule mining, classification and clustering.
- Explore data mining techniques in various applications like social, scientific and environmental context.

Unit I – Introduction to Data Warehousing

Basic Concepts – Data Warehousing Components, Building a Data Warehouse, Database Architectures for Parallel Processing, Parallel DBMS Vendors, Multidimensional Data Model – Data Warehouse Schemas for Decision Support, Concept Hierarchies, Characteristics of OLAP Systems, Typical OLAP Operations, OLAP and OLTP.

Learning Outcomes:

At the end of the unit, students will be able to

1. Identify the component of Data warehouse.
2. Create the architecture of Data warehouse.
3. Apply different types of OLAP operations.

Unit II – Introduction to Data Mining

Knowledge Discovery Process , Data Mining Techniques ,Issues, applications, Data Objects and attribute types, Statistical description of data, Data Preprocessing ,Cleaning, Integration, Reduction, Transformation and discretization, Data Visualization, Data similarity and dissimilarity measures.

Learning Outcomes:

At the end of the unit, students will be able to

1. Summarize the data processing steps.
2. Apply data cleaning process.

Unit III – Association Rule Mining

Mining Frequent Patterns, Associations and Correlations.

Mining Methods- Pattern Evaluation Method, Pattern Mining in Multilevel, Multi-Dimensional Space, Constraint Based Frequent Pattern Mining, Classification using Frequent Patterns.

Learning Outcomes:

At the end of the unit, students will be able to:

1. Understand Association Rules
2. Apply different Mining Methods

3. Review Classification using Frequent Patterns

Unit IV – Classification and Clustering

Decision Tree Induction, Bayesian Classification, Rule Based Classification, Classification by Back Propagation, Support Vector Machines, Lazy Learners.

Model Evaluation and Selection-Techniques to improve Classification Accuracy.

Clustering Techniques – Cluster analysis, Partitioning Methods, Hierarchical Methods, Density Based Methods, Grid Based Methods, Evaluation of clustering, Clustering high dimensional data, Clustering with constraints, Outlier analysis-outlier detection methods.

Learning Outcomes:

At the end of the unit, students will be able to:

1. Creating Decision Tree
2. Evaluate Classification techniques

Unit V – Weka Tool

Datasets – Introduction, Iris plants database, Breast cancer database, Auto imports database, Introduction to WEKA.

The Explorer – Getting started, Exploring the explorer, Learning algorithms, Clustering algorithms, Association–rule learners.

Learning Outcomes:

At the end of the unit, students will be able to:

1. Investigate WEKA tool
2. Explain learning, clustering algorithms

Text Books:

1. Jiawei Han and Micheline Kamber, —Data Mining Concepts and Techniques, Third Edition, Elsevier, 2012.
2. Pang Ning Tan, Michael Steinbach, Vipin Kumar - - Introduction to Data Mining, Pearson Publishers, 2016

Reference Books:

1. Alex Berson and Stephen J.Smith, —Data Warehousing, Data Mining & OLAP‖, Tata McGraw – Hill Edition, 35th Reprint 2016.
2. K.P. Soman, Shyam Diwakar and V. Ajay, —Insight into Data Mining Theory and Practice, Eastern Economy Edition, Prentice Hall of India, 2006.
3. Ian H.Witten and Eibe Frank, —Data Mining: Practical Machine Learning Tools and Techniques, Elsevier, Second Edition.

Course Outcomes:

At the end of the course, student will be able to

1. Illustrate a Data warehouse system and business analysis.
2. Describe building blocks of databases and data warehouses.
3. Implement pre-processing and visualization techniques for knowledge analysis.
4. Use frequent pattern and association rule mining techniques to calculate support and confidence of business data.
5. Implement classification techniques and clustering methods for a given data.
6. Use WEKA tool to explore learning and clustering algorithms for a given database.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Wireless Sensor Networks

(Professional Elective-I)

III B.Tech - I Semester **SRIT R20**

Course Code	Category	Hours/Week			Credits	Maximum Marks		
		L	T	P	C	CIA	SEE	Total
R204GA05505	PEC	3	0	0	3	40	60	100

Objectives

- To study the concepts of sensor networks.
- To understand the WSN node Architecture and Network Architecture.
- To identify the Wireless Sensor Network Platforms.
- To design and develop wireless sensor node.
- To study the research issues in different layers of sensor networks.

Unit I – Introduction

Introduction: Wireless Networks, Adhoc Networks, Comparison of Adhoc and Sensor Networks, Applications of Sensor Networks, Challenges and Hurdles in Sensor network design.

Wireless Transmission Technology and Systems: Bluetooth; IEEE 802.11a/b/g/n series of wireless LANs; ZigBee; Radio-frequency identification (RFID).

Sensor-node Architecture: Hardware components, Energy consumption of sensor nodes, Operating systems and execution environments, Physical layer and transceiver design considerations in Wireless Sensor Networks.

Learning Outcomes:

1. Understand self-configured wireless networks.
2. Understand infrastructureless wireless networks.

Unit II – Medium Access Protocols

Medium Access Control Protocols for Wireless Sensor Networks: Fundamentals of MAC Protocols, Performance Requirements.

Types of MAC protocols - Schedule-Based and Random Access-Based Protocols, Sensor-MAC, Zebra-MAC.

Learning Outcomes:

1. Evaluate the performance of schedule based and random Medium Access Control protocols for power consumption, fairness, channel utilization and control packet overhead.

Unit III – Network Protocols

Routing Protocols for Wireless Sensor Networks: Fundamentals of Routing Protocols, Performance Requirements, Routing Strategies in Wireless Sensor Networks - Flooding and its variants, LEACH, Power-Efficient Gathering in Sensor Information Systems, Directed diffusion, Geographical routing.

Learning Outcomes:

1. Evaluate the performance of Geographic routing protocols for power consumption, scalability and latency parameters.

Unit IV – End –To- End Delivery

Transport Control Protocols for Wireless Sensor Networks: Traditional Transport Control Protocols-TCP, UDP; Feasibility of Using TCP or UDP for WSNs, Transport Protocol Design Issues, Existing Transport Control Protocols- CODA (Congestion Detection and Avoidance), ESRT (Event-to-Sink Reliable Transport) Performance of Transport Control Protocols.

Learning Outcomes:

1. Evaluate the performance of transport control protocols for congestion detection and avoidance, reliability and control packet overhead parameters.

Unit V – Middleware for Wireless Sensor Networks

Middleware for Wireless Sensor Networks: WSN Middleware Principles, Middleware Architecture, Existing Middleware-MiLAN (Middleware Linking Applications and Networks), IrisNet (Internet-Scale Resource Intensive Sensor Networks Services).

Time Synchronization and Localization: Time synchronization protocols based on sender/receiver synchronization, Localization approaches- proximity, trilateration and triangulation.

Learning Outcomes:

1. Understand the Sensor management, sensor network middleware.

Text Books:

1. Siva Ram Murthy and B. S. Manoj, Ad hoc Wireless Networks Architecture and Protocols, 2nd edition, Pearson Edition, 2007.
2. Charles E. Perkins, Ad hoc Networking, Addison – Wesley, 2000.

Reference Books:

1. Stefano Basagni, Marco Conti, Silvia Giordano and Ivan stojmenovic, Mobile ad-hoc networking, Wiley-IEEE press, 2004.
2. Mohammad Ilyas, The handbook of ad-hoc wireless networks, CRC press, 2002.
3. T. Camp, J. Boleng, and V. Davies "A Survey of Mobility Models for Ad-hoc Network".

Course Outcomes:

At the end of the course, student will be able to

1. Describe the principles of sensor networks and mobile ad hoc networks, and their impact on protocol design.
2. Develop MAC and routing protocols for sensor networks.
3. Develop efficient routing protocols for sensor and mobile networks.
4. Evaluate the performance of transport control protocols for congestion detection and avoidance, reliability and control packet overhead parameters.
5. Understand and develop information dissemination protocols for sensor and mobile networks.
6. Evaluate the performance of Geographic routing protocols for power consumption, scalability and latency parameters.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Software Testing Methodologies

(Professional Elective-I)

III B.Tech - I Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA05506	**PEC**	3	0	0	3	40	60	100

Objectives

- To study fundamental concepts in software testing and discuss various software testing issues and solutions in software unit, integration, regression and system tests.
- To learn how to plan a test project, design test cases and data, conduct testing, manage software problems and defects, generate a test report.
- To expose the advanced software tests concepts such as object-oriented software test methods, web-based and component-based software testing.
- To understand software test automation problems and solutions.
- To learn how to write software test documents and communicate with engineers in various forms.

Unit I – Software Testing

Introduction: Evolution, Myths & Facts, Goals, Psychology, definition, Model for testing, Effective Vs Exhaustive Software Testing.
Software Testing Terminology and Methodology: Software Testing Terminology, Software Testing Life Cycle, Software Testing Methodology.
Verification and Validation: Verification & Validation Activities, Verification, Verification of Requirements, High level and low-level designs, verifying code, Validation.

Learning Outcomes:
At the end of this unit, the student will be able to
1. Understand the fundamentals of testing.
2. Analyze Software Testing Life Cycle.
3. Illustrate Software Testing Methodology.

Unit II – Testing Techniques

Dynamic Testing: Black Box testing techniques, Boundary Value Analysis, Equivalence class Testing, State Table based testing, Decision table-based testing, Cause-Effect Graphing based testing, Error guessing.
White-Box Testing: need, Logic Coverage criteria, Basis Path testing, Graph matrices, Loop testing, data flow testing, mutation testing.

Learning Outcomes:
At the end of this unit, the student will be able to
1. Understand the process of various Black Box testing techniques.
2. Understand the process of various White Box testing techniques.
3. Differentiate between Black Box testing and White Box testing techniques.

Unit III – Static Testing

Inspections, Structured Walkthroughs, Technical Reviews.
Validation activities: Unit testing, Integration Testing, Functional testing, system testing, acceptance testing.
Regression testing: Progressive Vs regressive testing, Regression test ability, Objectives of regression testing, Regression testing types, Regression testing techniques.

Learning Outcomes:
At the end of this unit, the student will be able to
1. Understand the process of static testing.
2. Analyze various validation activities.
3. Differentiate between Progressive and Regressive testing.

4. Understand the Regression Testing Techniques.

Unit IV – Efficient Test Suite Management

Growing nature of test suite, Minimizing the test suite and its benefits, test suite prioritization, Types of test case prioritization, prioritization techniques, measuring the effectiveness of a prioritized test suite. Software Quality Management: Software Quality metrics, SQA models. Debugging: process, techniques, correcting bugs.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Understand test suite minimization.
2. Understand the test case prioritization techniques.
3. Understand the Software Quality Management.
4. Analyze the application of a cost of software quality system.

Unit V – Automation and Testing Tools

Need for automation, categorization of testing tools, selection of testing tools, Cost incurred, Guidelines for automated testing, overview of some commercial testing tools such as Win Runner, Load Runner, Jmeter and JUnit. Test Automation using Selenium tool.
Testing Object Oriented Software: basics, Object oriented testing.
Testing Web based Systems: Challenges in testing for web-based software, quality aspects, web engineering, testing of web-based systems, Testing mobile systems

Learning Outcomes: At the end of this unit, the student will be able to

1. Understand the process of automation.
2. Implement automated testing using Selenium tool.
3. Implement Object Oriented Software Testing.
4. Implement web-based software testing.

Text Books:

1. Software Testing, Principles and Practices, Naresh Chauhan, Oxford.
2. Software Testing, Yogesh Singh, CAMBRIDGE.

Reference Books:

1. Foundations of Software testing, Aditya P Mathur, 2ed, Pearson.
2. Software testing techniques – Baris Beizer, Dreamtech, second edition.
3. Software Testing, Principles, techniques and Tools, M G Limaye, TMH.

Course Outcomes:

At the end of the course, student will be able to

1. Identify and understand various software testing problems, apply software testing knowledge and engineering methods and solve these problems by designing and selecting software test models, criteria, strategies, and methods.
2. Design and conduct a software test process for a software project.
3. Analyze the needs of software test automation.
4. Use various communication methods and skills to communicate with their teammates to conduct their practice-oriented software testing projects.
5. Basic understanding and knowledge of contemporary issues in software testing, such as component-based, web based and object-oriented software testing problems.
6. Write test cases for given software to test it before delivery to the customer and write test scripts for both desktop and web-based applications.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Web Development Technologies Lab

(Computer Science and Engineering)

III B.Tech-I Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA05509	**PCC**	0	0	3	1.5	40	60	100

Objectives

- To develop client-server application using modern technologies.
- To build single-page web applications using AngularJS.
- Create dynamic web pages using Java Script and AJAX.
- To enhance the knowledge to Node.js, ReactJS and mongodb.

List of Experiments

1. Write a procedure to download and Installation of Apache HTTP Server, Apache Tomcat Server and XAMPP Server.
2. To create an HTML page by using basic HTML Tags, Table Tags, List Tags, Image Tags, anchor Tags.
3. To create a simple student Bio - data form using html5. It should contain the following First Name, Last Name (text box), address (multiline text box), Gender (radio button Male, Female, Transgender), Email Id, Phone No., Known Technologies (checkboxes – C, Java, Angular JS, ReactJS, Node.js etc.), Extracurricular activities (text box), Nationality (combo box), Submit and Reset button.
4. Design an HTML page by using frames such that page is divided into 3 frames 20% on top to show contents of College Name and Department Name, 40% in left to show list of faculty names, remaining on right to show faculty profiles/details (Note: if you click particular faculty, it will display only that faculty details).
5. Design the webpage by applying the different styles using inline, internal & external style sheets.
6. Write an XML file which will display the Book information which includes the following: a) Title of the book b) Author Name c) ISBN number d) Publisher name e) Edition f) Price.
7. Create a Schema to describe a library. Library has one or more - books, members and staffs. (a) Each book has BookID (Attribute), Title, one or more Authors, Publisher Year of Publication, ISBN and Price. (b) Each Member has MemeberID (Attribute), Name, Address, Phone number, (c) Each Staff has StaffID(Attribute), Name, Address, Phone number. Each Author has AuthorID(Attribute), Name, Address, Phone number. (d) Each Publisher has PublisherID(Attribute), Name, Address, Phone number. Use the above DTD in a sample XML document.
8. Write a PHP program to keep track of the number of visitors visiting the web page and to display this count of visitors, with proper headings.
9. Implement the web application using PHP, the user is first served a login page which takes user's name and password. After submitting the details, the server checks these values against the data from a database and takes the following decisions. If name and password match, serves a welcome page with user's full name. If name matches and password doesn't match, then serves "password

mismatch" page. If name is not found in the database, serves a registration page, where user's full name is asked and on submitting the full name, it stores, the login name, password and full name in the database (hint: use session for storing the submitted login name and password).

10. Write a PHP program to Insert, update and delete using student database.
11. a). Implement the web application using PHP, A web application that lists all cookies stored in the browser on clicking "List Cookies" button. Add cookies if necessary.

    b). Implement the web application using PHP, which takes a name as input and on submit it shows a hello <name> page where<name> is taken from the request. It shows the start time at the right top corner of the page and provides a logout button. On clicking this button, it should show a logout page with "Thank You". <name> message with the duration of usage (hint: Use session to store name and time).
12. Write a Program to hide paragraph using jQuery.
13. Create a web page using Bootstrap.
14. Create an HTML page to perform CRUD operations using ReactJS, Node.js and mongoDB.

Reference Books:

1. Web Technologies: HTML, JavaScript, PHP, Java, JSP, XML, And AJAX, Black Book, Kogent Learning Solutions Inc.
2. Web Technologies, Uttam K. Roy, 1st edition 7th impression, 2012, Oxford Higher Education.

Course Outcomes:

At the end of the course, student will be able to

1. Install Apache HTTP Server, Apache Tomcat Server and XAMPP Server to run web applications in local host.
2. Create dynamic web pages by using HTML and CSS, XML.
3. Design interactive web pages using JavaScript and its objects.
4. Implement client and server-side scripting using PHP and CRUD operations.
5. Create a web page using cookies, sessions and HTTP headers in PHP.
6. Develop an application using Node.js, Reactjs and MongoDB.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Computer Networks and Operating Systems Lab

(Computer Science and Engineering)

III B.Tech - I Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA05510	**PCC**	0	0	3	1.5	40	60	100

Objectives

- To familiarize with packet tracer simulator.
- To explore on scheduling algorithms, synchronization Problems and file and Memory management Mechanisms.

List of Experiments:

1. Installation and Configure CISCO Packet Tracer Simulator.
2. Implementation of IP Addressing in simple network.
3. Static Routing Configuration.
4. Configuring Routing Information Protocol (RIP).
5. Configuring Open Shortest Path First (OSPF) Protocol.
6. Implement DHCP & DNS.
7. Demonstration of Telnet & SSH.
8. Implement e-mail using SMTP / POP3.
9. Write a program to stimulate following CPU scheduling Algorithms.
   a) FCFS b) SJF c) Round Robin d) Priority.
10. Write a program to stimulate Producer-Consumer Problem using Semaphores.
11. Write a program to stimulate Dining-philosophers problem.
12. Write a Program to stimulate MVT and MFT.
13. Write a Program to stimulate the following contiguous memory allocation techniques.
    a) Worst Fit b) Best Fit c) First Fit
14. Write a Program to stimulate the following page replacements algorithms.
    a) FIFO b) LRU c) OPTIMAL
15. Write a Program to stimulate the following File Organization Techniques.
    a) Single Level Directory b) Second Level Directory
16. Write a Program to stimulate the following file allocation strategies.
    a) Sequential b) Indexed c) Linked
17. Write a Program to stimulate the following Bankers algorithm.
    a) Dead Lock Avoidance b) Dead Lock Prevention
18. Write a Program to stimulate the following Disk scheduling Algorithms.
    a) FCFS b) SCAN c) C-SCAN

Reference Books:

1. "Computer Networking: A Top-Down Approach Featuring the Internet", J.F. Kurose and K.W.Ross, 6th Ed., Pearson Education, 2012.
2. An Introduction to Operating Systems, P.C.P Bhatt, 2nd edition, PHI.
3. Compilers: Principles, Techniques, and Tools (Second Edition) Alfred Aho, Monica Lam, Ravi Sethi, and Jeffrey Ullman. Addison-Wesley.

Course Outcomes:

At the end of the lab, students will be able to

1. Demonstrate implementation of Packet Tracer.
2. Implement the routing protocols.

3. Create IP address for both static and dynamic protocols.
4. Implement the IPC between Processes and Synchronization mechanism.
5. Stimulate the algorithms of CPU Scheduling, Disk Scheduling, Page replacements and Bankers.
6. Stimulate the File and Memory Allocation techniques.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Essence of Indian Traditional Knowledge

(Common to all Branches)

III B.Tech-I Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA5MC03	**NCMC**	2	0	0	0	40	0	40

Objective:

- To facilitate the students with the concepts of Indian traditional knowledge and to makethem understand the Importance of roots of knowledge system.

UNIT I - Introduction to Traditional knowledge

Define traditional knowledge, nature and characteristics, scope and importance, kinds of traditional knowledge, the physical and social contexts in which traditional knowledge develop, the historical impact of social change on traditional knowledge systems. Indigenous Knowledge (IK), characteristics, traditional knowledge vis-à-vis indigenous knowledge, traditional knowledge Vs western knowledge traditional knowledge vis-à-vis formal knowledge.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Know about the importance of traditional knowledge.

UNIT II - Protection of Traditional knowledge

The need for protecting traditional knowledge Significance of TK Protection, value of TK inglobal economy, Role of Government to harness TK.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Understand the need for protection of traditional knowledge.

UNIT III - Legal frame work and TK

A: The Scheduled Tribes and Other Traditional Forest Dwellers (Recognition of Forest Rights)Act, 2006, Plant Varieties Protection and Farmer's Rights Act, 2001 (PPVFR Act); B: The Biological Diversity Act 2002 and Rules 2004, the protection of traditional knowledgebill, 2016. Geographical indicators act 2003.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Understand various legal acts related to traditional knowledge.

UNIT IV - Traditional knowledge and Intellectual Property

Systems of traditional knowledge protection, Legal concepts for the protection of traditional knowledge, Certain non IPR mechanisms of traditional knowledge protection, Patents and traditional knowledge, Strategies to increase protection of traditional knowledge, global legal FORA for increasing protection of Indian Traditional Knowledge.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Understand the intellectual property rights related to traditional knowledge.

UNIT V - Traditional knowledge in Different sectors

Traditional knowledge and engineering, Traditional medicine system, TK and biotechnology, TK in agriculture, Traditional societies depend on it for their food and healthcare needs, Importance of conservation and sustainable development of environment, Management ofbiodiversity, Food security of the country and protection of TK.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Know about traditional knowledge in different sectors.

Text Books:

1. Traditional Knowledge System in India, by Amit Jha, 2009.
2. Traditional Knowledge System and Technology in India by Basanta Kumar Mohanta andVipin Kumar Singh, Pratibha Prakashan 2012.

Reference Books:

1. Traditional Knowledge System in India by Amit Jha Atlantic publishers, 2002
   "Knowledge Traditions and Practices of India" Kapil Kapoor1, Michel Danino2

Course Outcomes:

At the end of the course, student will be able to

1. Know about the roots of Indian Traditional Knowledge and legal acts related to it.
2. Explain the concept of Traditional knowledge and its importance
3. Know the need and importance of protecting traditional knowledge.
4. Know the various enactments related to the protection of traditional knowledge.
5. Describe the concepts of Intellectual property to protect the traditional knowledge.
6. Discuss the traditional knowledge in different sectors.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Compiler Design

(Common to CSE, CSM & CSD)

III B.Tech - II Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA05513	**PCC**	3	0	0	3	40	60	100

Objectives

- Understand the basics of constructing compilers by applying mathematics and engineering principles.
- Construct a system for parsing the sentences in a context free grammar.
- Illustrate the various optimization techniques for designing compilers.
- Explore the process of Intermediate code Generation.
- Illustrate the methods of implementing a Code Generator for compilers.

Unit I – Introduction to Compilers and Lexical Analysis

Overview of Compilation: Introduction, Structure of compiler, phases of the compiler.
Lexical Analysis: The role of the lexical analyzer, input buffering, specification of tokens, recognition of tokens, the lexical analyzer generator (LEX/FLEX).

Learning Outcomes:

At the end of this unit, the student will be able to

1. Know the various types of language translators.
2. Explain the functions of each phase of the compiler.
3. Illustrate the usage of LEX tool.

Unit II – Top-Down Parsers

Syntax Analysis (Part-1): Introduction, context free grammars, Top-Down parsing: Brute force parsing, recursive descent parsing, predictive parsing and error recovery in predictive parsing.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Know the importance of parsing techniques.
2. Understand the construction of top-down parsers.

Unit III – Bottom-Up Parsers

Syntax Analysis (Part-2): Shift reduce parsing, operator precedence parsing. Introduction to LR Parser, Canonical LR and look head LR, error recovery in LR parsers, parser generator (YACC).

Learning Outcomes:

At the end of this unit, the student will be able to

1. Know the construction of LR parsers.
2. Differentiate top down and Bottom-Up parser.
3. Illustrate the error recovery in LR parsers.

Unit IV – Syntax Directed Translation & Intermediate Code Generation

Syntax-Directed Translation: Syntax-Directed Definitions, evaluation orders for SDD's, application of SDT, SDT schemes, implementing L-attribute SDD's.
Intermediate Code Generation: Variants of syntax trees, three address code, Quadruples, Triples, Type expressions, Type equivalence, Type checking, Translation of expressions, control flow statements, switch statement, procedures, back patching.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Explain the process of syntax directed translation.
2. Write the intermediate code using three address codes.
3. Understand the symbol table organization.

Unit V – Code Optimization and Generation

Machine Independent Optimization: The principal sources of optimization, basic blocks, flow graphs, loop optimization, DAG representation of basic block, local optimization, peephole optimization.
Code Generation: Issues in the design of a code generator, a simple code generator, register allocation and assignment.

Learning Outcomes:
At the end of this unit, the student will be able to

1. Reduce the code using machine independent optimization techniques.
2. Explain the various code generation algorithms.
3. Design a compiler for new languages.

Text Books:

1. Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman, Compilers: Principles, Techniques and Tools, Second Edition, Pearson Education, 2009.
2. Compiler Design, K. Muneeswaran., Oxford University Press, 2012.

Reference Books:

1. Compiler Construction, K.V.N Sunitha, Pearson, 2013.
2. Engineering a Compiler, Second Edition, Keith D. Cooper & Linda Torczon., Morgan Kaufmann, Elsevier, 2011.
3. Compilers Principles and Practice, Parag H. Dave, Himanshu B. Dave., Pearson.

Course Outcomes:

At the end of the course, student will be able to

1. Describe the functionalities of each phase of the compiler.
2. Illustrate the lexical analyser and its tools to recognize the tokens.
3. Construct the top-down parsers using context free grammar.
4. Differentiate top down and bottom-up parsers with its implementation.
5. Write the SDD and intermediate code for back end of the compiler.
6. Use various optimization methods to generate the target code effectively.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Artificial Intelligence & Machine Learning

(Computer Science and Engineering)

III B.Tech - II Sem **SRIT R20**

Course Code	Category	Hours/Week			Credits	Maximum Marks		
		L	T	P	C	CIA	SEE	Total
R204GA05601	PCC	3	0	0	3	40	60	100

Objectives

- Introduce and define the meaning of Intelligence and explore various paradigms for knowledge encoding in computer systems.
- To understand the concepts of state space representation, exhaustive search, heuristic search together with the time and space complexities.
- Introduce subfields of AI such as NLP, GamePlaying, Bayesian Models, etc.
- Introduce the concept of learning patterns from data and develop a strong theoretical foundation for understanding state of the art Machine Learning algorithms.

Unit I – Introduction

AI Fundamentals: Defining Artificial Intelligence, Defining AI techniques, State Space Search and Heuristic Search Techniques, Defining problems as State Space search, Production systems and characteristics, Hill Climbing, Breadth first and depth first search, Best First search, Constraint Satisfaction.

Learning Outcomes:

At the end of this unit, the students will be able to

1. Recognize the importance of Artificial Intelligence.
2. Identify how intelligent agent is related to its environment.
3. Know various uninformed and informed search strategies.

Unit II - Knowledge Representation

Representations and Mappings, Approaches to knowledge representation, Using Predicate Logic and Representing Knowledge as Rules, Representing simple facts in logic, Procedural vs Declarative knowledge, Logic Programming, Forward vs backward reasoning, Resolution.

Learning Outcomes:

At the end of this unit, the students will be able to

1. Demonstrate the importance the constraint satisfaction problems.
2. Explain effective usage of propositional logic in knowledge representation.
3. Learn how resolution works.

Unit III – Representation of Uncertain knowledge

Symbolic Logic under Uncertainty, Probability and Bayes' Theorem, Certainty factors, Probabilistic Graphical Models, Bayesian Networks, Markov Networks, Fuzzy Logic.

Learning Outcomes:

At the end of this unit, the students will be able to

1. Knowing the importance of Bayes' theorem and its applications
2. Illustrate about Bayesian Belief Networks

Unit IV - Fundamentals of Machine Learning

Introduction: Idea of Machines learning from data, Classification of problem – Regression and Classification, Supervised, Unsupervised learning, Reinforcement Learning

Linear Regression: Model representation for single variable, single variable Cost Function, Gradient Decent for Linear Regression, Multivariable model representation, Multivariable cost function, Gradient Decent in practice, Normal Equation and non-invertibility.

Learning Outcomes:

At the end of this unit, the students will be able to

1. Explain different types of learning

2. Explore various statistical learning mechanisms

Unit V – Introduction to Neural Networks

Neural Networks: Non-linear Hypothesis, Biological Neurons, Model representation, Intuition for Neural Networks, Multiclass classification, Cost Function, Back Propagation Algorithm, Back Propagation Intuition, Weights initialization, Neural Network Training.

Learning Outcomes:

At the end of this unit, the students will be able to

1. Appraise artificial neural networks as one of the most effective learning methods currently known to interpret complex real-world sensor data.
2. Illustrate the principles of Probability for classification as an important area of Machine Learning Algorithms.

Text Books:

1. Artificial Intelligence: A Modern Approach, Stuart Russel, Peter Norvig.
2. Machine Learning, Tom M. Mitchell.

Reference Books:

1. Artificial Intelligence, 2nd Edition, Rich and Knight.
2. Building Machine Learning Systems with Python, Richert & Coelho.
3. Artificial Intelligence, Shivani Goel, Pearson Education.

Course Outcomes:

At the end of the course, the students will be

1. Describe the building blocks of artificial intelligence and machine learning.
2. Implement search algorithm for a problem and estimate its time and space complexities.
3. Use different knowledge representation for solving complex tasks.
4. Solve problems involving uncertain information using probabilistic techniques.
5. Implement machine learning techniques which are suitable for a given problem.
6. Demonstrate various types of neural networks and their cost functions.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Android Application Development

(Computer Science and Engineering)

III B.Tech-II Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA05602	**PCC**	3	0	0	3	40	60	100

Objectives

- To understand fundamentals of android operating systems.
- Illustrate the various components, layouts and views in creating android applications
- To understand advance concepts of android programming.
- To understand the fundamentals of Kotlin programming in android.

Unit I – Introduction to Android

Introduction: Android Studio Installation and Configuration of SDK and JDK, Basic Building blocks – Activities, Services, Broadcast Receivers & Content providers, UI Components -Views & notifications, Components for communication -Intents & Intent Filters, Android API levels.
Basics of Android: AndroidManifest.xml, Uses-permission & uses-sdk, Resources & R.java, Assets, Layouts & Drawable Resources, Activities and Activity lifecycle, First sample Application.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Illustrate the importance of Building blocks of android.
2. Explore fundamentals of UI components.
3. Demonstrate the installations of Android Studio.
4. Explore design and implementation of the app according to Activity life cycle.

Unit II – UI Design

Basic UI Design: Form widgets, Text Fields, Layouts – (dip, dp, sip, sp) versus px, Shared Preferences, Preferences from xml, Menus, Intents, Fragments, Time and Date, Images and media, Composite, Alert Dialogs & Toast Messages, Popup.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Design using Form widgets and Menus.
2. Demonstrate various Layouts in UI Design.
3. Create an app using various intents and Fragments.
4. Illustrate the Alert Dialogs & Toast, popup.

Unit III – Database Connectivity and Adapters

Overview of Content Providers: SQLite Programming, SQLiteOpenHelper, SQLiteDatabse, Cursor, Reading and updating.
Linkify: Web URLs, Email address, text, map address, phone numbers, MatchFilter & TransformFilter.
Adapters and Widgets: Introduction to Adapters and its types ListView - ListActivity, Custom listview, GridView using adapters, Gallery using adapters.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Explain the concept of SQLite Database and its operations.
2. Implement cursors for content provider through SQLite.
3. Demonstrate the various Adapters.
4. Demonstrate the significance of Widgets with various views.

Unit IV – Services & Broadcast Receivers

Notifications: Broadcast Receivers, Services and notifications, Alarms.
Custom Components: Custom Tabs, Custom animated popup panels, other components.
Services: Overview of services in Android, implementing a Service, Service lifecycle, Inter Process Communication (AIDL Services).

Learning Outcomes:
At the end of this unit, the student will be able to

1. Demonstrate the Broadcast Receivers Services and notifications.
2. Demonstrate the various custom components.
3. Explain need of Services in Android.
4. Implement the services using service life cycle.

Unit V – Application Development Using Kotlin

Introduction Kotlin, intelli-j, Features, Structure of Kotlin Programming. Convert Main Activity to kotlin Code. Operators, Data types, Variables, Conditionals, Lists and arrays, Null safety. Functions, Classes & Objects, Extensions.
First App: Your first app, Anatomy of an Android app, Layouts and resources in Android Activities, Make an app interactive, Gradle: Building an Android app in kotlin.

Learning Outcomes:
At the end of this unit, the student will be able to

1. Explain the fundamentals of Kotlin Programming.
2. Explain the connectivity of IntelliJ-j with kotlin.
3. Illustrate the sunctions, classes and Extensions in kotlin.
4. Create an android app using Kotlin programming language and Gradle building.

Text Books:

1. Android programming by B.M Halwani, Pearson Education, 2013.
2. Kotlin in Aciton by Dmitry Jemerov and Svetlana Isakova, Manning Publications, 2017.

Reference Books:

1. Android application Development for Java Programmers, James C Sheusi, Cengage Learning.
2. Learn Kotlin for Android development, Peter spath, Leipzig- Apress, -2019.
3. Kotlin for Android Developers by Antonio Leiva, 2017.

Course Outcomes:

At the end of the course, student will be able to

1. Describe data sharing with different applications and sending and intercepting Messages. Broad casting.
2. Know the building blocks of an android application.
3. llustrate the advancement in android app development.
4. Design the application with database connectivity using modern tools.
5. Design an application using broadcast services.
6. Develop an application using kotlin programming in android.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Introduction to Big Data

(Professional Elective-II)

III B.Tech - II Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA05603	**PEC**	3	0	0	3	40	60	100

Objectives:

- To Understand Big Data Analysis for comparing other systems like Hadoop.
- To gain knowledge about working on Hadoop File System and Learn the design of Hadoop File system.
- To learn for Data processing use MapReduce Job.
- To learn how to analyze Big Data using different Tools like HiveQl.
- To understand the importance of Big data in comparison with traditional databases.

Unit I – Introduction

Introduction to Big Data, A brief history of Hadoop, Hadoop: Data storage and Analysis, Comparison with other systems. Grid computing, Distributed File systems leading to Hadoop file system.
Hadoop File System: Introduction, Hadoop Architecture, Internals of Hadoop File Systems. Apache Hadoop and the Hadoop Ecosystem. Linux refresher, VM Ware Installation of Hadoop.

Learning Outcomes:

At the end of this unit, the students will be able to

1. Understand the fundamental concepts of big data.
2. Explore about Hadoop and its components.
3. Demonstrate the installation of VM Ware.

Unit II –Hadoop File System

The Design of HDFS, HDFS Concepts, Command line interface to HDFS, Hadoop file systems. Interfaces, Java interface to Hadoop. Anatomy of a file read. Anatomy of a file writes. Replica placement and coherency Model. Parallel Copying with distribution, Keeping of HDFS Cluster balanced.

Learning Outcomes:

At the end of this unit, the students will be able to

1. Know about architecture and commands in HDFS.
2. Explore HDFS as balanced cluster.
3. Understand about Java interface to Hadoop.

Unit III –MapReduce

Introduction: Analyzing Data with unix tools, analyzing data with Hadoop, Java MapReduce classes (new API) Data flow, Combiner functions, Running a distributed MapReduce job, Job Configuration API. Setting up the development environment, Managing configuration, Writing a unit test with MR Unit, Running a job in local job runner, Running on a cluster, Launching a job, The MapReduce Web UI.

Learning Outcomes:

At the end of this unit, the students will be able to

1. Analyze the data Hadoop.
2. Explore the importance of MapReduce.
3. Implement unit test with MapReduce unit.

Unit IV - Classic MapReduce

Classic MapReduce, Job submission, Job Initialization, Task Assignment, Task Execution, Progress and Status Updates, Job Completion, Shuffle and Sort on Map and Reducer side, Configuration Tuning, MapReduce Types, Input Formats.,Output formats, Sorting, Map side and Reduce side joins. Running simple word count Map-Reduce program on the cluster, Additional examples of M-R Programming.

Learning Outcomes:

At the end of this unit, the students will be able to

1. Analyze how the job can Assigned and executed in MapReduce.
2. Understand the I/O formats of MapReduce.
3. Implement the MapReduce Program to running simple word count problem.

Unit V – Hive Shell

The Hive Shell, Hive Services, Hive Clients, The Meta Store, Comparison with Traditional Data bases, Hive Ql, HBasics- Concepts. Implementation. Java and Map reduce clients, Loading data, web queries. Big Data analytics using Map-Reduce programming: K-Means clustering, using Big Data analytics libraries using Mahout.

Learning Outcomes:

At the end of this unit, the students will be able to

1. Understand the Hive Concepts.
2. Implementation of Hive Clents through Java and MapReduce.
3. Understand K-means clustering using Big Data Mahout libraries.

Text Books:

1.Tom White, Hadoop, "The Definitive Guide" 3rd Edition, O'Reilly Publications, 2012.
2. Hadoop in Action by Chuk Lam, Manning Publications.

Reference Books:

1. Network Data Analytics, A Hands-On Approach for Application Development.
2. Authors: Srinivasa, K.G., G M, Siddesh, H., Srinidhi.Big Data, Black Book: Covers Hadoop 2, MapReduce, Hive, YARN, Pig, R and Data Visualization Paperback – 1 January 2016.
3. Big Data Analytics, Introduction to Hadoop, Spark, and Machine-Learning Paperback – 16 February 2019 by Raj Kamal (Author), PreetiSaxena (Author).

Course Outcomes:

At the end of the course, the students will be to

1. Know the Bigdata Concepts using Hadoop.
2. Learn the design issues of Hadoop File system.
3. Demonstrate the Anatomy of Hadoop File System.
4. Describe about Hadoop MapReduce job Initialization and Execution.
5. Implement MapReduce program on the cluster.
6. Understand HiveShell services and HiveQl Concepts.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Cyber Security

(Professional Elective-II)

III B.Tech -II Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA05604	PEC	3	0	0	3	40	60	100

Objectives:

- Understand essential building blocks and basic concepts of cyber security.
- Describe various methods and tools sued in cybercrime.
- Describe the legal issues and ethics in Cyber security.
- Understand the tools used in cyber crime
- Understand the need of computer forensic.

Unit I – INTRODUCTION

Cybercrime: Mobile and Wireless devices-Trend mobility-authentication service security-Attacks on mobile phones-mobile phone security Implications for organizations- Organizational measurement for Handling mobile-Security policies and measures in mobile computing era. Cases.

Learning Outcomes:

After completing this Unit, students will be able to

1. Understand various Security policies and measures.
2. Explain different types of attacks on mobile.

Unit II – Tools and methods used in cyber crime

Proxy servers and Anonymizers- Phishing Password cracking-Key loggers and Spy wares-Virus and worms-Trojan Horse and Backdoors-Steganography-SQL Injection-Buffer overflow-Attacks on wireless network. Cases.

Learning Outcomes:

After completing this Unit, students will be able to

1. Outline the attacks on browser, Web and email.
2. Explain the security aspects of Operating Systems.

Unit III – computer forensic

Understanding computer forensic-Historical background of cyber forensic, Forensic analysis of e-mail-Digital forensic life cycle-Network forensic-Setting up a computer forensic Laboratory-Relevance of the OSI 7 Layer model to computer Forensic- Computer forensic from compliance perspectives. Cases.

Learning Outcomes:

After completing this Unit, students will be able to

1. Identify the network security threats and attacks.
2. Design the Counter measures to defend the network security attacks.
3. Analyze the security tools and techniques for Cloud computing

Unit IV – Forensic of Hand –Held Devices

Understanding cell phone working characteristics- Hand-Held devices and digital forensic-Toolkits for Hand-Held device-Forensic of i-pod and digital music devices-Techno legal Challenges with evidence from hand-held Devices. Cases.

Learning Outcomes:

After completing this Unit, students will be able to

1. Interpret the need for Privacy and its impacts of Emerging Technologies.
2. Explain how to handle incidents and deal with Disaster.

Unit V – cybercrimes and IPR issues

Cyber Security –Organizational implications-cost of cybercrimes and IPR issues Web threats for organizations: the evils and Perils-Social media marketing Security and privacy Implications-Protecting people privacy in the organization's Forensic best practices for

organizations. Cases.

Learning Outcomes:

After completing this Unit, students will be able to

1. Adapt legal issues and ethics in computer security.
2. Elaborate on the Emerging topics.

Text Books:

1. Nina Godbole, Sunit Belapure, "Cyber Security", Wiley India, 2012.
2. Harish Chander**,** "cyber laws & IT protection", PHI learning pvt. ltd, 2012.

Reference Books:

1. Dhiren R Patel, "Information security theory & practice", ,PHI learning pvt ltd,2010.
2. MS.M.K.Geetha & Ms.Swapne Rama,"Cyber Crimes and Fraud Management",MACMILLAN,2012. Pankaj Agarwal : Information Security& Cyber Laws (Acme Learning), Excel, 2013.
3. Vivek Sood, Cyber Law Simplified, TMH, 2012.

Course Outcomes:

At the end of the course, student will be able to

1. Classify various Cyber Crimes in Mobile and Wireless devices.
2. Assess cyber security risk management policies in order to protect an organization's critical information.
3. Describe tools and methods used in cybercrimes.
4. Understand the fundamentals of Computer Forensic.
5. Use computer forensics tools in data analysis to identify cyber-crime.
6. Understand the risk remediation strategies in Cyber Security.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Software Project Management

(Professional Elective-II)

III B.Tech - II Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA05605	**PEC**	3	0	0	3	40	60	100

Objectives

- To describe and determine the purpose and importance of project management from the perspectives of planning, tracking and completion of project.
- To compare and differentiate organization structures and project structures.
- To implement a project to manage project schedule, expenses and resources with the application of suitable project management tools.

Unit I – Conventional Software Management

The waterfall model, conventional software management performance, Evolution of Software Economics, Improving Software Economics, Automation, Achieving required quality, Peer inspections.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Describe the evolution of software economics.
2. Analyze various models to automate the software development process.

Unit II – The Old Way and The New

The principles of conventional software Engineering, principles of modern software management, transitioning to an iterative process. Life Cycle Phases, Artifacts of The Process- The artifact sets, Management artifacts, Engineering artifacts, programmatic artifacts.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Categorize different life cycle phases.
2. Analyze engineering and production stages.
3. Describe various artifact sets.

Unit III – Model Based Software Architectures

A Management perspective and technical perspective, Work Flows of the Process, Checkpoints of the Process- Major mile stones, Minor Milestones, Periodic status assessments.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Describe various workflows.
2. Summarize the check points of the process.
3. Illustrate periodic status assessments of the project.

Unit IV – Iterative Process Planning

Work breakdown structures, planning guidelines, cost and schedule estimating, Iteration planning process, Pragmatic planning, Project Organizations and Responsibilities.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Develop the WBS structure of any project.
2. Understand the responsibilities of Organizations.
3. Describe the evolution of organization.

Unit V – Process Automation

Automation Building blocks, The Project Environment, Project Control and Process Instrumentation, Project Estimation and Management: COCOMO model, Critical Path Analysis,

PERT technique, Monte Carlo approach.

Learning Outcomes: At the end of this unit, the student will be able to

1. Identify seven core metrics.
2. Analyze the cost estimation using PERT and CPM techniques.

Text Books:

1. Software Project Management, Walker Royce, Pearson Education, 2005.
2. Software Project Management, Bob Hughes, 4th edition, Mike Cotterell, TMH.

Reference Books:

1. Software Project Management, Joel Henry, Pearson Education.
2. Software Project Management in practice, Pankaj Jalote, Pearson Education, 2005.
3. Effective Software Project Management, Robert K.Wysocki, Wiley,2006

Course Outcomes:

At the end of the course, student will be able to

1. Describe the fundamental concepts of planning and organization of software project.
2. Illustrate the management of conventional software.
3. Demonstrate old and new trends of software engineering artifacts.
4. Explain management and technical perspectives of model-based software architecture.
5. Acquire the knowledge of iterative planning process for cost and schedule estimation.
6. Exemplify various process automation tools.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Compiler Design Lab

(Computer Science and Engineering)

III B.Tech - II Semester **SRIT R20**

Course Code	Category	Hours/Week			Credits	Maximum Marks		
		L	T	P	C	CIA	SEE	Total
R204GA05608	PCC	0	0	3	1.5	40	60	100

Objective:

- To understand the implementation of Lexical Analyzer, parser and compiler design aspects.
- To write codes for various top down and bottom up parsers and verify them for correctness.
- To understand LEX and YACC tools.

List of Experiments

1)Design a lexical analyzer for a given language and the lexical analyzer should ignore redundant spaces, tabs and new lines. It should also ignore comments. Although the syntax specification states that identifiers can be arbitrarily long, you may restrict the length to some reasonable value. Simulate the same in C language.
2)Write a C program to identify whether a given line is a comment or not.
3)Write a C program to recognize strings under 'a', 'a*b+', 'abb'.
4)Write a C program to test whether a given identifier is valid or not.
5)Write a C program to simulate lexical analyzer for validating operators.
6)Implement the lexical analyzer using JLex, flex or other lexical analyzer generating tools.
7) Write a C program for implementing the functionalities of predictive parser for the mini language specified in Note 1.
8) a) *Write a C program for constructing of LL (1) parsing.
b) *Write a C program for constructing recursive descent parsing.
9) Write a C program to implement LALR parsing.
10) a) Write a C program to implement operator precedence parsing.
b) Write a C program to implement Program semantic rules to calculate the expression that takes an expression with digits, + and * and computes the value

Note 1:

A simple language written in this language is

```
{int a[3],t1,t2;
T1=2;
A[0]=1;a[1]=2;a[t]=3;
T2=-( a[2]+t1*6)/(a[2]-t1);
If t2>5then
Print(t2)
Else{
Int t3;
T3=99;
T2=25;
Print(-t1+t2*t3);/*this is a comment on 2 lines*/
}endif
}
```

Comments(zero or more characters enclosed between the standard C/JAVA Style comment brackets/*…*/)can be inserted .The language has rudimentary support for1-dimenstional array,the declaration int a[3] declares an array of three elements,referenced as a[0],a[1] and a[2].Note also you should worry about the scooping of names.

Reference Books:

1. Modern Compiler Implementation in C- Andrew N. Appel, Cambridge University Press.
2. Modern Compiler Design- Dick Grune, Henry E. Bal, Cariel T. H. Jacobs, Wiley dreamtech

Course Outcomes:

At the end of the lab, student will be able to

1. Develop working functionality of analysis phase of the compiler.
2. Implement Lexical analyser for given language using C and LEX tools.
3. Implement recursive and predictive parser for a given grammar.
4. Implement Operator precedence parser.
5. Implement LL(1), LALR parser for the given grammar.
6. Implement Program semantic rules to evaluate the expression.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Artificial Intelligence & Machine Learning Lab

(Computer Science and Engineering)

III B.Tech - II Sem								SRITR20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA05609	**PCC**	0	0	3	1.5	40	60	100

Objectives:

This course will enable students to

- Demonstrate the uninformed and informed AI search algorithms
- Make use of Data sets in implementing the machine learning algorithms.
- Implement the machine learning concepts and algorithms.
- The programs can be implemented in Python.
- Data sets can be taken from standard repositories. (https://archive.ics.uci.edu/ml/datasets.html) or (www. kaggle.com) or constructed by the students.

Lab Experiments:

1. Write a Program to Implement Breadth First Search using Python.
2. Write a Program to Implement Tic-Tac-Toe game using Python
3. Write a Program to Implement 8-Puzzle problem using Python
4. Write a Program to Implement Water-Jug problem using Python
5. Write a Program to Implement Travelling Salesman Problem using Python
6. Write a Program to Implement Missionaries-Cannibals Problems using Python
7. Implement and demonstrate the FIND-S algorithm for finding the most specific hypothesis based on a given set of training data samples. Read the training data from a .CSV file.
8. For a given set of training data examples stored in a .CSV file, implement and demonstrate the Candidate-Elimination algorithm to output a description of the set of all hypotheses consistent with the training examples.
9. Write a program to demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.
10. Build an Artificial Neural Network by implementing the Back Propagation algorithm and test the same using appropriate data sets.
11. Write a program to implement the naïve Bayesian classifier for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering few test data sets
12. Assuming a set of documents that need to be classified, use the naïve Bayesian Classifier model to perform this task. Built-in API can be used to write the program. Calculate the accuracy, precision, and recall for your data set.
13. Write a program to construct a Bayesian network considering medical data. Use this model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set. You can use Python ML library classes.
14. Write a program to implement k-Nearest Neighbour algorithm to classify the iris data set. Print both correct and wrong predictions. Python ML library classes can be used for this problem.

Reference Books:

1. Stephen Marsland, "Machine Learning -An Algorithmic Perspective", Second Edition, Chapman and Hall/CRC Machine Learning and Pattern Recognition Series, 2014.
2. Andreas C. Müller and Sarah Guido "Introduction to Machine Learning with Python: A Guide for Data Scientists", Oreilly.
3. Shai Shalev-Shwartz , Shai Ben-David, "Understanding Machine Learning: From Theory to Algorithms" , Cambridge University Press.

Course Outcomes:

At the end of the course, student will be able to

1. Implement uninformed search algorithms.
2. Implement heuristics search algorithms.

3. Implement procedures for the machine learning algorithms.
4. Write Python programs for various Machine Learning algorithms.
5. Choose different data sets for the implementation of Machine Learning algorithms.
6. Identify and apply Machine Learning algorithms to solve real world problems.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Android Application Development Lab

(Computer Science and Engineering)

III B.Tech - II Semester **SRIT R20**

Course Code	Category	Hours/Week			Credits	Maximum Marks		
		L	T	P	C	CIA	SEE	Total
R204GA05610	PCC	0	0	3	1.5	40	60	100

Objective:

- To develop android application using modern technologies.
- To implement the various components, layouts and views in creating android applications.
- Design the App with database connectivity using modern tools.

List of Experiments

1) Installation of Android studio, Kotlin- IntelliJ.
2) Create an application for various choosing concepts.
   a. checkbox b. Radio button c. Radio group d. Spinner
3) Creating the application by using the Activity and Fragment Life cycle.
4) Design the Application using Intents.
5) Create an application by using fragments.
6) Design an application for List View.
7) Create an application to play the audio and video clips.
8) Design the Application for Designing Image gallery.
9) Design the application for Menus and Action Bars.
10) Create an application for Login form with SQLite Database Connectivity.
11) Create an application for web view.
12) Design an application with Date and Time Pickers.
13) Create an application on services (Background Task).
14) Design the Layouts for an activity using Kotlin.
15) Design Calculator app using kotlin programming.

Reference Books:

3. "Dawn Griffiths, David Griffiths" Head First Android Development: A Brain-Friendly Guide, O'Reilly Media.
4. "Dawn Griffiths, David Griffiths" Head First Kotlin: A Brain-Friendly Guide, O'Reilly Media

Course Outcomes:

At the end of the course, student will be able to

1. To Create application data sharing with different concepts of sending and intercepting Messages.
2. Develop an application using services and publishing android applications.
3. To Illustrate the advancement in android app development.
4. To develop applications using IntelliJ and android studio.
5. Create an android app using java programming.
6. Create an android app using Kotlin programming.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Life Sciences for Engineers

(Common to all Branches)

III B.Tech - II Semester								SRIT R20
Course Code	Category	Hours/Week			Credits	Maximum Marks		
R204GA5MC04	NCMC	L	T	P	C	CIA	SEE	Total
		2	0	0	0	40	-	40

Objectives

- Introduce the molecular basis of life.
- Provide the basis for classification of living organisms.
- Describe the transfer of genetic information.
- Introduce the techniques used for modification of living organisms.
- Describe the applications of biomaterials.

Unit I-INTRODUCTION TO LIFE SCIENCES & LIVING ORGANISMS

Why we need to study Life Sciences? Comparison and differences of biological organisms with manmade systems (Eye & Camera, Bird flying & Aircraft), Biological observations of 18th Century that led to major discoveries. Classification of living organisms, Cellular basis of life, differences between prokaryotes and eukaryotes, classification on the basis of carbon and energy sources.

Learning Outcomes:

After completing this unit, the student will be able to

1. Summarize the basis of life.
2. Explain the differences between biological organisms and manmade systems.

Unit II-BIO-MOLECULES & MACROMOLECULES

Molecules of life: Water, Sugars, Starch, Cellulose, Amino acids, Structure and functions of proteins (primary, secondary, tertiary and quaternary structure), Structure and functions of nucleotides, nucleic acids, DNA (single and double strand) & RNA, hemoglobin, antibodies and enzymes, Industrial applications of enzymes and Fermentation process.

Learning Outcomes:

After completing this unit, the student will be able to

1. Explain the relationship between the structure and function of proteins.
2. Interpret the relationship between the structure and function of nucleic acids.

Unit III-HUMAN PHYSIOLOGY

Bioenergetics, Respiration: Glycolysis and TCA cycle, Electron transport chain and oxidative phosphorylation, Mechanism of photosynthesis, Human physiology, Neurons, Synaptic and Neuromuscular junctions.

Learning Outcomes:

After completing this unit, the student will be able to

1. Apply thermodynamic principles to biological systems.
2. Explain the mechanism of respiration and photosynthesis.
3. Summarize the principles of information transfer and processing in humans.

Unit IV-GENES, DNA & RNA

Mendel's laws, gene mapping, Mitosis and Meiosis, single gene disorders in humans, Genetic code, DNA replication, Transcription, Translation. Discuss the concept of complementation using human genetics. Recombinant DNA Technology: recombinant vaccines, transgenic microbes, plants and animals, animal cloning, biosensors, biochips.

Learning Outcomes:

After completing this unit, the student will be able to

1. Differentiate mitosis and meiosis.
2. Explain the medical importance of gene disorders.
3. Identify DNA as a genetic material in the molecular basis of information transfer.

Unit V-METABOLISM

Thermodynamics as applied to biological systems. Exothermic and endothermic versus endergonic and exergonic reactions. Concept of Keq and its relation to standard free energy. ATP as an energy currency. This should include the breakdown of glucose to CO_2 + H_2O (Glycolysis and Krebs cycle) and synthesis of glucose from CO_2 and H_2O (Photosynthesis).

Learning Outcomes:

After completing this unit, the student will be able to

1. Outline the principles of recombinant DNA technology.
2. Identify the potential of recombinant DNA technology.
3. Summarize the use of biological materials for diagnostic devices.

Text Books:

1. Biology: A global approach, N. A. Campbell, J. B. Reece, L. Urry, M. L. Cain and S. A. Wasserman, Pearson Education Ltd, 2018.
2. Biology for Engineers, Arthur T Johnson, CRC press, 2011.
3. Cell and Molecular Biology, De Robetis and De Robertis.

Reference Books:

1. The molecular biology of the cell, Alberts, Garland Science, 6th edition 2014.
2. Outlines of Biochemistry, E. E. Conn, P. K. Stumpf, G. Bruening and R. H. Doi, John Wiley and Sons, 2009.
3. Introduction to Biomedical Engineering, John Enderle and Joseph Bronzino, 3rd edition, 2012.

Course Outcomes:

At the end of the course, student will be able to

1. Explain catalytic properties of enzymes.
2. Summarize application of enzymes and fermentation in industry.
3. Identify DNA as a genetic material in the molecular basis of information transfer.
4. Apply thermodynamic principles to biological systems.
5. Analyze biological processes at the reductionist level.
6. Identify the potential of recombinant DNA technology.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Data Analytics

(Professional Elective-III)

IV B.Tech – I Sem								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA05701	**PEC**	3	0	0	3	40	60	100

Objectives:

- Big exposed to big data.
- Learn the different ways of data analysis.
- Be familiar with data streams.
- Learn the mining and clustering.
- Be familiar with the visualization.

Unit I – Introduction to Big Data

Introduction to Big Data Platform: Challenges of conventional systems, Web data, Evolution of Analytic scalability, analytic processes and tools, Modern data analytic tools, Statistical concepts.

Learning Outcomes:

At the end of this unit, the student will be able to:

1. State the challenges in conventional systems with data.
2. Differentiate between analysis and reporting.
3. Understand analytic processing tools.
4. Use statistical concepts in data analytics.

Unit II – Data Analysis

Regression modeling, Multivariate analysis, Bayesian modeling, inference and Bayesian networks, Support vector and kernel methods, Analysis of time series, Neural networks: learning and generalization, competitive learning, principal component analysis, Fuzzy logic: extracting fuzzy models from data, fuzzy decision trees, Stochastic search methods.

Learning Outcomes:

At the end of this unit, the student will be able to:

1. Understand regression modeling and analysis of time series.
2. Understand neural networks and fuzzy logic models and search methods.

Unit III – Mining Data Streams

Introduction to Streams Concepts, Stream data model and architecture, Stream Computing, Sampling data in a stream, filtering streams, counting distinct elements in a stream, estimating moments, counting oneness in a window, Decaying window, Real-time Analytics Platform (RTAP) applications: sentiment analysis, stock market predictions.

Learning Outcomes:

At the end of this unit, the student will be able to:

1. Describe the concept of streams and its distinct element.
2. Understand real time analytics platform applications.

Unit IV – Frequent Item sets and Clustering

Mining Frequent item sets, Market based model, Apriori Algorithm, handling large data sets in Main memory, Limited Pass algorithm, counting frequent item sets in a stream, Clustering Techniques: Hierarchical, K- Means, Clustering high dimensional data, Clustering for streams and Parallelism.

Learning Outcomes:

At the end of this unit, the student will be able to:

1. Identify the frequent item sets.
2. Perform clustering on high dimensional data using clustering methods.

Unit V – Frameworks and Visualization

MapReduce, Hadoop, Hive, MapR, Sharding, NoSQL Databases, S3, Hadoop Distributed file systems, Visualizations, Visual data analysis techniques.

Learning Outcomes:
At the end of this unit, the student will be able to:

1. Understand the file system of Hadoop.
2. Describe the data using visualization techniques.

Text Books

1. Michael Berthold, David J. Hand, Intelligent Data Analysis, Springer, 2007.
2. Anand Rajaraman and Jeffrey David Ullman, Mining of Massive Datasets Cambridge University Press, 2012.

Reference Books:

1. Bill Franks, Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with advanced analystics, John Wiley & sons, 2012.
2. Glenn J. Myatt, Making Sense of Data, John Wiley & Sons, 2007 Pete Warden, Big Data Glossary, O" Reilly, 2011.
3. Jiawei Han, Micheline Kamber "Data Mining Concepts and Techniques", Second Edition, Elsevier, Reprinted 2008.

Course Outcomes:

At the end of the course, student will be able to

1. Apply the statistical analysis methods.
2. Compare and contrast various soft computing frameworks.
3. Design distributed file systems.
4. Apply Stream data model.
5. Use Visualization techniques
6. Understand the importance of various clustering techniques.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Mobile Computing

(Professional Elective-III)

IV B.Tech- I Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA05702	**PEC**	3	0	0	3	40	60	100

Objectives

- Understand basic concepts of Mobile Computing, Mobile Tele communications, and Mobile Network Infrastructure.
- Understand the issues and solutions of MAC layer, Network Layer, and Transport Layer of Mobile Networks.
- Understand data delivery models and database issues in mobile environment.
- Understand Ad-hoc Networks and its related concepts.
- Understand Protocols and Platforms used in Mobile environments.

Unit I – Introduction

Mobile Communications, Mobile Computing – Paradigm, Promises/Novel Applications and Impediments and Architecture; Mobile and Handheld Devices, Limitations of Mobile and Handheld Devices. Radio Interfaces, Protocols, Localization, Calling, Handover, Security.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Familiar the basic concepts principles of mobile computing, Mobile Computing Architecture, and determine the applications and limitations of Mobile devices.
2. Get a working understanding of the applications, characteristics and limitations of mobile hardware devices including their user-interface modalities.

Unit II – (Wireless) Medium Access Control (MAC)

(Wireless) Medium Access Control (MAC): Motivation for a specialized MAC (Hidden and exposed terminals, Near and far terminals), SDMA, FDMA, TDMA, CDMA, Wireless LAN/ (IEEE 802.11).

Learning Outcomes:

At the end of this unit, the student will be able to

1. Determine the functionality of wireless MAC for a mobile network.
2. Explain about various multiple access methods.
3. Understand fundamentals of wireless Networks.

Unit III – Mobile Network Layer

IP and Mobile IP Network Layers, Packet Delivery and Handover Management, Location Management, Registration, Tunnelling and Encapsulation, Route Optimization, DHCP.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Understand IP, Mobile IP and DHCP that provide the ability of a host to stay connected regardless of their location.

Unit IV – Mobile Transport Layer

Conventional TCP/IP Protocols, Indirect TCP, Snooping TCP, Mobile TCP, Other Transport Layer Protocols for Mobile Networks.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Illustrate the fundamentals of TCP protocols.

2. Apply the mechanism of to send packet streams from various applications simultaneously over a network.

Unit V – Data Dissemination and Synchronization

Communications Asymmetry, Classification of Data Delivery Mechanisms, Data Dissemination, Broadcast Models, Selective Tuning and Indexing Methods, Data Synchronization – Introduction, Software, and Protocols.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Describe the timer mechanism, flow control, congestion control of data.
2. Learning knowledge on event, timer, and message processing.

Text Books:

1. Jochen Schiller, "Mobile Communications", Addison-Wesley, Second Edition, 2009.
2. Raj Kamal, "Mobile Computing", Oxford University Press,2007, ISBN: 0195686772.

Reference Books:

1. Jochen Schiller, "Mobile Communications", Addison-Wesley, Second Edition, 2004.
2. Stojmenovic and Cacute, "Handbook of Wireless Networks and Mobile Computing", Wiley, 2002, ISBN 0471419028.
3. Reza Behravanfar, "Mobile Computing Principles: Designing and Developing.

Course Outcomes:

At the end of the course, student will be able to

1. Able to think and develop new mobile application.
2. Able to debate on any new technical issue related to this new paradigm and come up with a solution(s).
3. Apply knowledge of TCP/IP extensions for mobile networking.
4. Able to develop new ad hoc network applications and/or algorithms/protocols.
5. Able to explain & develop any existing or new protocol related to mobile environment.
6. Analyze security, energy efficiency, mobility, scalability, and their unique characteristics in mobile networks.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Software Requirements & Estimation

(Professional Elective-III)

IV B.Tech – I Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA05703	**PEC**	3	0	0	3	40	60	100

Objectives

- Understand the good practices for requirements engineering.
- Understand Requirements elicitation, elicitation techniques.
- Acquire the knowledge on analysis models, Software quality attributes.
- Understand software estimation, size estimation.
- Understand Effort, Schedule and Cost Estimation.

Unit I – Software Requirements

What and Why: Essential Software requirement, good practices for requirements engineering, Improving requirements processes, Software requirements and risk management.

Software Requirements Engineering: Requirements elicitation, requirements analysis documentation, review, elicitation techniques, analysis models, Software quality attributes, risk reduction through prototyping, setting requirements priorities, verifying requirements quality.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Identify the essential software requirements of a project.
2. Understand the good practices for requirements gathering.
3. Illustrate requirements elicitation techniques.
4. Understand the requirements elicitation techniques.

Unit II – Software Requirements Management & Modeling

Software Requirements Management: Requirements management Principles and practices, Requirements attributes, Requirements Traceability Matrix, Links in requirements chain.

Software Requirements Modeling: Use Case Modeling, Analysis Models, Dataflow diagram, state transition diagram, class diagrams, Object analysis.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Implement good practices of requirements management.
2. Understand the change management Process.
3. Analyze the components of requirements chain.
4. Implement the requirements using use case modeling diagrams.

Unit III – Software Estimation

Components of Software Estimations, Estimation methods, Problems associated with estimation, Key project factors that influence estimation.

Size Estimation: Two views of sizing, Function Point Analysis, Mark II FPA, Full Function Points, LOC Estimation, Conversion between size measures.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Explore the software estimation methods.
2. Analyze the factors that influences the estimation.
3. Illustrate the size estimation of a software project.

Unit IV – Effort, Schedule and Cost Estimation

Effort, Schedule and Cost Estimation: What is Productivity? Estimation Factors, Approaches to Effort and Schedule Estimation, COCOMO II, Putnam Estimation Model, Algorithmic models, Cost Estimation.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Illustrate various types of effort and cost estimation models.
2. Analyze the cost estimation using algorithmic models.
3. Understand the COCOMO II model to estimate the cost, effort and schedule to develop new software.

Unit V – Tools for Requirements Management and Estimation Requirements Management

Benefits of using a requirements management tool, commercial requirements management tool, Rational Requisite pro, Caliber – RM, implementing requirements management automation.

Software Estimation Tools: Desirable features in software estimation tools, IFPUG, USC's COCOMO II, SLIM (Software Life Cycle Management) Tools.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Explore the benefits of requirements management tools.
2. Demonstrate various types of Symmetric-key Algorithms.
3. Analyze the process automation for requirements management.
4. Illustrate tools for software estimation.

Text Books:

1. Swapna Kishore, Rajesh Naik, Software Requirements and Estimation, 1st Edition, Tata Mc Graw Hill, 2001.
2. "Data Communications and Networking", Behrouz A. Forouzon, Sophia Chung Fegan, McGraw Hill Higher Education Fifth Edition, 2012.

Reference Books:

1. Karl E. Weigers, Software Requirements, 2nd Edition, Microsoft Press, 2003.
2. Estimating Software Costs, Second edition, Capers Jones, TMH, 2007.
3. Managing Software Requirements, Dean Leffingwell & Don Widrig, Pearson Education, 2003.

Course Outcomes:

At the end of the course, student will be able to

1. Identify the minimum requirements for the development of application.
2. Analyze requirement elicitation techniques and prototyping.
3. Gain knowledge about requirement management, their principles and practices.
4. Analyze use case modeling and different data diagrams.
5. Estimating the software in terms of size, cost, effort and schedule.
6. Analyze the requirements management process automation.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Artificial Neural Networks

(Professional Elective-IV)

IV B.Tech – I Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA05704	**PEC**	3	0	0	3	40	60	100

Objectives

- To introduce the fundamental techniques and principles of Neural Networks.
- To study the different models in ANN and their applications.
- To familiarize deep learning concepts with Convolutional Neural Network case studies.

Unit I – Introduction to Neural Networks

Introduction, Humans and Computers, Organization of the Brain, Biological and Artificial Neuron Models, Characteristics of ANN, McCulloch-Pitts Model, Potential Applications of ANN.

Learning Outcomes:

At the end of this unit, the student will be able to:

1. Understand the how humans and computers are different in thinking.
2. Understand the organization of brain neuron models.
3. Understand ANN and its applications.

Unit II – Essentials of Artificial Neural Networks

Artificial Neuron Model, Operations of Artificial Neuron, Types of Neuron Activation Function, ANN Architectures, Classification Taxonomy of ANN, Learning Strategy (Supervised, Unsupervised, Reinforcement), Learning Rules.

Learning Outcomes:

At the end of this unit, the student will be able to:

1. Understand ANN, its operations and architecture.
2. Understand classifications of ANN.

Unit III – Single Layer Feed Forward Networks

Introduction, Perceptron Models: Discrete, Continuous and Multi-Category;
Training Algorithms: Discrete and Continuous Perceptron Networks, Limitations of the Perceptron Model.

Learning Outcomes:

At the end of this unit, the student will be able to:

1. Study the importance of perceptron models and its limitations.

Unit IV – Multi-Layer Feed Forward Networks

Credit Assignment Problem, Generalized Delta Rule, Derivation of Backpropagation (BP) Training, Summary of BP Algorithm, Kolmogorov Theorem, Learning Difficulties and Improvements.

Learning Outcomes:

At the end of this unit, the student will be able to:

1. Understand the importance of back propagation in training.
2. State the learning difficulties and improvements in multi-layer feed forward networks.

Unit V – Associative Memories

Pattern Mathematics, Hebbian Learning, Concepts of Associative Memory, Bidirectional Associative Memory (BAM) Architecture, BAM Training
Algorithms: Storage and Recall Algorithm, BAM Energy Function. Neural network applications: Process identification, control, fault diagnosis.

Learning Outcomes:

At the end of this unit, the student will be able to:

1. Understand the concepts of associative memory.
2. Discuss the associative memory algorithms.

Text Books:

1. Laurene Fausett, "Fundamentals of Neural Networks", Pearson Education, 2004.
2. Simon Haykin, "Neural Networks- A comprehensive foundation", Pearson Education, 2003.

Reference Books:

1. S.N.Sivanandam, S.Sumathi,S. N. Deepa "Introduction to Neural Networks using MATLAB 6.0", TATA Mc Graw Hill, 2006.
2. S. Rajasekharan and G. A. Vijayalakshmi pai, "Neural Networks, Fuzzy logic, Genetic algorithms: synthesis and applications", PHI Publication, 2004.
3. Timothy J. Ross, " Fuzzy Logic with Engineering Applications", Tata McGraw-Hill Inc. 2000.

Course Outcomes:

At the end of the course, student will be able to

1. Explain the basic concepts in Neural Networks and applications.
2. Identify the deep learning algorithms which are more appropriate for various types of learning tasks.
3. Discuss feed forward networks and their training issues.
4. Distinguish different types of ANN architectures.
5. Explain the deep learning concepts using Back Propagation Neural Network.
6. Implement neural network algorithms to solve real-world problems.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Internet of Things and Its Applications

(Professional Elective-IV)

IV B.Tech – I Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA05705	**PEC**	3	0	0	3	40	60	100

Objectives

- Understand IoT Access technologies.
- Study the IoT protocols.
- Design and Development of IoT based applications.
- Basics of IoT Data Analytics and supporting services.
- Study about various IoT case studies and industrial applications.

Unit I – Fundamentals of IoT

Evolution of Internet of Things, Enabling Technologies, M2M Communication, IoT World Forum (IoTWF) standardized architecture, Simplified IoT Architecture, Core IoT Functional Stack, Fog, Edge and Cloud in IoT, Functional blocks of an IoT ecosystem, Sensors, Actuators, Smart Objects and Connecting Smart Objects

Learning Outcomes:

At the end of this unit, the student will be able to

1. Understand IoT architecture.
2. Discuss functional blocks of an IoT ecosystem.

Unit II – IoT Protocols

IoT Access Technologies: Physical and MAC layers, topology and Security of IEEE 802.15.4, 802.11ah and Lora WAN, Network Layer: IP versions, Constrained Nodes and Constrained Networks,6LoWPAN, Application Transport Methods: SCADA, Application Layer Protocols: CoAP and MQTT.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Understand various Network layer IoT Protocols.
2. Understand Application layer and Transport layer protocols.

Unit III – Design and Development

Design Methodology, Embedded computing logic, Microcontroller, System on Chips, IoT system building blocks.

Learning Outcomes:

At the end of this unit, the student will be able to:

1. Study the importance of Embedded computing logic in IoT system design.

Unit IV – Data Analytics and Supporting Services

Data Analytics: Introduction, Structured Versus Unstructured Data, Data in Motion versus Data at Rest, IoT Data Analytics Challenges, Data Acquiring, Organizing in IoT/M2M.

Supporting Services: Computing Using a Cloud Platform for IoT/M2M Applications/Services, Everything as a service and Cloud Service Models.

Learning Outcomes:

At the end of this unit, the student will be able to:

1. Understand the importance Data analytics with IoT.
2. State the difficulties and improvements in Data Analytics.

Unit V – Case Studies/Industrial Applications

IoT applications in home, infrastructures, buildings, security, Industries, Home appliances, other IoT electronic equipment's, Industry 4.0 concepts.

Learning Outcomes:

At the end of this unit, the student will be able to:

1. Understand the implementation of various IoT applications.
2. Discuss the Industry 4.0 concepts.

Text Books:

1. IoT Fundamentals: Networking Technologies, Protocols and Use Cases for Internet of Things, David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Rob Barton and Jerome Henry, Cisco Press, 2017.
2. Internet of Things – A hands-on approach, Arshdeep Bahga, Vijay Madisetti, Universities Press, 2015.

Reference Books:

1. The Internet of Things – Key applications and Protocols, Olivier Hersent, David Boswarthick, Omar Elloumi and Wiley, 2012 (for Unit2).
2. "From Machine-to-Machine to the Internet of Things – Introduction to a New Age of Intelligence",Jan Ho¨ ller, VlasiosTsiatsis, Catherine Mulligan, Stamatis, Karnouskos, Stefan Avesand. David Boyle and Elsevier, 2014.
3. Architecting the Internet of Things,Dieter Uckelmann, Mark Harrison, Michahelles and Florian (Eds), Springer,2011.

Course Outcomes:

At the end of the course, student will be able to

1. Understand the basics of IoT.
2. Illustrate the functional stack of IoT.
3. Implement the state of the Architecture of an IoT.
4. Understand design methodology and hardware platforms involved in IoT.
5. Understand how to analyze and organize the data.
6. Compare IOT Applications in Industrial & real world.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Software Quality Assurance

(Professional Elective-IV)

IV B.Tech - I Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA05706	**PEC**	3	0	0	3	40	60	100

Objectives

- Understand the basic tenets of software quality and quality factors.
- Be exposed to the Software Quality Assurance (SQA) architecture and the details of SQA components.
- Understand of how the SQA components can be integrated into the project life cycle.
- Be familiar with the software quality infrastructure.
- Be exposed to the management components of software quality.

Unit I – Introduction to Software Quality & Architecture

Need for Software quality, Quality challenges, Software Quality Assurance (SQA) – Definition and objectives, Software quality factors, McCall's quality model, SQA system and architecture, Software Project life cycle Components, Pre project quality components, Development and quality plans.

Learning Outcomes:

At the end of this unit, the student will be able to:

1. Understand the need of software quality assurance.
2. Analyze SQA system.
3. Illustrate architecture of SQA.
4. Analyze the usage of SQA in Software Project life cycle.

Unit II – SQA Components and Project Life Cycle

Software Development methodologies, Quality assurance activities in the development process-Verification & Validation, Reviews, Software Testing, Software Testing implementations, Quality of software maintenance, Pre-Maintenance of software quality components, Quality assurance tools, CASE tools for software quality, Software maintenance quality, Project Management.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Understand the process of Quality assurance in the software development.
2. Implement the software testing process.
3. Analyze the quality assurance tools.
4. Implement the SQA in the Project.

Unit III – Software Quality Infrastructure

Procedures and work instructions, Templates, Checklists, 3S development - Staff training and certification Corrective and preventive actions, Configuration management, Software change control, Configuration management audit, Documentation control, Storage and retrieval.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Implement Procedures and work instructions in SQA.
2. Determine the training and updating methods for staff.
3. Implement corrective and preventive actions process to develop the solution.
4. Understand the issues of controlled document approval, storage and retrieval.

Unit IV – Software Quality Management & Metrics

Project process control, Computerized tools, Software quality metrics – Objectives of quality measurement, Process metrics, Product metrics – Implementation – Limitations of software metrics, Cost of software quality – Classical quality cost model – Extended model – Application of Cost model.

Learning Outcomes:
At the end of this unit, the student will be able to

1. Implement of software quality metrics.
2. Understand the classic model and extended model of cost of software quality.
3. Analyze the application of a cost of software quality system

Unit V – Standards, Certifications & Assessments

Quality management standards – ISO 9001 and ISO 9000-3, capability Maturity Models – CMM and CMMI assessment methodologies, Bootstrap methodology, SPICE Project, SQA project process standards – IEEE std 1012 & 1028, Organization of Quality Assurance – Department management responsibilities, Project management responsibilities, SQA units and other actors in SQA systems.

Learning Outcomes: At the end of this unit, the student will be able to

1. Understand the scope of quality management standards.
2. Analyze CMM and CMMI assessment methodology.
3. Understand the SQA project process standards.
4. Analyze Project management responsibilities for quality assurance.

Text Books:

1. Daniel Galin, "Software Quality Assurance", Pearson Publication, 2009.
2. Schulmeyer, G.G., McManus, J.I., Handbook of Software Quality Assurance. Prentice Hall 1999.

Reference Books:

1. Alan C. Gillies, "Software Quality: Theory and Management", International Thomson Computer Press.1997.
2. Mordechai Ben-Menachem "Software Quality: Producing Practical Consistent Software", International Thompson Computer Press, 1997.

Course Outcomes:

At the end of the course, student will be able to

1. Utilize the concepts in software development life cycle.
2. Demonstrate their capability to adopt quality standards.
3. Assess the quality of software product.
4. Apply the concepts in preparing the quality plan & documents.
5. Efficiently perform Quality Assurance activities using modern software tools.
6. Effectively manage a Testing and Quality Assurance project.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Deep Learning

(Professional Elective-V)

IV B.Tech – I Semester **SRIT R20**

Course Code	Category	Hours/Week			Credits	Maximum Marks		
		L	T	P	C	CIA	SEE	Total
R204GA32702	PEC	3	0	0	3	40	60	100

Objectives

- To acquire the knowledge on Deep Learning Concepts
- To learn various types of unsupervised deep learning models.
- To gain knowledge on regularization techniques for deep learning.
- To learn optimization strategies for large scale applications
- To Acquire knowledge on applications of computer vision and Natural Language Processing.

Unit I – Introduction to Deep Learning

Deep Feed forward networks, Gradient-Based learning, Hidden Units, Architecture Design, Back-Propagation, Differentiation Algorithms, ReLu Heuristics for avoiding bad local minima. Heuristics for faster training. Nestors accelerated gradient descent. Regularization. Dropout.

Learning outcomes:

At the end of this unit, the student will be able to

1. Familiarize with feed forward networks
2. Differentiate between sigmoid and ReLu functions

Unit II - Deep Un-Supervised Learning

Unsupervised Learning Network- Introduction, Fixed Weight Competitive Nets, Magnet, Hamming Network, Kohonen Self-Organizing Feature Maps, Learning Vector Quantization, Counter Propagation Networks, Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN).

Learning outcomes:

At the end of this unit, the student will be able to

1. Know different types of neural networks
2. Understand the features of unsupervised learning networks

Unit III - Regularization for Deep Learning

Regularization and Under-Constrained Problems, Dataset Augmentation, Noise Robustness, Semi-Supervised learning, multi-task learning, Early Stopping, Sparse Representations, Bagging and other Ensemble Methods, Dropout, Tangent Classifier.

Learning outcomes:

At the end of this unit, the student will be able to

1. Understand the concept of data augmentation.
2. Know over fitting problem in decision trees.
3. Distinguish the terms like bagging, boosting and ensembling.

Unit IV - Optimization to train Deep Models

Challenges in Neural Network Optimization, Basic Algorithms, Parameter Initialization Strategies, Algorithms with Adaptive Learning Rates, Approximate Second-Order Methods, Optimization Strategies, Meta-Algorithms.

Learning outcomes:

At the end of this unit, the student will be able to

1. Know about optimization of neural network
2. Understand different optimization techniques

Unit V - Applications

Large-Scale Deep Learning, Image segmentation, Object detection, automatic image captioning, Image generation with Generative adversarial networks, video to text with LSTM models. Neural Summarization, Opinion Mining using Recurrent Neural Networks.

Learning outcomes:

At the end of this unit, the student will be able to

1. Classify image segmentation
2. Understand about LSTM Models

Text Books:

1. Ian Good fellow and Yoshua Bengio and Aaron Courville, "Deep Learning", An MIT Press Book, 2015.
2. Bengio, Yoshua. "Learning deep architectures for AI." Foundations and trends in Machine Learning 2.1 (2009).

Reference Books:

1. Simon Haying, "Neural Networks and Learning Machines" 3rd Edition, Pearson Prentice Hall.
2. "Neural Networks and Deep Learning: A Text Book", Charu C. Aggarwal, Springer Edition, September 2018.
3. "Deep Learning in Python / Pytorch ", Francois Chollet, Manning Publications

Course Outcomes:

At the end of the course, student will be able to

1. Understand the concepts of Neural Networks and Deep learning.
2. Choose the appropriate learning Networks in modeling.
3. Use an efficient regularization algorithm for Deep Learning Models
4. Apply optimization strategies for large scale applications.
5. Develop deep learning applications to Computer vision and NLP.
6. Understand about object detection and image classification.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Blockchain Fundamentals

(Professional Elective-V)

IV B.Tech – I Semester **SRIT R20**

Course Code	Category	Hours/Week			Credits	Maximum Marks		
		L	T	P	C	CIA	SEE	Total
R204GA05707	PEC	3	0	0	3	40	60	100

Objectives

- To provide conceptual understanding of the function of Block chain as a method of securing distributed ledgers, how consensus on their contents is achieved, and the new applications thatthey enable.
- Impart strong technical understanding of Blockchain technologies.
- To cover the technological underpinnings of block chain operations as distributed data structures and decision-making systems, their functionality and different architecture types.
- To provide a critical evaluation of existing "smart contract" capabilities and platforms, andexamine their future directions, opportunities, risks and challenges
- Develop familiarity of current technologies, tools, and implementation strategies.

Unit I – Introduction

Blockchain Introduction, Concepts of Blockchain Systems, Key Problem Challenges and Solutions, Bitcoin Concept, Merkle Tree, hardness of mining, Transaction Verifiability, anonymity, Attacks on Bitcoin, Double-spend attacks, Selfish mining, Security of Transactions in Bitcoin, Privacy in Bitcoin.

Learning Outcomes:

At the end of this unit, the student will be able to
Understand the structure of a block chain.

Unit II – Cryptographic Primitives in Blockchain

Cryptosystems in practice, Cryptographic Hash Functions, Digital Signatures-Aggregate Signature,Threshold Signature.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Explain cryptographic building blocks and reason about their security.
2. Exploit applications of Blockchain in real world sceneries.

Unit III - Blockchain Platforms

Blockchain-Ethereum, Smart Contracts - Attacks on smart contracts, Permissioned Blockchain —Hyperledger, Blockchain Applications & Use Cases.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Develop Blockchain based solutions and write smart contract using Hyperledger Fabric andEthereum frameworks.
2. Integrate ideas from various domains and implement them using Blockchain technology.

Unit IV - Consensus Protocols

The consensus problem- Byzantine Generals problem, Asynchronous Byzantine Agreement, Consensus mechanisms used in Bitcoin Blockchain, Ethereum Blockchain and Hyperledger Blockchain.

Learning Outcomes:

At the end of this unit, the student will be able to

1. To understand the Consensus Protocols.

2. To Apply Consensus Protocols using Blockchain.

Unit V- Blockchain (BoT)

Advantages of integrating Blockchain to IoT, Trust Building, Cost Reduction, Accelerate Data Exchanges, Scaled Security for IoT.

Learning Outcomes:

At the end of this unit, the student will be able to

1. Understand the integration of Blockchain with IoT.
2. Understand the security concepts in BoT.

Text Books:

1. Pethuru Raj, Kavita Saini, Chellammal Surianarayanan, "Blockchain Technology and Applications", CRC Press, 2021.
2. Chandramouli Subramanian, "Blockchain Technology", Universities Press, 2020.

Reference Books:

1. Beginning Blockchain a Beginner's Guide to Building Blockchain Solutions, Bikramaditya Singhal, Gautam Dhameja, Priyansu Sekhar Panda, Apress, 2018.
2. Mastering Ethereum, Andreas M. Antonopoulos Dr. Gavin Wood, O'Reilly First 2018.
3. Blockchain Enabled Applications, Vikram Dhillon David Metcalf Max Hooper, Apress 2017.

Course Outcomes:

At the end of the course, student will be able to

1. Know the basics and advantages of Block Chain Technology.
2. Describe Security of transactions and Privacy in Bitcoin.
3. Illustrate Cryptographic primitives in Block Chain Technology.
4. Describe working of Block Chain including Private and Public platforms.
5. Use Consensus Protocols for Block Chain Technology.
6. Discuss the advantages of integrating Block Chain Technology to IoT.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

Agile Methodologies

(Professional Elective-V)

IV B.Tech - I Semester								SRIT R20
Course Code	**Category**	**Hours/Week**			**Credits**	**Maximum Marks**		
		L	**T**	**P**	**C**	**CIA**	**SEE**	**Total**
R204GA05708	**PEC**	3	0	0	3	40	60	100

Objectives

- To understand the agile concept and its importance in software development.
- Understand how an iterative, incremental development process leads to faster delivery of more useful software.
- To acquire complete knowledge on Xtreme programming.
- To know complete modeling of agile process on XP environment.

Unit I –Fundamentals of Agile

Agile Software Development: Traditional Model vs. Agile Model, Classification of Agile Methods, Agile Manifesto and Principles, Agile project management, Design and development practices in Agile projects.

Learning outcomes:

At the end of the unit, students will be able to:

1. Classify different agile methods for software development
2. Describe the origins and motivations of the Agile Manifesto
3. Construct different agile models

Unit II –Agile Frameworks

Lean Production, SCRUM, Crystal, Feature Driven Development, Adaptive Software Development, Extreme Programming: Method Overview, Lifecycle, Work Products, Roles and Practices.

Learning outcomes:

At the end of the unit, students will be able to:

1. Analyze what scrum methodology is
2. Justify extreme programming
3. Distinguish agile modelling and planning XP projects

Unit III –Agility and Knowledge Management

Agile Information Systems – Agile Decision Making –– Institutional Knowledge Evolution Cycle – Development, Acquisition, Refinement, Distribution, Deployment, Leveraging – KM in Software Engineering – Managing Software Knowledge – Challenges of Migrating to Agile Methodologies – Agile Knowledge Sharing.

Learning outcomes:

At the end of the unit, students will be able to:

1. Define Knowledge management cycle
2. Demonstrate incremental software development
3. Explain the importance of agile methodologies

Unit IV –Agility and Requirements Engineering

Impact of Agile Processes in RE, Current Agile Practices, Variance, Overview of RE Using Agile Managing Unstable Requirements, Requirements Elicitation, Agile Requirements . Abstraction Model: Requirements Management & Prioritization in Agile Environment, Agile Requirements Modeling and Generation, Concurrency in Agile Requirements Generation.

Learning outcomes:

At the end of the unit, students will be able to:

1. Identify tools to help with agile development

2. Define the requirements for agile modelling

Unit V –Agility and Quality Assurance

Developing Incremental Requirements, Agile Approach to Quality Assurance, Customer Tests, Test- Driven Development, Incremental Design and Architecture, Spike Solutions, Performance Optimization.

Learning outcomes:

At the end of the unit, students will be able to:

1. Outline the practices that keep the code clean and allow the entire team to contribute to development.
2. Implement testing activities within an Agile project using various strategies.
3. Able to design the architecture as incremental approach.

Text Books:

1. David J. Anderson and Eli Schragenheim, —Agile Management for Software Engineering: Applying the Theory of Constraints for Business Results, Prentice Hall, 2003.
2. James Shore and Shane Warden, "The Art of Agile Development", O'REILLY, 2007.
3. Hazza and Dubinsky, "Agile Software Engineering, Series: Undergraduate Topics in Computer Science", Springer, 2009.

Reference Books:

1. Robert C. Martin, "Agile Software Development, Principles, Patterns, and Practices" , PHI, 2002.
2. Bhuvan Unhelkar, "The Art of Agile Practice: A Composite Approach for Projects and Organizations", CRC Press.
3. Craig Larman, —Agile and Iterative Development: A Managers Guide, Addison-Wesley, 2004

Course Outcomes:

At the end of the course, student will be able to

1. Use agile methods in various development environments.
2. Design and model agile methods in SCRUM.
3. Create own agile method by customizing to a particular situation.
4. Develop techniques and tools for improving team collaboration and software.
5. Apply regaining control techniques.
6. Quality, show how agile approaches can be scaled up to the enterprise level.

SRINIVASA RAMANUJAN INSTITUTE OF TECHNOLOGY

SRIT R20

OPEN ELECTIVES offered by CSE Department

BACHELOR OF TECHNOLOGY
in
COMPUTER SCIENCE & ENGINEERING

Open Elective-I:: R20

Course Code	Course Name	Subject Area	Periods perweek			Credits	Scheme of Examination Max. Marks		
			L	T	P		CIA	SEE	Total
R204GA05507	Essentials of Python Programming	OEC	3	0	0	3	40	60	100
R204GA05508	Computer Organization & Operating System	OEC	3	0	0	3	40	60	100

Open Elective-II :: R20

Course Code	Course Name	Subject Area	Periods perweek			Credits	Scheme of Examination Max. Marks		
			L	T	P		CIA	SEE	Total
R204GA05606	Mean Stack Technology	OEC	3	0	0	3	40	60	100
R204GA05607	Introduction to Artificial Intelligence	OEC	3	0	0	3	40	60	100

Open Elective- III :: R20

Course Code	Course Name	Subject Area	Periods perweek			Credits	Scheme of Examination Max. Marks		
			L	T	P		CIA	SEE	Total
R204GA05709	Data Science	OEC	3	0	0	3	40	60	100
R204GA05710	Fundamentals of Security in Computing	OEC	3	0	0	3	40	60	100

Open Elective-IV :: R20

Course Code	Course Name	Subject Area	Periods perweek			Credits	Scheme of Examination Max. Marks		
			L	T	P		CIA	SEE	Total
R204GA05711	Virtualization and Cloud Computing	OEC	3	0	0	3	40	60	100
R204GA05712	Blockchain Technology and Applications	OEC	3	0	0	3	40	60	100
R204GA05715	Programming In Java	OEC	3	0	0	3	40	60	100