Study Scheme & Syllabus
of

Bachelor of Technology
Computer Engineering

B. Tech (CoE)

Batch 2019 onwards

By Department of Academics

IK Gujral Punjab Technical University

Bachelor of Technology in Computer Engineering

It is a Graduate (UG) Programme of 4 years duration (8 semesters)

Courses & Examination Scheme:

First Semester

Course Code	Course Type	Course Title	Load Allocations			Marks Distribution		Total Marks	Credits
			L	T	P	Internal	External		
BTPH104-18	Basic Science Course	Semiconductor Physics	3	1	0	40	60	100	4
BTPH114-18	Basic Science Course	Semiconductor Physics (Lab)	0	0	3	30	20	50	1.5
BTAM104-18	Basic Science Course	Math-1	3*	1	0	40	60	100	4
BTEE101-18	Engineering Science Course	Basic Electrical Engineering	3	1	0	40	60	100	4
BTEE102-18	Engineering Science Course	Basic Electrical Engineering (Lab)	0	0	2	30	20	50	1
BTME101-18	Engineering Science Course	Engineering Graphics & Design	1	0	4	60	40	100	3
BMPD101-18		Mentoring and Professional Development	0	0	2	Satisfactory / Un-Satisfactory			Non Credit
Total			**12**	**2**	**15**	**290**	**360**	**650**	**20.5**

*These are the minimum contact hrs. allocated . The contact hrs. may be increased by institute as per the need based on the content of subject.

Second Semester

Course Code	Course Type	Course Title	Load Allocations			Marks Distribution		Total Marks	Credits
			L	T	P	Internal	External		
BTCH101-18	Basic Science Course	Chemistry-I	3	1	0	40	60	100	4
BTCH102-18	Basic Science Course	Chemistry-I (Lab)	0	0	3	30	20	50	1.5
BTA204-18	Basic Science Course	Maths-II	3*	1	0	40	60	100	4
BTPS101-18	Engineering Science Course	Programming for Problem Solving	3	0	0	40	60	100	3
BTPS102-18	Engineering Science Course	Programming for Problem Solving (Lab)	0	0	4	30	20	50	2
BTMP101- 18	Engineering Science Course	Workshop / Manufacturing Practices	1	0	4	60	40	100	3
BTHU101-18	Humanities and Social Sciences including Management courses	English	2	0	0	40	60	100	2
BTHU102-18	Humanities and Social Sciences including Management courses	English (Lab)	0	0	2	30	20	50	1
BMPD201-18		Mentoring and Professional Development	0	0	2	Satisfactory / Un-Satisfactory			Non-Credit
Total			**12**	**2**	**15**	**290**	**360**	**650**	**20.5**

*These are the minimum contact hrs. allocated. The contact hrs. may be increased by institute as per the need based on the content of subject.

Third Semester

Course Code	Course Type	Course Title	Load Allocations			Marks Distribution		Total Marks	Credits
			L	T	P	Internal	External		
BTES 301-18	Engineering Science Course	Digital Electronics	3	0	0	40	60	100	3
BTCS 301-18	Professional Core Courses	Data structure & Algorithms	3	1	0	40	60	100	3
BTCS 302-18	Professional Core Courses	Object Oriented Programming	3	0	0	40	60	100	3
BTAM304-18	Basic Science Course	Mathematics-III	4	1	0	40	60	100	3
HSMC 101/102-18	Humanities & Social Sciences Including Management \Courses	Foundation Course in Humanities (Development of Societies/Philosophy)	2	1	0	40	60	100	3
BTES302-18	Engineering Science Course	Digital Electronics Lab	0	0	2	30	20	50	1
BTCS303-18	Professional Core Courses	Data structure & Algorithms Lab	0	0	4	30	20	50	2
BTCS304-18	Professional Core Courses	Object Oriented Programming lab.	0	0	4	30	20	50	2
BTCS305-18	Professional Core Courses	IT Workshop*	0	0	2	30	20	50	1
		Summer Institutional Training	0	0	0	60	40	100	Satisfactory/Un satisfactory
Total			**15**	**3**	**12**	**380**	**420**	**800**	**21**

*Syllabus to be decided by respective institute internally. It may include latest technologies.

Fourth Semester

Course Code	Course Type	Course Title	Load Allocations			Marks Distribution		Total Marks	Credits
			L	T	P	Internal	External		
BTCS 401-18	Professional Core Courses	Discrete Mathematics	3	1	0	40	60	100	4
BTES 401-18	Engineering Science Course	Computer Organization & Architecture	3	1	0	40	60	100	3
BTCS 402-18	Professional Core Courses	Operating Systems	3	1	0	40	60	100	3
BTCS 403-18	Professional Core Courses	Design & Analysis of Algorithms	3	1	0	40	60	100	3
HSMC 122-18	Humanities & Social Sciences including Management Courses	Universal Human Values-II	2	1	0	40	60	100	3
EVS101- 18	Mandatory Courses	Environmental Sciences	1	-	-	-	-	-	0
BTES 402-18	Engineering Science Course	Computer Organization & Architecture Lab	0	0	2	30	20	50	1
BTCS 404-18	Professional Core Courses	Operating Systems Lab	0	0	4	30	20	50	2
BTCS 405-18	Professional Core Courses	Design & Analysis of Algorithms Lab	0	0	4	30	20	50	2
Total			**15**	**5**	**10**	**290**	**360**	**650**	**21**

There will be 4-6 weeks summer industrial training after 4th sem.

Fifth Semester

Course Code	Type of Course	Course Title	Hours per Week			Marks Distribution		Total Marks	Credits
			L	T	P	Internal	External		
BTES 501-18	Engineering Science	Enterprise Resource Planning	3	0	0	40	60	100	3
BTCS 501-18	Professional Core Courses	Database Management Systems	3	0	0	40	60	100	3
BTCS 502-18	Professional Core Courses	Formal Language & Automata Theory	3	0	0	40	60	100	3
BTCS 503-18	Professional Core Courses	Software Engineering	3	0	0	40	60	100	3
BTCS 504-18	Professional Core Courses	Computer Networks	3	0	0	40	60	100	3
BTCS XXX-18	Professional Elective	Elective-I	3	0	0	40	60	100	3
MC	Mandatory Courses	Constitution of India/ Essence of Indian Traditional Knowledge	2	-	-	100	-	100	S/US
BTCS 505-18	Professional Core Courses	Database Management Systems Lab	0	0	4	30	20	50	2
BTCS 506-18	Professional Core Courses	Software Engineering Lab	0	0	2	30	20	50	1
BTCS 507-18	Professional Core Courses	Computer Networks Lab	0	0	2	30	20	50	1
BTCS XXX-18	Professional Elective	Elective-I Lab	0	0	2	30	20	50	1
	Professional Training	Industrial *Training	-	-	-	60	40	100	S/US
Total			**20**	**0**	**10**	**460**	**440**	**900**	**23**

* 4-6 weeks industrial training undertaken after 4th semester in summer vacations.

Sixth Semester

Course Code	Type of Course	Course Title	L	T	P	Internal	External	Total Marks	Credits
BTCS 601-18	Professional Core Courses	Compiler Design	3	0	0	40	60	100	3
BTCS 602-18	Professional Core Courses	Artificial Intelligence	3	0	0	40	60	100	3
BTCS UUU-18	Professional Elective Courses	Elective-II	3	0	0	40	60	100	3
BTCS YYY-18	Professional Elective Courses	Elective-III	3	0	0	40	60	100	3
BTOE ***	Open Elective Courses	Open Elective-I	3	0	0	40	60	100	3
BTCS 603-18	Project	Project-1	0	0	6	60	40	100	3
BTCS 604-18	Professional Core Courses	Compiler Design Lab	0	0	2	30	20	50	1
BTCS 605-18	Professional Core Courses	Artificial Intelligence Lab	0	0	2	30	20	50	1
BTCS UUU-18	Professional Elective Courses	Elective-II lab	0	0	2	30	20	50	1
BTCS YYY-18	Professional Elective Courses	Elective-III lab	0	0	2	30	20	50	1
Total			**15**	**0**	**14**	**380**	**420**	**800**	**22**

Seventh Semester

Course Code	Type of Course	Course Title	Hours per Week			Marks Distribution		Total Marks	Credits
			L	T	P	Internal	External		
BTCS 701-18	Professional Core Courses	Network Security and Cryptography	3	0	0	40	60	100	3
BTCS 702-18	Professional Core Courses	Data Mining and Data Warehousing	3	0	0	40	60	100	3
BTOE ***	Open Elective Courses	Open Elective-II	3	0	0	40	60	100	3
BTCS ZZZ-18	Professional Elective	Elective- IV	3	0	0	40	60	100	3
BTCS TTT-18	Professional Elective Courses	Elective-V	3	0	0	40	60	100	3
BTCS 703-18	Project	Project-II	0	0	12	120	80	200	6
BTCS ZZZ-18	Professional Elective	Elective- IV lab	0	0	2	30	20	50	1
BTCS TTT-18	Professional Elective	Elective- V lab	0	0	2	30	20	50	1
Total			**15**	**0**	**14**	**380**	**420**	**800**	**23**

Eighth Semester

Course Code	Course Title	Marks Distribution		Total Marks	Credits
		Internal	External		
BTCS 801-18	Semester Training	300	200	500	16

LIST OF ELECTIVES

Elective-I

BTCOE 508-18	Microprocessor & Assembly Language Programming
BTCOE 509-18	Computer Peripheral & Interface
BTCOE 510-18	Graph Theory
BTCOE 511-18	Microprocessor & Assembly Language Programming lab
BTCOE 512-18	Computer Peripheral & Interface lab
BTCOE 513-18	Graph Theory Lab
BTCS 510-18	Programming in Python
BTCS 513-18	Programming in Python Lab

Elective- II

BTCS714-18	Parallel Computing
BTCS715-18	Parallel Computing Lab
BTCOE 601	Advance Computer Architecture
BTCOE 602	Advance Computer Architecture Lab
BTCOE 603	Real Time Embedded Systems
BTCOE 604	Real Time Embedded Systems Lab
BTCS608-18	Internet of Things
BTCS609-18	Internet of Things Lab

Elective- III

BTCS 616-18	Data Science
BTCS 617-18	Data Science Lab
BTCOE 605	Wireless Sensor Network
BTCOE 606	Wireless Sensor Network Lab
BTCOE 607	Hardware Security
BTCOE 608	Hardware Security Lab
BTCOE 609	Human Computer Interaction
BTCOE 610	Human Computer Interaction Lab

BTCS ZZZ: Elective-IV

BTCS618-18	Machine learning
BTCS 619-18	Machine Learning lab
BTCS 620-18	Mobile Application Development
BTCS 621-18	Mobile Application Development lab
BTCS 704-18	Deep Learning
BTCS 705-18	Deep Learning Lab
BTCS 712-18	Blockchain Technologies
BTCS 713-18	Blockchain Technologies Lab

BTCS TTT: Elective-V

BTCS 612-18	Cloud computing
BTCS 613-18	Cloud computing lab
BTCS 706-18	Distributed databases
BTCS 707-18	Distributed databases lab
BTCOE 701-19	Big Data Analytics
BTCOE 702-19	Big Data Analytics lab
BTCOE 703-19	Software Defined Networks
BTCOE 704-19	Software Defined Networks lab

L IST OF COURSES FOR HONOURS DEGREE

In order to have an Honours degree, a student choose 18-20 credits from the following courses in addition.

Course Code	Type of Course	Course Title	Hours per Week			Marks Distribution		Total Marks	Credits
			L	T	P	Internal	External		
BTCS H01-18	Professional Elective Courses	Graph Theory	3	0	0	40	60	100	3
BTCS H02-18	Professional Elective Courses	Computer Vision	3	0	0	40	60	100	3
BTCS 611-18	Professional Elective Courses	Embedded Syst ems	3	0	0	40	60	100	3
BTCS H03-18	Professional Elective Courses	Software Project Management	3	0	0	40	60	100	3
BTCS H04-18	Professional Elective Courses	Cryptography & Network Security	3	0	0	40	60	100	3
BTCS H05-18	Professional Elective Courses	Internet-of-Things	3	0	0	40	60	100	3
BTCS 804-18	Professional Elective Courses	Data Analytics	3	0	0	40	60	100	3
BTCS 608-18	Professional Elective Courses	Machine Learning	3	0	0	40	60	100	3
BTCS H06-18	Professional Elective Courses	ICT in Agriculture and Rural Development	3	0	0	40	60	100	3
BTCS H07-18	Professional Elective Courses	Computational Technologies for Smart Cities	3	0	0	40	60	100	3
BTCS H08-18	Professional Elective Courses	Computer Forensics	3	0	0	40	60	100	3

First Semester

BTPH104-18	Semiconductor Physics	L-3, T-1, P-0	Credits – 4
Prerequisite (if any): Introduction to Quantum Mechanics desirable			
Course Objectives: The aim and objective of the course on Semiconductor Physics is to introduce the students of B. Tech. class to the formal structure of semiconductor physics so that they can use these in Engineering as per their requirement.			
Course Outcomes: At the end of the course, the student will be able to			
CO1	Understand and explain the fundamental principles and properties of electronic materials and Semiconductors		
CO2	Understand and describe the interaction of light with semiconductors in terms of fermi golden rule.		
CO3	Understand and describe the impact of solid-state device capabilities and limitations on electronic circuit performance		
CO4	Understand the design, fabrication, and characterization techniques of Engineered semiconductor materials		
CO5	Develop the basic tools with which they can study and test the newly developed devices and other semiconductor applications.		

Detailed Syllabus:

PART-A

UNIT 1: Electronic materials (10 lectures)

Free electron theory of metals, Density of states in 1D, 2D, and 3D, Bloch's theorem for particles in a periodic potential, Energy band diagrams, Kronig-Penny model (to introduce origin of band gap), Energy bands in solids, E-k diagram, Direct and indirect bandgaps, Types of electronic materials: metals, semiconductors, and insulators, Occupation probability, Fermi level, Effective mass.

UNIT II: Semiconductors (10 lectures)

Intrinsic and extrinsic semiconductors, Dependence of Fermi level on carrier-concentration and temperature (equilibrium carrier statistics), Carrier generation and recombination, Carrier transport: diffusion and drift, p-n junction, Metal-semiconductor junction (Ohmic and Schottky), Semiconductor materials of interest for optoelectronic devices.

PART-B

UNIT III: Light-semiconductor interaction (10 lectures)

Optical transitions in bulk semiconductors: absorption, spontaneous emission, and stimulated emission; Einstein coefficients, Population inversion, application in semiconductor Lasers; Joint density of states, Density of states for phonons, Transition rates (Fermi's golden rule), Optical loss and gain; Photovoltaic effect, Exciton, Drude model.

UNIT IV: Measurement Techniques (10 lectures)

Measurement for divergence and wavelength using a semiconductor laser, Measurements for carrier density, resistivity, hall mobility using Four-point probe and van der Pauw method, Hot-point probe measurement, capacitance-voltage measurements, parameter extraction from diode I-V characteristics.

Reference books and suggested reading:
1. J. Singh: Semiconductor Optoelectronics: Physics and Technology, McGraw-Hill Inc. (1995).
2. B. E. A. Saleh and M. C. Teich: Fundamentals of Photonics, John Wiley & Sons, Inc., (2007).
3. S. M. Sze: Semiconductor Devices: Physics and Technology, Wiley (2008).
4. A. Yariv and P. Yeh, Photonics: Optical Electronics in Modern Communications, Oxford University Press, New York (2007).
5. P. Bhattacharya: Semiconductor Optoelectronic Devices, Prentice Hall of India (1997).
6. Ben G. Streetman: Solid State Electronics Devices, Pearson Prentice Hall.
7. D.A. Neamen, "Semiconductor Physics and Devices", Times Mirror High Education Group, Chicago, 1997.
8. E.S. Yang, "Microelectronic Devices", McGraw Hill, Singapore, 1988.
9. Online course: "Semiconductor Optoelectronics" by M R Shenoy on NPTEL.
10. Online course: "Optoelectronic Materials and Devices" by Monica Katiyar and Deepak Gupta on NPTEL.

BTPH114-18	Semiconductor Physics Lab	L-0, T-0, P-3	Credits - 1.5

Pre-requisite (if any): (i) High-school education			

Course Objectives: The aim and objective of the Lab course on Semiconductor Physics is to introduce the students of B.Tech. class to the formal structure of semiconductor physics so that they can use these in Engineering as per their requirement.

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

CO1	Able to verify some of the theoretical concepts learnt in the theory courses.
CO2	Trained in carrying out precise measurements and handling sensitive equipment.
CO3	Introduced to the methods used for estimating and dealing with experimental uncertainties and systematic "errors."
CO4	Learn to draw conclusions from data and develop skills in experimental design
CO5	Write a technical report which communicates scientific information in a clear and concise manner.

Detailed Syllabus:

Note: Students are expected to perform about 10-12 experiments from the following list, selecting minimum of 7-8 from the Section-A and 3-4 from the Section-B.

Section-A

1. To study the characteristic of different PN junction Diode-Ge and Si.
2. To analyze the suitability of a given Zener diode as a power regulator.
3. To find out the intensity response of a solar cell/Photo diode.
4. To find out the intensity response of a LED.
5. To determine the band gap of a semiconductor.
6. To determine the resistivity of a semiconductor by four probe method.
7. To confirm the de Broglie equation for electrons.
8. To study voltage regulation and ripple factor for a half-wave and a full-wave rectifier without and with different filters.
9. To study the magnetic field of a circular coil carrying current.
10. To find out polarizability of a dielectric substance.
11. To study B-H curve of a ferro-magnetic material using CRO.
12. To find out the frequency of AC mains using electric-vibrator.
13. To find the velocity of ultrasound in liquid.
14. To study the Hall effect for the determination of charge current densities.
15. Distinguish between Diamagnetic material, Paramagnetic and ferromagnetic material.
16. Measurement of susceptibility of a liquid or a solution by Quincke's method.
17. To study the sample with the nano-scale objects and measure surface topography with different scales, width and height of nano objects, and force-distance curves using AFM.
18. To study the temperature coefficient of Resistance of copper.
19. To determine the ratio k/e Using a transistor.
20. To compare various capacitance and verify the law of addition of capacitance.
21. To determine dipole moment of an organic molecule acetone.
22. To measure the temperature dependence of a ceramic capacitor.
23. Verification of the curie Weiss law for the electrical susceptibility of a ferromagnetic material.
24. To study the laser beam characteristics like; wave length using diffraction grating aperture & divergence.
25. To study laser interference using Michelson's Interferometer.
26. Study of diffraction using laser beam and thus to determine the grating element.

Section-B

Virtual lab:

1. To draw the static current-voltage (I-V) characteristics of a junction diode.
2. To plot the characteristics of thermistor and hence find the temperature coefficient of resistance.
3. To determine the resistivity of semiconductors by Four Probe Method.
4. To study Zener diode voltage as regulator and measure its line and load regulation.
5. To study the B-H Curve for a ferromagnetic material.
6. To study the Hall effect experiment to determine the charge carrier density.
7. To determine the magnetic susceptibilities of paramagnetic liquids by Quincke's Method.
8. To study the phenomena of magnetic hysteresis and calculate the retentivity, coercivity and saturation magnetization

of a material using a hysteresis loop tracer.

9. Verification and design of combinational logic using AND, OR, NOT, NAND and XOR gates.

Reference books and suggested reading:

1. Advanced Practical Physics for students, B.L. Flint and H.T. Worsnop, 1971, Asia Publishing House.
2. Advanced level Physics Practicals, Michael Nelson and Jon M. Ogborn, 4th Edition, reprinted 1985, Heinemann Educational Publishers.
3. A Text Book of Practical Physics, I. Prakash & Ramakrishna, 11 th Edn, 2011, Kitab Mahal.
4. Engineering Practical Physics, S. Panigrahi & B. Mallick, 2015, Cengage Learning India Pvt. Ltd.
5. Practical Physics, G.L. Squires, 2015, 4 th Edition, Cambridge University Press.
6. Laboratory Experiments in College Physics, C.H. Bernard and C.D. Epp, John Wiley and Sons, Inc., New York, 1995.
7. Practical Physics, G.L. Squires, Cambridge University Press, Cambridge, 1985.
8. Experiments in Modern Physics, A.C. Melissinos, Academic Press, N.Y., 1966.
9. Practical Physics, C L Arora, S. Chand & Company Ltd.
10. http://www.vlab.co.in
11. 11. http://vlab.amrita.edu/index.php?sub=1

BTAM104-18	Mathematics Paper-I (Calculus & Linear Algebra)	4L, 1T, 0P	credits - 4

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

Detailed Contents:

Section-A

Unit-I: Calculus (13 hours)

Rolle's theorem, Mean value theorems, Statements of Taylor's and Maclaurin theorems with remainders; Indeterminate forms and L' Hôpital's rule; Maxima and minima. Evaluation of definite and improper integrals; Applications of definite integrals to evaluate surface areas and volumes of revolutions; Beta and Gamma functions and their properties.

Unit-II: Matrix Algebra (12 hours)

Matrices, vectors addition and scalar multiplication, matrix multiplication; Linear systems of equations, linear Independence, rank of a matrix, determinants, Cramer's Rule, inverse of a matrix, Gauss elimination and Gauss-Jordan elimination.

Section-B

Unit-III: Linear Algebra (13 hours)

Vector Space, linear dependence of vectors, basis, dimension; Linear transformations (maps), range and kernel of a linear map, rank and nullity, statement of rank-nullity theorem, Matrix associated with a linear map.

Unit-IV: Linear Algebra (Contd.) (12 hours)

Eigenvalues, eigenvectors, symmetric, skew-symmetric, and orthogonal Matrices, eigen bases; Similar matrices, diagonalization.

Suggested Text/Reference Books

1. G.B. Thomas and R.L. Finney, Calculus and Analytic geometry, 9th Edition, Pearson, Reprint, 2002.
2. Erwin Kreyszig, Advanced Engineering Mathematics, 9th Edition, John Wiley & Sons, 2006.
3. D. Poole, Linear Algebra: A Modern Introduction, 2nd Edition, Brooks/Cole, 2005.
4. Veerarajan T., Engineering Mathematics for first year, Tata McGraw-Hill, New Delhi, 2008.
5. B.V. Ramana, Higher Engineering Mathematics, Tata McGraw Hill New Delhi, 11th Reprint, 2010.

6. N.P. Bali and Manish Goyal, A text book of Engineering Mathematics, Laxmi Publications, Reprint, 2010.
7. B.S. Grewal, Higher Engineering Mathematics, Khanna Publishers, 35th Edition, 2000.
8. V. Krishnamurthy, V.P. Mainra and J.L. Arora, An introduction to Linear Algebra, Affiliated East–West press, Reprint 2005.

Course Outcomes: The students will be able

To apply differential and integral calculus to notions of curvature and to improper integrals. Apart from various applications, they will have a basic understanding of Beta and Gamma functions. The essential tools of matrices and linear algebra including linear transformations, eigenvalues, diagonalization and orthogonalization.

BTEE-101-18	Basic Electrical Engineering	[L: 3; T:1; P : 0]	credits - 4

Pre-requisites (if any): Nil

Detailed contents:

Module 1: DC Circuits (8 hours)

Electrical circuit elements (R, L and C), voltage and current sources, Kirchoff's current and voltage laws, analysis of simple circuits with dc excitation. Superposition, Thevenin's and Norton's Theorems. Time-domain analysis of first-order RL and RC circuits.

Module 2: AC Circuits (8 hours)

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

Module 3: Transformers (6 hours)

Magnetic materials, BH characteristics, ideal and practical transformer, equivalent circuit, losses in transformers, regulation and efficiency. Auto-transformer and three phase transformer connections.

Module 4: Electrical Machines (8 hours)

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

Module 5: Power Converters (6 hours)

DC-DC buck and boost converters, duty ratio control. Single-phase and three-phase voltage source inverters; sinusoidal modulation.

Module 6: Electrical Installations (6 hours)

Components of LT Switchgear: Switch Fuse Unit (SFU), MCB, ELCB, MCCB, Types of Wires and Cables, Earthing. Types of Batteries, Important Characteristics for Batteries. Elementary calculations for energy consumption, power factor improvement and battery backup.

Suggested Text / Reference Books

1. D. P. Kothari and I. J. Nagrath, "Basic Electrical Engineering", Tata McGraw Hill, 2010.
2. D. C. Kulshreshtha, "Basic Electrical Engineering", McGraw Hill, 2009.
3. L. S. Bobrow, "Fundamentals of Electrical Engineering", Oxford University Press, 2011.
4. E. Hughes, "Electrical and Electronics Technology", Pearson, 2010.
5. V. D. Toro, "Electrical Engineering Fundamentals", Prentice Hall India, 1989.

Course Outcomes

1. To understand and analyze basic electric and magnetic circuits
2. To study the working principles of electrical machines and power converters.
3. To introduce the components of low voltage electrical installations

----

BTEE-102-18	Basic Electrical Engineering Laboratory	[L: 0; T:0; P : 2]	1 credit

Pre-requisites (if any): Nil

List of experiments/demonstrations:

- Basic safety precautions. Introduction and use of measuring instruments – voltmeter, ammeter, multi-meter, oscilloscope. Real-life resistors, capacitors and inductors.
- Measuring the steady-state and transient time-response of R-L, R-C, and R-L-C circuits to a step change in voltage (transient may be observed on a storage oscilloscope). Sinusoidal steady state response of R-L, and R-C circuits – impedance calculation and verification. Observation of phase differences between current and voltage. Resonance in R-L-C circuits.
- Transformers: Observation of the no-load current waveform on an oscilloscope (non-sinusoidal wave-shape due to B-H curve nonlinearity should be shown along with a discussion about harmonics). Loading of a transformer: measurement of primary and secondary voltages and currents, and power.
- Three-phase transformers: Star and Delta connections. Voltage and Current relationships (line-line voltage, phase-to-neutral voltage, line and phase currents). Phase-shifts between the primary and secondary side. Cumulative three-phase power in balanced three-phase circuits.
- Demonstrate of cut-out sections of machines: dc machine (commutator-brush arrangement), induction machine (squirrel cage rotor), synchronous machine (field winging - slip ring arrangement) and single-phase induction machine.
- Torque Speed Characteristic of separately excited dc motor.
- Synchronous speed of two and four-pole, three-phase induction motors. Direction reversal by change of phase-sequence of connections. Torque-Slip Characteristic of an induction motor. Generator operation of an induction machine driven at super-synchronous speed.
- Synchronous Machine operating as a generator: stand-alone operation with a load. Control of voltage through field excitation.
- Demonstration of (a) dc-dc converters (b) dc-ac converters – PWM waveform (c) the use of dc-ac converter for speed control of an induction motor and (d) Components of LT switchgear.

Laboratory Outcomes

I. Get an exposure to common electrical components and their ratings.
II. Make electrical connections by wires of appropriate ratings.
III. Understand the usage of common electrical measuring instruments.
IV. Understand the basic characteristics of transformers and electrical machines.
V. Get an exposure to the working of power electronic converters.

Sr. No.	Suggested List of Experiments
1.	To verify Ohm's Law and its limitations.
2.	To verify Kirchhoff's Laws.
3.	To measure the resistance and inductance of a coil by ammeter-voltmeter method
4.	To find voltage-current relationship in a R-L series circuit and to determine the power factor of the circuit
5.	To verify the voltage and current relations in star and delta connected systems.
6.	To measure power and power factor in a single- phase AC circuit.
7.	To verify series and parallel resonance in AC circuits.
8.	To observe the B-H loop of ferromagnetic core material on CRO.
9.	To use a bridge rectifier for full- wave rectification of AC supply and to determine the relationship between RMS and average values of the rectified voltage
10.	To measure the minimum operating voltage, current drawn, power consumed, and the power factor of a fluorescent tube light.
11.	To connect measuring analog and digital instruments to measure current, voltage, power and power factor.
12.	To obtain the characteristics of a transistor under common base (CB) and common emitter (CE) configuration.
13.	To perform open- and short circuit tests on a single- phase transformer and calculate its efficiency
14.	To start and reverse the direction of rotation of a (i) DC motor (ii) Induction motor

| 15. | Determining of voltage regulation of transformer by directly loading. |
| 16. | Study of starters for (i) DC motor (ii) Induction motor |

BTME101-18	Engineering Graphics & Design (Theory & Lab)	L:1 T:0 P:4	Credits - 3
Pre-requisites (if any): Nil			

Detailed contents:

Traditional Engineering Graphics:

Principles of Engineering Graphics; Orthographic Projection; Descriptive Geometry; Drawing Principles; Isometric Projection; Surface Development; Perspective; Reading a Drawing; Sectional Views; Dimensioning & Tolerances; True Length, Angle; intersection, Shortest Distance.

Computer Graphics:

Engineering Graphics Software; -Spatial Transformations; Orthographic Projections; Model Viewing; Co-ordinate Systems; Multi-view Projection; Exploded Assembly; Model Viewing; Animation; Spatial Manipulation; Surface Modelling; Solid Modelling; Introduction to Building Information Modelling (BIM)

(Except the basic essential concepts, most of the teaching part can happen concurrently in the laboratory)

Module 1: Introduction to Engineering Drawing covering

Principles of Engineering Graphics and their significance, usage of Drawing instruments, lettering, Conic sections including the Rectangular Hyperbola (General method only); Cycloid, Epicycloid, Hypocycloid and Involute; Scales – Plain, Diagonal and Vernier Scales;

Module 2: Orthographic Projections covering

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

Module 3: Projections of Regular Solids covering

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

Module 4: Sections and Sectional Views of Right Angular Solids covering

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

Module 5: Isometric Projections covering

Principles of Isometric projection – Isometric Scale, Isometric Views, Conventions; Isometric Views of lines, Planes, Simple and compound Solids; Conversion of Isometric Views to Orthographic Views and Vice-versa, Conventions;

Module 6: Overview of Computer Graphics covering

listing the computer technologies that impact on graphical communication, Demonstrating knowledge of the theory of CAD software [such as: The Menu System, Toolbars (Standard, Object Properties, Draw, Modify and Dimension), Drawing Area (Background, Crosshairs, Coordinate System), Dialog boxes and windows, Shortcut menus (Button Bars), The Command Line (where applicable), The Status Bar, Different methods of zoom as used in CAD, Select and erase objects.; Isometric Views of lines, Planes, Simple and compound Solids];

Module 7: Customisation & CAD Drawing

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

Module 8: Annotations, layering & other functions covering

applying dimensions to objects, applying annotations to drawings; Setting up and use of layers, layers to create drawings, Create, edit and use customized layers; Changing line lengths through modifying existing lines (extend/lengthen); Printing documents to paper using the print command; orthographic projection techniques; Drawing sectional views of composite right regular geometric solids and project the true shape of the sectioned surface; Drawing annotation, Computer-aided design (CAD) software modeling of parts and assemblies. Parametric and nonparametric solid, surface, and wireframe models. Part editing and two-dimensional documentation of models. Planar projection theory, including sketching of perspective, isometric, multiview, auxiliary, and section views. Spatial visualization exercises. Dimensioning guidelines, tolerancing techniques; dimensioning and scale multi views of dwelling;

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

Geometry and topology of engineered components: creation of engineering models and their presentation in standard 2D blueprint form and as 3D wire-frame and shaded solids; meshed topologies for engineering analysis and tool-path generation for component manufacture; geometric dimensioning and tolerancing; Use of solid-modeling software for creating associative models at the component and assembly levels; floor plans that include: windows, doors, and fixtures such as WC, bath, sink, shower, etc. Applying colour coding according to building drawing practice; Drawing sectional elevation showing foundation to ceiling; Introduction to Building Information Modelling (BIM).

Suggested Text/Reference Books:

1. Bhatt N.D., Panchal V.M. & Ingle P.R., (2014), Engineering Drawing, Charotar Publishing House
2. Shah, M.B. & Rana B.C. (2008), Engineering Drawing and Computer Graphics, Pearson Education
3. Agrawal B. & Agrawal C. M. (2012), Engineering Graphics, TMH Publication
4. Narayana, K.L. & P Kannaiah (2008), Text book on Engineering Drawing, SciTech Publishers
5. (Corresponding set of) CAD Software Theory and User Manuals Course Outcomes

Course Outcomes

All phases of manufacturing or construction require the conversion of new ideas and design concepts into the basic line language of graphics. Therefore, there are many areas (civil, mechanical, electrical, architectural and industrial) in which the skills of the CAD technicians play major roles in the design and development of new products or construction. Students prepare for actual work situations through practical training in a new state-of-the-art computer designed CAD laboratory using engineering software. This course is designed to address:

- to prepare you to design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability
- to prepare you to communicate effectively
- to prepare you to use the techniques, skills, and modern engineering tools necessary for engineering practice

The student will learn :

- Introduction to engineering design and its place in society
- Exposure to the visual aspects of engineering design
- Exposure to engineering graphics standards
- Exposure to solid modelling
- Exposure to computer-aided geometric design
- Exposure to creating working drawings
- Exposure to engineering communication

Engineering Graphics & Design (Practical)
Course Assessment Methods

End Semester Assessment:
1. University Theory Exam: Nil
2. University Practical Exam: 40 Marks (Evaluation of Traditional Engineering Graphics part of 20 Marks should be

based upon written test by External Practical Examiner & Evaluation of Computer Graphics part of 20 marks should be based upon lab performance using computer graphics software & viva voce by External Practical Examiner)

Internal Assessment:

1. 60 Marks (20 marks for day to day work, 20 marks for written test & 20 marks for internal viva voce)

Second Semester

BTCH101-18	Chemistry-I (Theory)	L:3 T:1 P:0	Credits: 4

Pre-requisites (if any): Nil

Detailed contents

1. **Atomic and molecular structure (12 lectures)**

Schrodinger equation. Particle in a box solutions and their applications for conjugated molecules and nanoparticles. Forms of the hydrogen atom wave functions and the plots of these functions to explore their spatial variations. Molecular orbitals of diatomic molecules and plots of the multicenter orbitals. Equations for atomic and molecular orbitals. Energy level diagrams of diatomic. Pi-molecular orbitals of butadiene and benzene and aromaticity. Crystal field theory and the energy level diagrams for transition metal ions and their magnetic properties. Band structure of solids and the role of doping on band structures.

2. **Spectroscopic techniques and applications (8 lectures)**

Principles of spectroscopy and selection rules. Electronic spectroscopy. Fluorescence and its applications in medicine. Vibrational and rotational spectroscopy of diatomic molecules. Applications. Nuclear magnetic resonance and magnetic resonance imaging, surface characterisation techniques. Diffraction and scattering.

3. **Intermolecular forces and potential energy surfaces (4 lectures)**

Ionic, dipolar and van Der Waals interactions. Equations of state of real gases and critical phenomena. Potential energy surfaces of H3, H2F and HCN and trajectories on these surfaces.

4. Use of free energy in chemical equilibria (6 lectures)

Thermodynamic functions: energy, entropy and free energy. Estimations of entropy and free energies. Free energy and emf. Cell potentials, the Nernst equation and applications. Acid base, oxidation reduction and solubility equilibria. Water chemistry. Corrosion.

Use of free energy considerations in metallurgy through Ellingham diagrams.

5. **Periodic properties (4 Lectures)**

Effective nuclear charge, penetration of orbitals, variations of s, p, d and f orbital energies of atoms in the periodic table, electronic configurations, atomic and ionic sizes, ionization energies, electron affinity and electronegativity, polarizability, oxidation states, coordination numbers and geometries, hard soft acids and bases, molecular geometries

6. **Stereochemistry (4 lectures)**

Representations of 3 dimensional structures, structural isomers and stereoisomers, configurations and symmetry and chirality, enantiomers, diastereomers, optical activity, absolute configurations and conformational analysis. Isomerism in transitional metal compounds

7. **Organic reactions and synthesis of a drug molecule (4 lectures)**

Introduction to reactions involving substitution, addition, elimination, oxidation, reduction, cyclization and ring openings. Synthesis of a commonly used drug molecule.

Suggested Text Books
1. University chemistry, by B. H. Mahan
2. Chemistry: Principles and Applications, by M. J. Sienko and R.A. Plane
3. Fundamentals of Molecular Spectroscopy, by C. N. Banwell
4. Engineering Chemistry (NPTEL Web-book), by B. L. Tembe, Kamaluddin and M. S. Krishnan
5. Physical Chemistry, by P. W. Atkins (
6. Organic Chemistry: Structure and Function by K. P. C. Volhardt and N. E. Schore, 5th Edition
   http://bcs.whfreeman.com/vollhardtschore5e/default.asp

Course Outcomes

The concepts developed in this course will aid in quantification of several concepts in chemistry that have been introduced at the 10+2 levels in schools. Technology is being increasingly based on the electronic, atomic and molecular level modifications.

Quantum theory is more than 100 years old and to understand phenomena at nanometer levels, one has to base the description of all chemical processes at molecular levels. The course will enable the student to:

- Analyse microscopic chemistry in terms of atomic and molecular orbitals and intermolecular forces.
- Rationalise bulk properties and processes using thermodynamic considerations.
- Distinguish the ranges of the electromagnetic spectrum used for exciting different molecular energy levels in various spectroscopic techniques.
- Rationalise periodic properties such as ionization potential, electronegativity, oxidation states and electronegativity.
- List major chemical reactions that are used in the synthesis of molecules

----

BTCH102-18	Chemistry-I (Lab.)	L:0 T:0 P:3	Credits- 1.5
Choice of 10-12 experiments from the following			

- Determination of surface tension and viscosity
- Thin Layer Chromatography
- Ion exchange column for removal of hardness of water
- Colligative properties using freezing point depression
- Determination of the rate constant of a reaction
- Determination of cell constant and conductance of solutions
- Potentiometry-determination of redox potentials and emf
- Synthesis of a polymer/drug
- Saponification/acid value of an oil
- Chemical analysis of a salt
- Lattice structures and packing of spheres
- Models of potential energy surfaces
- Chemical oscillations- Iodine clock reaction
- Determination of the partition coefficient of a substance between two immiscible liquids
- Adsorption of acetic acid by charcoal
- Use of the capillary viscometers to the demonstrate of the isoelectric point as the pH of minimum viscosity for gelatin sols and/or coagulation of the white part of egg.

Laboratory Outcomes

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

- Estimate rate constants of reactions from concentration of reactants/products as a function of time
- Measure molecular/system properties such as surface tension, viscosity, conductance of solutions, redox potentials, chloride content of water, etc
- Synthesize a small drug molecule and analyse a salt sample

BTA204-18	Mathematics Paper-II (Probability & Statistics)	4L:1T:0P	credits - 4
Course Objective:			

The objective of this course is to familiarize the students with statistical techniques. It aims to equip the students with standard concepts and tools at an intermediate to advanced level that will serve them well towards tackling various problems in the discipline.

Detailed Content:

<div align="center">

Section-A

</div>

Unit I: (10 hours)

Measures of Central tendency: Moments, skewness and kurtosis, Variance, Correlation coefficient, Probability, conditional probability, independence; Discrete random variables, Independent random variables, expectation of Discrete random variables.

Unit II: (15 hours)

Probability distributions: Binomial, Poisson and Normal, Poisson approximation to the binomial distribution, evaluation of statistical parameters for these three distributions, Correlation and regression – Rank correlation.

Section-B

Unit III: (10 hours)

Continuous random variables and their properties, distribution functions and densities, normal and exponential densities. Curve fitting by the method of least squares- fitting of straight lines, second degree parabolas.

Unit IV; (15 hours)

Test of significance: Large sample test for single proportion, difference of proportions, single mean, difference of means, and difference of standard deviations.

Test for single mean, difference of means and correlation coefficients, test for ratio of variances - Chi-square test for goodness of fit and independence of attributes.

Suggested Text/Reference Books
- Erwin Kreyszig, Advanced Engineering Mathematics, 9th Edition, John Wiley & Sons, 2006.
- P. G. Hoel, S. C. Port and C. J. Stone, Introduction to Probability Theory, Universal Book Stall, 2003 (Reprint).
- S. Ross, A First Course in Probability, 6th Ed., Pearson Education India, 2002.
- W. Feller, An Introduction to Probability Theory and its Applications, Vol. 1, 3rd Ed., Wiley, 1968.
- N.P. Bali and Manish Goyal, A text book of Engineering Mathematics, Laxmi Publications, Reprint, 2010.
- B.S. Grewal, Higher Engineering Mathematics, Khanna Publishers, 35th Edition, 2000.
- T. Veerarajan, Engineering Mathematics (for semester III), Tata McGraw-Hill, New Delhi, 2010.

Course Outcomes:
The students will learn:
- The ideas of probability and random variables and various discrete and continuous probability distributions and their properties. The basic ideas of statistics including measures of central tendency, correlation and regression and the statistical methods of studying data samples.

BTPS101-18	Programming for Problem Solving (Theory)	L:3 T:0 P:0	Credits: 3
Pre-requisites (if any): Nil			
Detailed contents			

Unit 1

Introduction to Programming **(4 lectures)**
Introduction to components of a computer system (disks, memory, processor, where a program is stored and executed, operating system, compilers etc.) –
(1 lecture).

Idea of Algorithm: steps to solve logical and numerical problems. Representation of Algorithm: Flowchart/Pseudocode with examples. **(1 lecture)**

From algorithms to programs; source code, variables (with data types) variables and memory locations, Syntax and Logical Errors in compilation, object and executable code- **(2 lectures)**

Unit 2

Arithmetic expressions and precedence **(2 lectures)**

Conditional Branching and Loops **(6 lectures)**
Writing and evaluation of conditionals and consequent branching **(3 lectures)**
Iteration and loops (**3 lectures**)

Unit 3

Arrays **(6 lectures)**
Arrays (1-D, 2-D), Character arrays and Strings

Unit 4

Basic Algorithms **(6 lectures)**
Searching, Basic Sorting Algorithms (Bubble, Insertion and Selection), Finding roots of equations, notion of order of complexity through example programs (no formal definition required)

Unit 5

Function **(5 lectures)**

Functions (including using built in libraries), Parameter passing in functions, call by value, Passing arrays to functions: idea of call by reference

Unit 6

Recursion **(4 -5 lectures**)

Recursion, as a different way of solving problems. Example programs, such as Finding Factorial, Fibonacci series, Ackerman function etc. Quick sort or Merge sort.

Unit 7

Structure **(4 lectures)**
Structures, Defining structures and Array of Structures

Unit 8

Pointers **(2 lectures)**

Idea of pointers, Defining pointers, Use of Pointers in self-referential structures, notion of linked list (no implementation)

Unit 9

File handling (only if time is available, otherwise should be done as part of the lab)

Suggested Text Books:

  (i)   Byron Gottfried, Schaum's Outline of Programming with C, McGraw-Hill
  (ii)   E. Balaguruswamy, Programming in ANSI C, Tata McGraw-Hil

Suggested Reference Books
  (i)   Brian W. Kernighan and Dennis M. Ritchie, The C Programming Language, Prentice Hall of India.

Course Outcomes

The student will learn
  - To formulate simple algorithms for arithmetic and logical problems.
  - To translate the algorithms to programs (in C language).
  - To test and execute the programs and correct syntax and logical errors.
  - To implement conditional branching, iteration and recursion.
  - To decompose a problem into functions and synthesize a complete program using divide and conquer approach.
  - To use arrays, pointers and structures to formulate algorithms and programs.
  - To apply programming to solve matrix addition and multiplication problems and searching and sorting problems.

- To apply programming to solve simple numerical method problems, namely rot finding of function, differentiation of function and simple integration.

BTPS102-18	Programming for Problem Solving (Lab)	L:0 T:0 P:4	Credits: 2

Pre-requisites (if any): Nil

[The laboratory should be preceded or followed by a tutorial to explain the approach or algorithm to be implemented for the problem given.]

Tutorial 1: Problem solving using computers:
Lab1: Familiarization with programming environment

Tutorial 2: Variable types and type conversions:
Lab 2: Simple computational problems using arithmetic expressions

Tutorial 3: Branching and logical expressions:
Lab 3: Problems involving if-then-else structures

Tutorial 4: Loops, while and for loops:
Lab 4: Iterative problems e.g., sum of series

Tutorial 5: 1D Arrays: searching, sorting:
Lab 5: 1D Array manipulation

Tutorial 6: 2D arrays and Strings
Lab 6: Matrix problems, String operations

Tutorial 7: Functions, call by value:
Lab 7: Simple functions

Tutorial 8 &9: Numerical methods (Root finding, numerical differentiation, numerical integration):
Lab 8 and 9: Programming for solving Numerical methods problems

Tutorial 10: Recursion, structure of recursive calls
Lab 10: Recursive functions

Tutorial 11: Pointers, structures and dynamic memory allocation
Lab 11: Pointers and structures

Tutorial 12: File handling:
Lab 12: File operations

Laboratory Outcomes

- To formulate the algorithms for simple problems
- To translate given algorithms to a working and correct program
- To be able to correct syntax errors as reported by the compilers
- To be able to identify and correct logical errors encountered at run time
- To be able to write iterative as well as recursive programs
- To be able to represent data in arrays, strings and structures and manipulate them through a program
- To be able to declare pointers of different types and use them in defining self-referential structures.
- To be able to create, read and write to and from simple text files.

BTMP101-18	Workshop/Manufacturing Practices (Theory)	L:1 T:0 P:0	Credits:3

Pre-requisites (if any): Nil

Detailed contents
1. Manufacturing Methods- casting, forming, machining, joining, advanced manufacturing methods (3 lectures)
2. CNC machining, Additive manufacturing (1 lecture)
3. Fitting operations & power tools (1 lecture)

4. Electrical &Electronics (1 lecture)
5. Carpentry (1 lecture)
6. Plastic moulding, glass cutting (1 lecture)
7. Metal casting (1 lecture)
8. Welding (arc welding & gas welding), brazing (1 lecture)

Suggested Text/Reference Books:

(i) Hajra Choudhury S.K., Hajra Choudhury A.K. and Nirjhar Roy S.K., "Elements of Workshop Technology", Vol. I 2008 and Vol. II 2010, Media promoters and publishers private limited, Mumbai.
(ii) Kalpakjian S. And Steven S. Schmid, "Manufacturing Engineering and Technology", 4th edition, Pearson Education India Edition, 2002.
(iii) Gowri P. Hariharan and A. Suresh Babu," Manufacturing Technology – I" Pearson Education,2008.
(iv) Roy A. Lindberg, "Processes and Materials of Manufacture", 4 th edition, Prentice Hall India,1998.
(v) Rao P.N., " Manufacturing Technology" , Vol. I and Vol. II, Tata McGrawHill House, 2017.

Course Outcomes
Upon completion of this course, the students will gain knowledge of the different manufacturing processes which are commonly employed in the industry, to fabricate components using different materials.

BTMP101-18	Workshop Practice	L : 0; T:0 ; P : 4	credits - 2

1. Machine shop **(10 hours)**
2. Fitting shop **(8 hours)**
3. Carpentry **(6 hours)**
4. Electrical & Electronics **(8 hours)**
5. Welding shop (**8 hours (Arc welding 4 hrs + gas welding 4 hrs**)
6. Casting **(8 hours)**
7. Smithy (6 hours)
8. Plastic moulding& Glass Cutting **(6 hours)**

Examinations could involve the actual fabrication of simple components, utilizing one or more of the techniques covered above.

Laboratory Outcomes

Upon completion of this laboratory course, students will be able to fabricate components with their own hands. They will also get practical knowledge of the dimensional accuracies and dimensional tolerances possible with different manufacturing processes. By assembling different components, they will be able to produce small devices of their interest.

BTHU-101-18	English	2L: 0T: 0P	credits - 2

Course Outcomes:
- The objective of the course is to help the students become the independent users of English language.
- Students will acquire basic proficiency in reading & listening, comprehension, writing and speaking skills.
- Students will be able to understand spoken and written English language, particularly the language of their chosen technical field.
- They will be able to converse fluently.
- They will be able to produce on their own clear and coherent texts.

Detailed contents

Unit-1 Vocabulary Building & Basic Writing Skills

- The concept of Word Formation
- Root words from foreign languages and their use in English
- Acquaintance with prefixes and suffixes from foreign languages in English to form derivatives.
- Synonyms, antonyms, and standard abbreviations.
- Sentence Structures

- Use of phrases and clauses in sentences
- Importance of proper punctuation
- Creating coherence
- Organizing principles of paragraphs in documents
- Techniques for writing precisely

Unit-2 Identifying Common Errors in Writing

- Subject-verb agreement
- Noun-pronoun agreement
- Misplaced modifiers
- Articles
- Prepositions
- Redundancies
- Clichés

Unit-3 Mechanics of Writing

- Writing introduction and conclusion
- Describing
- Defining
- Classifying
- Providing examples or evidence

Unit-4 Writing Practices

- Comprehension
- Précis Writing
- Essay Writing
- Business Writing-Business letters, Business Emails, Report Writing, Resume/CV

Suggested Readings:

(i)   Practical English Usage. Michael Swan. OUP. 1995.
(ii)   Remedial English Grammar. F.T. Wood. Macmillan.2007
(iii)   On Writing Well. William Zinsser. Harper Resource Book. 2001
(iv)   Study Writing. Liz Hamp-Lyons and Ben Heasly. Cambridge University Press. 2006.
(v)   Communication Skills. Sanjay Kumar and Pushp Lata. Oxford University Press. 2011.
(vi)   Exercises in Spoken English. Parts. I-III. CIEFL, Hyderabad. Oxford University Press

----

BTHU-102-18	English Laboratory	0L: 0T: 2P	1 credit

Course Outcomes:
- The objective of the course is to help the students become the independent users of English language.
- Students will acquire basic proficiency in listening and speaking skills.
- Students will be able to understand spoken English language, particularly the language of their chosen technical field.
- They will be able to converse fluently
- They will be able to produce on their own clear and coherent texts.

Detailed contents

Interactive practice sessions in Language Lab on Oral Communication

- Listening Comprehension
- Self-Introduction, Group Discussion and Role Play
- Common Everyday Situations: Conversations and Dialogues
- Communication at Workplace

- Interviews
- Formal Presentations

Suggested Readings:

(i)   Practical English Usage. Michael Swan. OUP. 1995.
(ii)   Communication Skills. Sanjay Kumar and Pushp Lata. Oxford University Press. 2011.
(iii) Exercises in Spoken English. Parts. I-III. CIEFL, Hyderabad. Oxford University Press

Third Semester

Course Code: BTES 301-18	Course Title: Digital Electronics	3L:0T:0P	3 Credits

Detailed Contents:

Module 1:   NUMBER SYSTEMS:
Binary, Octal, Decimal, Hexadecimal. Number base conversions, 1's, 2'scomplements, signed Binary numbers. Binary Arithmetic, Binary codes: Weighted BCD, Graycode, Excess 3code, ASCII.
LOGIC GATES: AND, OR, NOT, NAND, NOR, Exclusive-OR and Exclusive-NOR. Implementations of Logic Functions using gates, NAND-NOR implementations.

Module 2:   BOOLEAN ALGEBRA:
Boolean postulates and laws–De-Morgan's Theorem, Principle of Duality, Boolean expression–Boolean function, Minimization of Boolean expressions–Sum of Products (SOP), Product of Sums (POS), Minterm, Maxterm, Canonical forms, Conversion between canonical forms, Karnaughmap Minimization, Don't care conditions, Quine- McCluskey method.

Module 3:   COMBINATIONAL CIRCUITS:
Design procedure – Adders, Subtractors, BCD adder, Magnitude Comparator, Multiplexer/ Demultiplexer, encoder/decoder, parity checker, code converters. Implementation of combinational logic using MUX, BCD to 7 segment decoder.

SEQUENTIAL CIRCUITS: Flip flops SR, JK, T, D and Master slave, Excitation table, Edge triggering, Level Triggering, Realization of one flip flop using other flip flops. Asynchronous/Ripple counters, Synchronous counters, Modulo-n counter, Ring Counters. Design of Synchronous counters: state diagram, Circuit implementation. Shift registers.

Module 4:   MEMORY DEVICES:
Classification of memories, RAM organization, Write operation, Read operation, Memory cycle. ROM organization, PROM, EPROM, EEPROM, Programmable logic array, Programmable array logic, complex Programmable logic devices (CPLDS), Field Programmable Gate Array (FPGA).
A/D & D/A CONVERTORS: Analog & Digital signals. Sample and hold circuit, A/D and D/A conversion techniques (Weighted type, R-2R Ladder type, Counter Type, Dual Slope type, Successive Approximation type).

COURSE OUTCOME: At the end of course the student will be able to:

1. Demonstrate the operation of simple digital gates, identify the symbols, develop the truth table for those gates; combine simple gates in to more complex circuits; change binary, hexadecimal, octal numbers to their decimal equivalent an vice versa.
2. Demonstrate the operation of a flip-flop. Design counters and clear the concept to f shift registers.
3. Study different types of memories and their applications. Convert digital signal into analog and vice versa.

Suggested Readings/Books:

1. Morris Mano, Digital Design, Prentice Hall of India Pvt. Ltd
2. Donald P. Leach and Albert Paul Malvino, Digital Principles and Applications, 5 ed., Tata
3. McGraw Hill Publishing Company Limited, New Delhi, 2003.
4. R.P.Jain, Modern Digital Electronics, 3ed. Tata McGraw–Hill publishing company limited, New Delhi, 2003.
5. Thomas L.Floyd, Digital Fundamentals, Pearson Education, Inc, New Delhi, 2003.
6. Ronald J. Tocci, Neal S. Widmer, Gregory L. Moss, Digital System – Principles and Applications, Pearson Education
7. Ghosal, Digital Electronics, Cengage Learning.

Course Code: BTCS 301-18	Course Title: Data Structure & Algorithms	3L:1T:0P	3Credits

Module1: Introduction

Basic Terminologies: Elementary Data Organizations, Data Structure Operations: insertion, deletion, traversal etc.; Analysis of an Algorithm, Asymptotic Notations, Time-Space trade off.
Searching: Linear Search and Binary Search Techniques and their complexity analysis.

[6 hrs] (CO1)

Module2: Stacks and Queues

ADT Stack and its operations: Algorithms and their complexity analysis, Applications of Stacks: Expression Conversion and evaluation– corresponding algorithms and complexity analysis. ADT queue, Types of Queue: Simple Queue, Circular Queue, Priority Queue; Operations on each types of Queues: Algorithms and their analysis.

[10 hrs] (CO2, CO4, CO5)

Module3: Linked Lists

Singly linked lists: Representation in memory, Algorithms of several operations: Traversing, Searching, Insertion into, Deletion from linked list; Linked representation of Stack and Queue, Header nodes, Doubly linked list: operations on it and algorithmic analysis; Circular Linked Lists: All operations their algorithms and the complexity analysis.
Trees: Basic Tree Terminologies, Different types of Trees: Binary Tree, Threaded Binary Tree, Binary Search Tree, AVL Tree; Tree operations on each of the trees and their algorithms with complexity analysis. Applications of Binary Trees. B Tree, B+ Tree: definitions, algorithms and analysis.

[10 hrs] (CO2, CO4, CO5)

Module4: Sorting and Hashing

Objective and properties of different sorting algorithms: Selection Sort, Bubble Sort, Insertion Sort, Quick Sort, Merge Sort, Heap Sort; Performance and Comparison among all the methods, Hashing.

[10 hrs] (CO3)

Module4: Graph

Basic Terminologies and Representations, Graph search and traversal algorithms and complexity analysis.

[6 hrs] (CO2, CO4)

Course Outcomes:

The student will be able to:

1. For a given algorithm student will able to analyze the algorithms to determine the time and computation complexity and justify the correctness;
2. Student will be able to handle operation like searching, insertion, deletion, traversing on various Data Structures and determine time and computational complexity;
3. Student will able to write an algorithm Selection Sort, Bubble Sort, Insertion Sort, Quick Sort, Merge Sort, Heap Sort and compare their performance in term of Space and Time complexity;
4. Students will be able to choose appropriate Data Structure as applied to specific problem definition; &
5. Demonstrate the reusability of Data Structures for implementing complex iterative problems.

Suggested Books:

1. "Classic Data Structures", Samanta and Debasis, 2nd edition, PHI publishers.
2. "Fundamentals of Data Structures", Illustrated Edition by Ellis Horowitz, Sartaj Sahni, and Computer Science Press.
3. "Data Structures with C (Schaum's Outline Series)", Seymour Lipschutz, 1st edition, McGraw Hill Education.

Reference Books:

1. Algorithms, Data Structures, and Problem Solving with C++, Illustrated Edition by Mark Allen Weiss, Addison-Wesley Publishing Company.
2. How to solve it by Computer, 2nd Impression by R. G. Dromey, Pearson Education.

| Course Code: BTCS 302-18 | Course Title: Object Oriented Programming | 3L:0T:0P | 3Credits |

Objectives of the course:

The course will introduce standard tools and techniques for software development, using object oriented approach, use of a version control system, an automated build process, an appropriate framework for automated unit and integration tests.

Detailed Contents:

- Abstract data types and their specification.
- How to implement an ADT. Concrete state space, concrete invariant, abstraction function. Implementing operations, illustrated by the Text example.
- Features of object-oriented programming. Encapsulation, object identity, polymorphism – but not inheritance.
- Inheritance in OO design.
- Design patterns. Introduction and classification. The iterator pattern.
- Model-view-controller pattern.
- Commands as methods and as objects.
- Implementing OO language features.
- Memory management.
- Generic types and collections
- GUIs. Graphical programming with Scala and Swing
- The software development process.

The concepts should be practised using C++ and Java. Pearl may also be introduced wherever possible.

Suggested books

1. Object-Oriented Programming in C++, Robert Lafore, SAMS Publiishing.
2. Object Oriented Programming with C++ by E Balagurusamy
3. C++ How to Program, 10th Edition, Paul J. Deitel, Deitel & Associates, Inc., Harvey Deitel, Harvey M. Deitel

Course Outcomes

After taking the course, students will be able to:
1. Specify simple abstract data types and design implementations, using abstraction functions to document them.
2. Recognise features of object-oriented design such as encapsulation, polymorphism, inheritance, and composition of systems based on object identity.
3. Name and apply some common object-oriented design patterns and give examples of their use.
4. Design applications with an event-driven graphical user interface.

Course Code: BTAM304-18	Course Title: Mathematics Paper-III (Calculus and Ordinary Differential Equations)	4L:1T:0P	4Credits

Detailed Contents:

Module1:
Limit, continuity for functions with severable variables, partial derivatives, total derivative, Maxima, minima and saddle points; Method of Lagrange multipliers, Multiple Integration: double and t r i p l e integrals (Cartesian and polar), Change of order of integration in double integrals, Change of variables (Cartesiantopolar), Applications of double and triple integrals to find surface area and volumes.

[CO1, CO2] (12Hrs)

Module2:
Sequence and series, Bolzano Weirstrass Theorem, Cauchy convergence criterion for sequence, uniform convergence, convergence of positive term series: comparison test, limit comparison test, D'Alembert's ratio test, Raabe's test, Cauchy root test, p-test, Cauchy integral test, logarithmic test, Alternating series, Leibnitz test, Power series, Taylor's series, Series for exponential, trigonometric and logarithmic functions. **[CO3] (13Hrs.)**

Module3:
Exact, linear and Bernoulli's equations, Euler's equations, Equations not of first degree: equations solvable for p, equations solvable for y, equations solvable for x and Clairaut'stype.

[CO4] (12hrs.)

Module4:
Second and higher order linear differential equations with constant coefficients, method of variation of parameters, Equations reducible to linear equations with constant coefficients: Cauchy and Legendre's equations.

[CO5] (12hrs.)

Suggested Books

1. G.B. Thomas and R.L. Finney, Calculus and Analytic geometry, 9th Edition, Pearson, Reprint, 2002.
2. T. Veerarajan, Engineering Mathematics for first year, Tata McGraw-Hill, New Delhi, 2008.
3. N.P. Bali and Manish Goyal, A text book of Engineering Mathematics, Laxmi Publications, Reprint, 2010.
4. Erwin Kreyszig, Advanced Engineering Mathematics, 9th Edition, John Wiley & Sons, 2006.
5. W.E. Boyce and R.C. DiPrima, Elementary Differential Equations and Boundary Value Problems, 9th Edition, Wiley India, 2009.
6. E.A.Coddington,An Introduction to Ordinary Differential Equations, PHI, 1995

Course Outcomes:

At the end of the course, the student will be able to:
1. 1.Understand the functions of several variables that are essential in most branches of engineering.
2. Apply multiple integrals to deal with areas and volumes of various structures which are quite significant in real world.
3. Formulate and solve engineering problems related to convergence, infinite series, power series and Taylor series.
4. Create, select and utilize the learnt techniques of first degree ordinary differential equations to model real world problems
5. Be acquainted with the knowledge required to solve higher order ordinary differential equations.

Course Code: HSMC 101-18	Course Title: Development of Societies	2L:1T:0P	3Credits

Detailed Contents:

Unit I: Social Development   (5 hours)

1. Concepts behind the origin of Family, Clan and Society
2. Different Social Systems
3. Relation between Human being and Society
4. Comparative studies on different models of Social Structures and their evolution

Unit II: Political Development   (3 hours)

1. Ideas of Political Systems as learnt from History
2. Different models of Governing system and their comparative study

Unit III: Economic Development   (18 hours)

1. Birth of Capitalism, Socialism, Marxism
2. Concept of development in pre-British, British and post British period-Barter, Jajmani
3. Idea of development in current context.
4. E. F. Schumacher's idea of development, Buddhist economics. Gandhian idea of development. Swaraj and Decentralization.

PROJECT: Possible projects in this course could be

- Interact with local communities and understand their issues.
- Study local cottage industry and agricultural practices. Role of engineering and specialized knowledge.
- Evaluation of technology in the context of its application. Social impact of technology. Environmental impact of technology. Evaluation from a holistic perspective.

Course Code: HSMC102-18	Course Title: PHILOSOPHY	2L:1T:0P	3Credits

Detailed Contents:
Unit1:
The difference between knowledge (Vidya) and Ignorance (Avidya):
- Upanishads;
- Six systems orthodox and Heterodox Schools of Indian Philosophy.
- Greek Philosophy:

Unit2:
Origin of Universe:
- Nasidiya Sukta:"Who really knows?"
- Brhadaranyaka Upanishad; Chandogya Upanishad: Non-self, Self, real and unreal.
- Taittiriya Upanishad: Siksha Valli.
- Plato's Symposium: Lack as the source of desire and knowledge.
- Socratic's method of knowledge as discovery.
- Language: Word as root of knowledge (Bhartrahari'sVakyapadiyam)
- Fourteen Knowledge basis as a sources of Vidya: Four Vedas; Six auxiliary sciences (Vedangas); Purana, Nyaya, Mimamsa and Dharma Sastras.

Unit3:

Knowledge as Power: Francis Bacon. Knowledge as both power and self-realization in Bagavad Gita.

Unit4:

Knowledge as oppression: M. Foucault. Discrimination between Rtam and Satyam in Indian Philosophy.

Unit5:

Knowledge as invention: Modern definition of creativity; scientific activity in the claim that Science invents new things at least through technology.

Unit6:

Knowledge about the self, transcendental self; knowledge about society, polity and nature.

Unit7:

Knowledge about moral and ethics codes.

Unit8:

Tools of acquiring knowledge: Tantrayuktis, asystem ofinquiry(Caraka, Sushruta, Kautilya, Vyasa)

READINGS
1. Copleston, Frederick, History of Philosophy, Vol. 1. Great Britain: Continuum.
2. Hiriyanna, M. Outlines of Indian Philosophy, Motilal Banarsidass Publishers; Fifth Reprint edition (2009)
3. Sathaye, Avinash, Translation of Nasadiya Sukta
4. Ralph T. H. Griffith. The Hymns of the Ṛgveda. Motilal Banarsidass: Delhi: 1973.
5. Raju, P. T. Structural Depths of Indian Thought, Albany: State University of New York
6. Press.
7. Plato, Symposium, Hamilton Press.
8. Kautilya Artha Sastra. Penguin Books, New Delhi.
9. Bacon, Nova Orgum
10. Arnold, Edwin. The Song Celestial.
11. Foucault, Knowledge/Power.
12. Wildon, Anthony, System of Structure.
13. Lele, W.K. The Doctrine of Tantrayukti. Varanasi: Chowkamba Series.
14. Dasgupta, S. N. History of Indian Philosophy, Motilal Banasidas, Delhi.
15. Passmore, John, Hundred Years of Philosophy, Penguin.

ASSESSMENT (indicative only):

Ask students to do term papers, for example, writing biographical details of founders, sustainers, transmitters, modifiers, rewriters; translating monographs of less known philosophers such as K.C. Bhattacharys, Daya Krishna, Gopinath Bhattacharya; comparative study of philosophical system such as Madhyastha Darshan.

OUTCOME OF THE COURSE:

Students will develop strong natural familiarity with humanities along with right Understanding enabling them to eliminate conflict and strife in the individual and society. Students shall be able to relate philosophy to literature, culture, society and lived experience can be considered.

Course Code:BTES302-18	Course Title: Digital Electronics Lab	0L:0T:2P	1Credits

List of Experiments:

1. To verify the Truth-tables of all logic gates.
2. To realize and verify the Half & full adder circuits using logic gates.
3. To realize Half & full subtract or circuits using logic gates.
4. To realize Encoder and Decoder circuits
5. To realize Multiplexer circuits
6. To realize 4-bit binary-gray & gray-binary converters.
7. To realize comparator circuit for two binary numbers of 2-bit each.
8. To realize Full adder & full subtractor circuits using encoder.
9. To design Full adder & full subtractor circuits using multiplexer.
10. To design and verify the Truth tables of all flip-flops.
11. To design Mod-6/Mod-9 synchronous up-down counter.

Course Outcomes

At the end of this course student will demonstrate the ability to:

1. Realize combinational circuits using logic gates.
2. Realize sequential circuits using logic gates.
3. Realize various types of Flip-flops and counters

Course Code: BTCS 303-19	Course Title: Data Structure & Algorithms Lab	0L:0T:4P	2Credits

List of Experiment:

Task1: Write a program to insert an element at end as well as at a given position in an array.

Task2: Write a program to delete an element from a given whose value is given or whose position is given.

Task3: Write a program to find the location of a given element using Linear Search.

Task4: Write a program to find the location of a given element using Binary Search.

Task 5: Write a program to implement push and pop operations on a stack using linear array.

Task 6: Write a program to convert an infix expression to a postfix expression using stacks.

Task 7: Write a program to evaluate a postfix expression using stacks.

Task 8: Write a recursive function for Tower of Hanoi problem.

Task9: Write a program to implement insertion and deletion operations in a queue using linear array.

Task10: Write a menu driven program to perform following insertion operations in a single linked list:
   i. Insertion at beginning
   ii. Insertion at end
   iii. Insertion after a given node
   iv. Traversing a linked list

Task11: Write a menu driven program to perform following deletion operations in a single linked list:
   i. Deletion at beginning
   ii. Deletion at end
   iii. Deletion after a given node

Task 12: Write a program to implement push and pop operations on a stack using linked list.

Task 13: Write a program to implement push and pop operations on a queue using linked list.

Task14: Program to sort an array of integers in ascending order using bubble sort.

Task15: Program to sort an array of integers in ascending order using selection sort.

Task 16: Program to sort an array of integers in ascending order using insertion sort.

Task17: Program to sort an array of integers in ascending order using quick sort.

Task 18: Program to traverse a Binary search tree in Pre-order, In-order and Post-order.

Task 19: Program to traverse graphs using BFS.

Task 20: Program to traverse graphs using DFS.

Lab Outcomes:

The student will be able to:

1. Improve practical skills in designing and implementing basic linear data structure algorithms;
2. Improve practical skills in designing and implementing Non-linear data structure algorithms;
3. Use Linear and Non-Linear data structures to solve relevant problems;
4. Choose appropriate Data Structure as applied to specific problem definition; &
5. Implement Various searching algorithms and become familiar with their design methods.

Course Code: BTCS 304-18	**Course Title:** Object Oriented Programming Lab	**0L:0T:4P**	**2Credits**

List of Experiment:

1. [Classes and Objects] Write a program that uses a class where the member functions are defined inside a class.
2. [Classes and Objects] Write a program that uses a class where the member functions are defined outside a class.
3. [Classes and Objects] Write a program to demonstrate the use of static data members.
4. [Classes and Objects] Write a program to demonstrate the use of const data members.
5. [Constructors and Destructors] Write a program to demonstrate the use of zero argument and parameterized constructors.
6. [Constructors and Destructors] Write a program to demonstrate the use of dynamic constructor.
7. [Constructors and Destructors] Write a program to demonstrate the use of explicit constructor.
8. [Initializer Lists] Write a program to demonstrate the use of initializer list.
9. [Operator Overloading] Write a program to demonstrate the overloading of increment and decrement operators.
10. [Operator Overloading] Write a program to demonstrate the overloading of binary arithmetic operators.
11. [Operator Overloading] Write a program to demonstrate the overloading of memory management operators.
12. [Typecasting] Write a program to demonstrate the typecasting of basic type to class type.
13. [Typecasting] Write a program to demonstrate the typecasting of class type to basic type.
14. [Typecasting] Write a program to demonstrate the typecasting of class type to class type.
15. [Inheritance] Write a program to demonstrate the multilevel inheritance.
16. [Inheritance] Write a program to demonstrate the multiple inheritance.
17. [Inheritance] Write a program to demonstrate the virtual derivation of a class.
18. [Polymorphism] Write a program to demonstrate the runtime polymorphism.
19. [Exception Handling] Write a program to demonstrate the exception handling.
20. [Templates and Generic Programming] Write a program to demonstrate the use of function template.
21. [Templates and Generic Programming] Write a program to demonstrate the use of class template.
22. [File Handling] Write a program to copy the contents of a file to another file byte by byte. The name of the source file and destination file should be taken as command-line arguments,
23. [File Handling] Write a program to demonstrate the reading and writing of mixed type of data.
24. [File Handling] Write a program to demonstrate the reading and writing of objects.

Fourth Semester

Course Code: BTCS401-18	Course Title: Discrete Mathematics	L:3;T:1; P:0	4Credits

Objectives of the course:

Throughout the course, students will be expected to demonstrate their understanding of Discrete Mathematics by being able to do each of the following:

1. Use mathematically correct terminology and notation.
2. Construct correct direct and indirect proofs.
3. Use division into cases in a proof.
4. Use counterexamples.
5. Apply logical reasoning to solve a variety of problems.

Detailed contents:

Module 1:

Sets, Relation and Function: Operations and Laws of Sets, Cartesian Products, Binary Relation, Partial Ordering Relation, Equivalence Relation, Image of a Set, Sum and Product of Functions, Bijective functions, Inverse and Composite Function, Size of a Set, Finite and infinite Sets, Countable and uncountable Sets, Cantor's diagonal argument and The Power Set theorem, Schroeder-Bernstein theorem. Principles of Mathematical Induction: The Well-Ordering Principle, Recursive definition, The Division algorithm: Prime Numbers, The Greatest Common Divisor: Euclidean Algorithm, The Fundamental Theorem of Arithmetic.

Module 2:

Basic counting techniques-inclusion and exclusion, pigeon-hole principle, permutation and combination.

Module 3:

Propositional Logic: Syntax, Semantics, Validity and Satisfiability, Basic Connectives and Truth Tables, Logical Equivalence: The Laws of Logic, Logical Implication, Rules of Inference, The use of Quantifiers. Proof Techniques: Some Terminology, Proof Methods and Strategies, Forward Proof, Proof by Contradiction, Proof by Contraposition, Proof of Necessity and Sufficiency.

Module 4:

Algebraic Structures and Morphism: Algebraic Structures with one Binary Operation, Semi Groups, Monoids, Groups, Congruence Relation and Quotient Structures, Free and Cyclic Monoids and Groups, Permutation Groups, Substructures, Normal Subgroups, Algebraic Structures with two Binary Operation, Rings, Integral Domain and Fields. Boolean Algebra and Boolean Ring, Identities of Boolean Algebra, Duality, Representation of Boolean Function, Disjunctive and Conjunctive Normal Form

Module 5:

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

Suggested books:

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

Suggested reference books:

1. J.P. Tremblay and R. Manohar, **Discrete Mathematical Structure and It's Application to Computer Science**", TMG Edition, TataMcgraw-Hill
2. Norman L. Biggs, **Discrete Mathematics**, 2nd Edition, Oxford University Press. Schaum's Outlines Series,

Seymour Lipschutz, Marc Lipson, 3. Discrete Mathematics, Tata McGraw - Hill

Course Outcomes:

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

Course Code: BTES402- 18	Course Title: Computer Organization and Architecture	3L:1T:0P	3Credits

Pre-requisites: Digital Electronics

Detailed Contents:

Module1: Functional blocks of a computer
CPU, memory, input-output subsystems, control unit. Instruction set architecture of a CPU– registers, instruction execution cycle, RTL interpretation of instructions, addressing modes, instruction set. Case study– instruction set of8085 processor.
Data representation: signed number representation, fixed and floating point representations, character representation. Computer arithmetic–integer addition and subtraction, ripple carry adder, carry look-ahead adder, etc. multiplication– shift-and add, Booth multiplier, carry save multiplier, etc. Division restoring and non-restoring techniques, floating point arithmetic.

[10 hrs] (CO1, CO2)

Module2: Introduction to x86 architecture.
CPU control unit design: Hardwired and micro-programmed design approaches, Case study– design of a simple hypothetical CPU.
Memory system design: semiconductor memory technologies, memory organization. Peripheral devices and their characteristics: Input-output subsystems, I/O device interface, I/O transfers– program controlled, interrupt driven and DMA, privileged and non-privileged instructions, software interrupts and exceptions. Programs and processes–role of interrupts in process state transitions, I/O device interfaces–SCII, USB.

[12 hrs] (CO2, CO4)

Module3: Pipelining
Basic concepts of pipelining, throughput and speed up, pipeline hazards.
Parallel Processors: Introduction to parallel processors, Concurrent access to memory and cache coherency.

[10 hrs] (CO5)

Module4: Memory Organization
Memory interleaving, concept of hierarchical memory organization, cache memory, cache size vs. block size, mapping functions, replacement algorithms, write policies.

[10 hrs] (CO3)

Course Outcomes:

The student will be able to:

1. Understand functional block diagram of microprocessor;
2. Apply instruction set for Writing assembly language programs;
3. Design a memory module and analyzeits operation by interfacing with the CPU;
4. Classify hardwired and microprogrammed control units; &
5. Understand the concept of pipelining and its performance metrics.

Suggested Books:

1. "Computer Organization and Architecture", Moris Mano,

2. "Computer Organization and Design: The Hardware/Software Interface",5th Edition by David A. Patterson and John L. Hennessy, Elsevier.

3. "Computer Organization and Embedded Systems", 6th Edition by Carl Hamacher, McGraw Hill Higher Education.

Reference Books:

1. "Computer Architecture and Organization", 3rd Edition by John P. Hayes, WCB/McGraw-Hill

2. "Computer Organization and Architecture: Designing for Performance", 10th Edition by William Stallings, Pearson Education.

3. "Computer System Design and Architecture", 2nd Edition by Vincent P. Heuring and Harry F. Jordan, Pearson Education.

CourseCode:BTCS402-19	Course Title: Operating Systems	3L:1T:0P	3Credits

Detailed Contents:

Module1: Introduction
Concept of Operating Systems, Generations of Operating systems, Types of Operating Systems, OS Services, System Calls, Structure of an OS-Layered, Monolithic, Microkernel Operating Systems, Concept of Virtual Machine. Case study on UNIX and WINDOWS Operating System.

[6 hrs] (CO1)

Module2: Processes
Definition, Process Relationship, Different states of a Process, Process State transitions, Process Control Block (PCB), Context switching
Thread: Definition, Various states, Benefits of threads, Types of threads, Concept of multithreads, Process Scheduling: Foundation and Scheduling objectives, Types of Schedulers, Scheduling criteria: CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time; Scheduling algorithms: Pre-emptive and Non-pre-emptive, FCFS, SJF, RR; Multiprocessor scheduling: Real Time scheduling: RM and EDF.

[10 hrs] (CO2, CO3)

Module3: Inter-process Communication
Critical Section, Race Conditions, Mutual Exclusion, Hardware Solution, Strict Alternation, Peterson's Solution, The Producer\Consumer Problem, Semaphores, Event Counters, Monitors, Message Passing, Classical IPC Problems: Reader's & Writer Problem, Dinning Philosopher Problem etc.

[8 hrs] (CO2)

Module4: Deadlocks
Definition, Necessary and sufficient conditions for Deadlock, Deadlock Prevention, Deadlock Avoidance: Banker's algorithm, Deadlock detection and Recovery.

[8 hrs] (CO3)

Module5: Memory Management
Basic concept, Logical and Physical address map, Memory allocation: Contiguous Memory allocation –Fixed and variable partition– Internal and External fragmentation and Compaction; Paging: Principle of operation– Page allocation–Hardware support for paging, Protection and sharing, Disadvantages of paging.
Virtual Memory: Basics of Virtual Memory– Hardware and control structures–Locality of reference, Page fault, Working Set, Dirty page/ Dirty bit–Demand paging, Page Replacement algorithms: Optimal, First in First Out (FIFO), Second Chance(SC), Not recently used (NRU) and Least Recently used(LRU).

[10 hrs] (CO4)

Module6: I/O Hardware
I/O devices, Device controllers, Direct memory access Principles of I/O Software: Goals of Interrupt handlers, Device drivers, Device independent I/O software, Secondary-Storage Structure: Disk structure, Disk scheduling algorithms

File Management: Concept of File, Access methods, File types, File operation, Directory structure, File System structure, Allocation methods (contiguous, linked, indexed), Free Space Management (bit vector, linked list, grouping), directory implementation (linear list, hash table), efficiency and performance.

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

[8 hrs] (CO5, CO6)

Course Outcomes:

The student will be able to:

1. Explain basic operating system concepts such as overall architecture, system calls, user mode and kernel mode;
2. Distinguish concepts related to processes, threads, process scheduling, race conditions and critical sections;
3. Analyze and apply CPU scheduling algorithms, deadlock detection and prevention algorithms;
4. Examine and categorize various memory management techniques like caching, paging, segmentation, virtual memory, and thrashing;
5. Design and implement file management system; &
6. Appraise high-level operating systems concepts such as file systems, disk-scheduling algorithms and various file systems.

Suggested Books:

1. Operating System Concepts Essentials, 9th Edition by Avi Silberschatz, Peter Galvin, Greg Gagne, Wiley Asia Student Edition.
2. Operating Systems: Internals and Design Principles, 5th Edition, William Stallings, Prentice Hall of India.

Reference Books:

1. Operating System: A Design-oriented Approach, 1st Edition by Charles Crowley, Irwin Publishing
2. Operating Systems: A Modern Perspective, 2nd Edition by Gary J. Nutt, Addison-Wesley
3. Design of the Unix Operating Systems, 8th Edition by Maurice Bach, Prentice-Hall of India
4. Understanding the Linux Kernel, 3rd Edition, Daniel P. Bovet, Marco Cesati, O'Reilly and Associates

----

Course Code: BTCS403- 18	Course Title: Design and Analysis of Algorithms	3L:1T:0P	3Credits

Pre-requisites: Data Structures

Detailed Contents:

Module1: Introduction

Characteristics of algorithm. Analysis of algorithm: Asymptotic analysis of complexity bounds– best, average and worst-case behaviour; Performance measurements of Algorithm, Time and space trade-offs, Analysis of recursive algorithms through recurrence relations: Substitution method, Recursion tree method and Masters' theorem.

[8 hrs] (CO1)

Module 2: Fundamental Algorithmic Strategies

Brute-Force, Greedy, Dynamic Programming, Branch- and-Bound and Backtracking methodologies for the design of algorithms; Illustrations of these techniques for Problem-Solving: Bin Packing, Knap Sack, TSP.

[10 hrs] (CO1, CO2)

Module 3: Graph and Tree Algorithms

Traversal algorithms: Depth First Search (DFS) and Breadth First Search (BFS); Shortest path algorithms, Transitive closure, Minimum Spanning Tree, Topological sorting, Network Flow Algorithm.

[10 hrs] (CO3)

Module 4: Tractable and Intractable Problems

Computability of Algorithms, Computability classes – P, NP, NP-complete and NP-hard. Cook's theorem, Standard NP-complete p r o b l e m s and Reduction techniques.

[8 hrs] (CO5)

Module 5: Advanced Topics

Approximation algorithms, Randomized algorithms, Heuristics and their characteristics.

[6 hrs] (CO1, CO4, CO5)

Course Outcomes:

The student will be able to:
1. For a given algorithms analyze worst-case running times of algorithms based on asymptotic analysis and justify the correctness of algorithms;
2. Explain when an algorithmic design situation calls for which design paradigm (greedy/ divide and conquer/backtrack etc.);
3. Explain model for a given engineering problem, using tree or graph, and write the corresponding algorithm to solve the problems;
4. Demonstrate the ways to analyze approximation/randomized algorithms (expected running time, probability of error); &
5. Examine the necessity for NP class based problems and explain the use of heuristic techniques.

Suggested Books:

1. Analysis and Design of Algorithms: A Beginner's Approach Paperback – 1 January 2015 by Rajesh K. Shukla
2. Introduction to Algorithms, 4TH Edition, Thomas H Cormen, Charles ELieserson, Ronald L Rivest and Clifford Stein, MIT Press/McGraw-Hill.
3. Data Structures and Algorithms in C++, Weiss, 4th edition, Pearson.
4. Fundamentals of Computer Algorithms– E. Horowitz, Sartaj Saini, Galgota Publications.

Reference Books

1. Algorithm Design, 1st Edition, Jon Kleinberg and Éva Tardos, Pearson.
2. Algorithm Design: Foundations, Analysis, and Internet Examples, Second Edition, Michael T Goodrich and Roberto Tamassia, Wiley.
3. Algorithms—A Creative Approach, 3RD Edition, Udi Manber, Addison-Wesley, Reading, MA.

Course Code: HSMC122-18	Course Title: Universal Human Values 2: Understanding Harmony	2L:1T:0P	3Credits

COURSE TOPICS:

The course has 28 lectures and 14 practice sessions in 5 modules:

Module1: Course Introduction- Need, Basic Guidelines, Content and Process for Value Education
1. Purpose and motivation for the course, recapitulation from Universal Human Values-I
2. Self-Exploration–what is it? – Its content and process; 'Natural Acceptance' and Experiential Validation- as the process for self-exploration.
3. Continuous Happiness and Prosperity- A look at basic Human Aspirations
4. Right understanding, Relationship and Physical Facility- the basic requirements for fulfilment of aspirations of every human being with their correct priority

5. Understanding Happiness and Prosperity correctly-A critical appraisal of the Current scenario.
6. Method to fulfil the above human aspirations: understanding and living in harmony at various levels. Include practice sessions to discuss natural acceptance in human being as the innate acceptance for living with responsibility (living in relationship, harmony and co-existence) rather than as arbitrariness in choice based on liking-disliking.

Module2: Understanding Harmony in the Human Being- Harmony in Myself!

1. Understanding human being as a co-existence of the sentient 'I' and the material 'Body'
2. Understanding the needs of Self ('I') and 'Body'-happiness and physical facility
3. Understanding the Body as an instrument of 'I' (I being the doer, seer and enjoyer)
4. Understanding the characteristics and activities of 'I' and harmony in 'I'
5. Understanding the harmony of I with the Body: Sanyam and Health; correct appraisal of Physical needs, meaning of Prosperity in detail
6. Programs to ensure Sanyam and Health.
7. Include practice sessions to discuss the role other shave played in making material goods available tome. Identifying from one's own life. Differentiate between
8. Prosperity and accumulation. Discuss program for ensuring health vs dealing with disease.

Module3: Understanding Harmony in the Family and Society- Harmony in Human- Human Relationship

1. Understanding values in human- human relationship; meaning of Justice (nine Universal values in relationships) and program for its fulfilment to ensure mutual happiness; Trust and Respect as the foundational values of relationship.
2. Understanding t h e meaning of Trust; Difference between intention and competence
3. Understanding t h e meaning of Respect, Difference between respect and differentiation; the other salient values in relationship.
4. Understanding the harmony in the society (society being an extension of family): Resolution,
   Prosperity, fearlessness (trust) and co-existence as comprehensive Human Goals.
5. Visualizing a universal harmonious order in society- Undivided Society, Universal Order-from family to world family.
   Include practice sessions to reflect on relationships in family, hostel and institute as extended family, real life examples, teacher-student relationship, goal of education etc. Gratitude as a universal value in relationships. Discuss with scenarios. Elicit examples from students' lives.

Module 4: Understanding Harmony in the Nature and Existence - Whole existence as Coexistence

1. Understanding the harmony in the Nature
2. Interconnectedness and mutual fulfilment among the four orders of nature- recyclability and self- regulation in nature
3. Understanding Existence as Co-existence of mutually interacting units in all- pervasive space
4. Holistic perception of harmony at all levels of existence.
   Include practice sessions to discuss human being as cause of imbalance in nature (film "Home" can be used), pollution, depletion of resources and role of technology etc.

Module5: Implications of the above Holistic Understanding of Harmony on Professional Ethics

1. Natural acceptance of human values
2. Definitiveness of Ethical Human Conduct
3. Basis for Humanistic Education, Humanistic Constitution and Humanistic Universal Order
4. Competence in professional ethics:
   a. Ability to utilize the professional competence for augmenting universal human order
   b. Ability to identify the scope and characteristics of people friendly and eco-friendly production systems,
   c. Ability to identify and develop appropriate technologies and management patterns for above production systems.

5. Case studies of typical holistic technologies, management models and production systems.
6. Strategy for transition from the present state to Universal Human Order:

    a. At the level of individual: as socially and ecologically responsible engineers, technologists and managers

    b. At the level of society: as mutually enriching institutions and organizations.

7. Sum up. Include practice Exercises and Case Studies will be taken up in Practice (tutorial) Sessions eg. To discuss the conduct as an engineer or scientist etc.

3. READINGS:

3.1 Text Book

1. Human Values and Professional Ethics by R R Gaur, R Sangal, G P Bagaria, Excel Books, New Delhi, 2010.

3.2 Reference Books

1. Jeevan Vidya: Ek Parichaya, A. Nagaraj, Jeevan Vidya Prakashan, Amarkantak, 1999.
2. Human Values, A.N.Tripathi, New Age International Publishers, New Delhi, 2004.
3. The Story of Stuff (Book).
4. The Story of My Experiments with Truth-by Mohan das Karam chand Gandhi
6. Small is Beautiful-E. F Schumacher.
7. Slow is Beautiful-Cecile Andrews
8. Economy of Permanence- JC Kumarappa
9. Bharat Mein Angreji Raj-Pandit Sunder lal
10. Re discovering India –by Dharampal
11. Hind Swarajor Indian Home Rule-by Mohan das K. Gandhi
12. India Wins Freedom- Maulana Abdul Kalam Azad
13. Vivekananda-Romain Rolland (English)
14. Gandhi-Romain Rolland (English)

Outcome of the Course:

By the end of the course, students are expected to become more aware of themselves, and their surroundings (family, society, nature); they would become more responsible in life, and in handling problems with sustainable solutions, while keeping human relationships and human nature in mind. They would have better critical ability. They would also become sensitive to their commitment towards what they have understood (human values, human relationship and human society). It is hoped that they would be able to apply what they have learnt to their own self in different day-to-day settings in real life, at least a beginning would be made in this direction.

This is only an introductory foundational input. It would be desirable to follow it up by

    a) Faculty-student or mentor-mentee programs throughout their time with the institution.

    b) Higher level courses on human values in every aspect of living. E.g. as a professional.

Course Code: EVS101-18	Course Title: Environmental Studies	1L:0T:0P	0Credits

COURSE TOPICS:

Detailed Contents:

Module1: Natural Resources:

Renewable and non-renewable resources Natural resources and associated problems.

    a) Forest resources: Use and over-exploitation, deforestation, case studies. Timber extraction, mining, dams and their effects on forest and tribal people.

    b) Water resources: Use and over-utilization of surface and groundwater, floods, drought, conflicts over water, dams-benefits and problems.

    c) Mineral resources: Use and exploitation, environmental effects of extracting and using mineral resources, case studies.

d) Food resources: World food problems, changes caused by agriculture and overgrazing, effects of modern agriculture, fertilizer-pesticide problems, waterlogging, salinity, case studies.

e) Energy resources: Growing energy needs, renewable and nonrenewable energy sources, use of alternate energy sources. Case studies.

f) Land resources: Landas are source, land degradation, man induced landslides, soil erosion and desertification.
   • Role of an individual in conservation of natural resources.
   • Equitable use of resources for sustainable lifestyles.

Module2: Ecosystems

Concept of an ecosystem. Structure and function of an ecosystem. Foodchains, foodwebs and ecological pyramids. Introduction, types, characteristic features, structure and function of following ecosystems:

a. Forest ecosystem
b. Aquatic ecosystems (ponds, streams, lakes, rivers, oceans, estuaries)

Module3: Biodiversity and its conservation

• Introduction–Definition: genetic, species and ecosystem diversity.
• Biodiversity at global, National and local levels.
• Inida as a mega-diversity nation
• Hot-sports of biodiversity.
• Threats to biodiversity: habitat loss, poaching of wildlife, man-wild life conflicts.
• Endangered and endemic species of India

Module4: Social Issues and the Environment

• From Unsustainable to Sustainable development
• Resettlement and rehabilitation of people; its problems and concerns.
• Environmental ethics: Issues and possible solutions.
•   Climate change, global warming, acid rain, ozone layer depletion, Nuclear accidents and holocaust.

Case Studies.

• Public awareness.

*ACTIVITIES

Nature club (birdwatching, recognizing plants at institute/at home, recognizing local animals, appreciating biodiversity

Impart knowledge and inculcate the habit of taking interest and understanding biodiversity in and around the college campus. The students should have been courage to take interest in bird watching, recognizing local plants, herbs and local animals. The students should be encouraged to appreciate the difference in the local biodiversity in their hometown, in the place of their study and other places they visit for vacation/breaks etc.

Following activities must be included. Identify a tree fruit flower peculiar to a place or having origin from the place.

Making high resolution big photographs of small creatures (bees, spiders, ants. mosquitos etc.) especially part of body so that people can recognize (games on recognizing animals/plants).

Videography/photography/information collections on specialties/unique features of different types of common creatures. Search and explore patents and rights related to animals, trees etc. Studying miracles of mechanisms of different body systems.

1(A)Awareness Activities:

a)   Small group meetings about water management, promotion of recycle use, generation of less waste, avoiding electricity waste

b)   Slogan making event

c)   Poster making event

d)   Cycle rally

e) Lectures from experts

f) Plantation

g) Gifting a tree to see its full growth

h) Cleanliness drive

i) Drive for segregation of waste

a) To live with some eminent environment a list for a week or so to understand his work

b) To work in kitchen garden for mess

c) To know about the different varieties of plants

d) Shutting down the fans and ACs of the campus for an hour or so

e) Visit to a local area to document environmental assets river/forest/grassland/hill/mountain/lake/Estuary/Wetlands

f) Visit to a local polluted Site-Urban/Rural/Industrial/Agricultural n) Visit to a Wild life sanctuary, National Park or Biosphere Reserve

Suggested Readings

1. Agarwal, K.C.2001 Environmental Biology, Nidi Publ. Ltd. Bikaner.

2. Bharucha Erach, The Biodiversity of India, Map in Publishing Pvt. Ltd.,Ahmedabad – 380013, India, Email: mapin@icenet.net(R)

3. BrunnerR.C., 1989, Hazardous Waste Incineration, McGrawHillInc.480p

4. Clark R.S., Marine Pollution, Clanderson Press Oxford(TB)

5. Cunningham, W.P. Cooper, T.H. Gorhani, E & Hepworth, M.T. 2001, Environmental Encyclopedia, Jaico Publ. House, Mumabai,1196p

6. Hawkins R.E., Encyclopedia of Indian Natural History, Bombay Natural History Society, Bombay(R)

7. Heywood, V.H & Waston, R.T.1995.Global Biodiversity Assessment. Cambridge Univ.Press1140p.

8. Mhaskar A.K., Matter Hazardous, Techno-Science Publication(TB)

9. Miller T.G. Jr. Environmental Science, Wadsworth Publishing Co.(TB)

10. Odum, E.P. 1971. Fundamentals of Ecology.W.B. Saunders Co. USA, 574p.

11. Townsend C., Harper J, and Michael Begon, Essentials of Ecology, Blackwell Science(TB).

12. Trivedi R.K., Handbook of Environmental Laws, Rules Guidelines, Compliances and Standards, Vol I and II, Enviro Media(R).

13. Trivedi R.K. and P.K. Goel, Introduction to air pollution, Techno-Science Publication(TB).

14. Wanger K.D., 1998 Environmental Management. W.B. Saunders Co. Philadelphia.

Course Code: BTCS402-18	Course Title: Computer Organization & Architecture Lab	0L:0T:2P	1Credits

List of Experiment:

Task 1: Computer Anatomy- Memory, Ports, Mother board and add-on cards.
Task 2: Dismantling and assembling PC.
Task 3: Introduction to 8085 kit.
Task 4: Addition of two 8 bit numbers, sum 8 bit.
Task 5: Subtraction of two 8 bit numbers.
Task 6: Find 1's complement of 8-bit number.
Task 7: Find 2's complement of 8-bit number.
Task 8: Shift an 8-bit no. by one bit.
Task 9: Find Largest of two 8 bit numbers.
Task 10: Find Largest among an array often numbers (8 bit).

Task 11:   Sum of series of 8 bit numbers.
Task 12:   Introduction to 8086 kit.
Task 13:   Addition and subtraction of two 16 bit numbers, sum 16 bit.
Task 14:   Implement of Booth's algorithm for arithmetic operations.
Task 15:   Find 1's and 2'scomplement of 16-bit number.
Task 16:   Implement simple programs using I/O based interface.

Lab Outcomes:

The student will be able to:

1. Assemble personal computer;
2. Implement the various assembly language programs for basic arithmetic and logical operations; &
3. Demonstrate the functioning of microprocessor/microcontroller based systems with I/O interface.

Reference Books:

1. Fundamentals of Microprocessors and Microcontrollers by B. Ram, Dhanpat Rai.

Course Code: BTCS 404-18	Course Title: Operating Systems Lab	0L:0T:4P	2Credits

List of Experiment:

Task1: Installation Process of various operating systems.

Task 2: Implementation of CPU scheduling algorithms to find turnaround time and waiting time.
       a)   FCFS
       b)   SJF
       c)   Round Robin (pre-emptive)
       d)   Priority.
Task 3: Virtualization, Installation of Virtual Machine Software and installation of Operating System on Virtual Machine.

Task 4: Commands for files & directories: cd, ls, cp, md, rm, mkdir, rmdir. Creating and viewing files using cat. File comparisons. Disk related commands: checking disk free spaces. Processes in linux, connecting processes with pipes, background processing, managing multiple processes. Background process: changing process priority, scheduling of processes at command, batch commands, kill, ps, who, sleep. Printing commands, grep, fgrep, find, sort, cal, banner, touch, file. File related commands ws, sat, cut, grep.

Task 5: Shell Programming: Basic of shell programming, various types of shell, Shell Programming in bash, conditional & looping statement, case statements, parameter passing and arguments, shell variables, shell keywords, creating shell programs for automate system tasks, report printing.

Task6: Implementation of Bankers algorithm for the purpose of deadlock avoidance.

Lab Outcomes:

The student will be able to:

1. Understand and implement basic services and functionalities of the operating system;
2. Analyze and simulate CPU Scheduling Algorithms like FCFS, Round Robin, SJF, and Priority;
3. Implement commands for files and directories;
4. Understand and implement the concepts of shell programming;
5. Simulate file allocation and organization techniques; &
6. Understand the concepts of deadlock in operating systems and implement them in multiprogramming system.

Reference Books:

1. Operating Systems: Design and Implementation, Albert S.Woodhulland Andrew S. Tanenbaum, Pearson Education.

Course Code: BTCS 405- 18	Course Title: Design and Analysis of Algorithms Lab	0L:0T:4P	2Credit

List of Experiment:

Task1: Code and analyze solutions to following problem with given strategies:

    i.   Knap Sack using greedy approach
    ii.   Knap Sack using dynamic approach

Task2: Code and analyze to find an optimal solution to matrix chain multiplication using dynamic programming.

Task3: Code and analyze to find an optimal solution to TSP using dynamic programming.

Task4: Implementing an application of DFS such as:

    i.   to find the topological sort of a directed acyclic graph
    ii.   to find a path from source to goal in amaze.

Task5: Implement an application of BFS such as:

    i.   to find connected components of an undirected graph
    ii.   to check whether a given graph is bipartite.

Task6: Code and analyze to find shortest paths in a graph with positive edge weights using Dijkstra's algorithm.

Task7: Code and analyze to find shortest paths in a graph with arbitrary edge weights using Bellman- Ford algorithm.

Task8: Code and analyze to find shortest paths in a graph with arbitrary edge weights using Flyods' algorithm.

Task9: Code and analyze to find the minimum spanning tree in a weighted, undirected graph using Prims' algorithm

Task10: Code and analyze to find the minimum spanning tree in a weighted, undirected graph using Kruskals' algorithm.

Task11: Coding any real world problem or TSP algorithm using any heuristic technique.

Lab Outcomes:

The student will be able to:
1. Improve practical skills in designing and implementing complex problems with different techniques;
2. Understand comparative performance of strategies and hence choose appropriate,to apply to specific problem definition;
3. Implement Various tree and graph based algorithms and become familiar with their design methods; &
4. Design and Implement heuristics for real world problems.

Reference Books

1. Data Structures and Algorithms in C++, Weiss, 4[th]edition, Pearson
2. Data Structures and Algorithms using Python and C++, David M. Reed and John Zelle, 2009 edition (available as ebook), Franklin Beedle & Associates.

Fifth Semester

BTES501-18	Enterprise Resource Planning	3L:0T:0P	3 Credits

Course Details:

UNIT 1 INTRODUCTION
ERP: An Overview, Enterprise – An Overview, Benefits of ERP, ERP and Related Technologies, Business Process Reengineering (BPR), Data Warehousing, Data Mining, OLAP, SCM

[9hrs., CO1]

UNIT II ERP IMPLEMENTATION
ERP Implementation Lifecycle, Implementation Methodology, Hidden Costs, Organizing the Implementation, Vendors, Consultants and Users, Contracts with Vendors, Consultants and Employees, Project Management and Monitoring

[9hrs., CO2]

UNIT III THE BUSINESS MODULES
Business modules in an ERP Package, Finance, Manufacturing, Human Resources, Plant Maintenance, Materials Management, Quality Management, Sales and Distribution

[9hrs., CO3]

UNIT IV THE ERP MARKET
ERP Market Place, SAP AG, Peoplesoft, Baan, JD Edwards, Oracle, QAD, SSA

[9hrs., CO4]

UNIT V ERP – PRESENT AND FUTURE
Turbo Charge the ERP System, EIA, ERP and e-Commerce, ERP and Internet, Future Directions

[6hrs., CO1]

TEXT BOOK
1. Alexis Leon, "ERP Demystified", Tata McGraw Hill, New Delhi, 2000

REFERENCES
1. Joseph A Brady, Ellen F Monk, Bret Wagner, "Concepts in Enterprise Resource Planning", Thompson Course Technology, USA, 2001.

2. Vinod Kumar Garg and Venkitakrishnan N K, "Enterprise Resource Planning – Concepts and Practice", PHI, New Delhi, 2003

Course outcomes: The students at the end will be able;
CO1: To know the basics of ERP
CO2: To understand the key implementation issues of ERP
CO3: To know the business modules of ERP
CO4: To be aware of some popular products in the area of ERP

………...

Course Code: BTCS501-18	Course Title: Database Management Systems	3L:0T:0P	3Credits

Detailed Contents:
Module 1: Database system architecture
Data Abstraction, Data Independence, Data Definition Language (DDL), Data Manipulation Language (DML). Data models: Entity-relationship model, network model, relational and object oriented Data models, integrity constraints, data manipulation operations.

[7hrs] (CO1,2)

Module 2: Relational query languages
Relational algebra, Tuple and domain relational calculus, SQL3, DDL and DML constructs, Open source and Commercial DBMS - MYSQL, ORACLE, DB2, SQL server. Relational database design: Domain and data dependency, Armstrong's axioms, Normal forms, Dependency preservation, Lossless design. Query processing and optimization: Evaluation of relational algebra expressions, Query equivalence, Join strategies, Query optimization algorithms.

[10hrs] (CO2,4)

Module 3:
Storage strategies, Indices, B-trees, hashing.

[3hrs] (CO3)

Module 4: Transaction processing
Concurrency control, ACID property, Serializability of scheduling, Locking and timestamp based schedulers, Multi-version and optimistic Concurrency Control schemes, Database recovery.

[6hrs] (CO3)

Module 5: Database Security
Authentication, Authorization and access control, DAC, MAC and RBAC models, Intrusion detection, SQL injection.

[8hrs] (CO 4,5)

Module 6: Advanced Topics
Object oriented and object relational databases, Logical databases, Web databases, Distributed databases.

[8hrs] (CO 5)

Course Outcomes:
At the end of study the student shall be able to:

CO1: write relational algebra expressions for a query and optimize the Developed expressions

CO2: design the databases using ER method and normalization.

CO3: construct the SQL queries for Open source and Commercial DBMS-MYSQL, ORACLE, and DB2.

CO4: determine the transaction atomicity, consistency, isolation, and durability.

CO5: Implement the isolation property, including locking, time stamping based on concurrency control and Serializability of scheduling.

Text Books:
1. "Database System Concepts", 6th Edition by Abraham Silberschatz, Henry F. Korth, S. Sudarshan, McGraw-Hill.

Reference Books:
1. "Principles of Database and Knowledge–Base Systems", Vol1 by J. D. Ullman, Computer Science Press.
2. "Fundamentals of Database Systems", 5th Edition by R. Elmasri and S. Navathe, Pearson Education.
3. "Foundations of Databases", Reprint by Serge Abiteboul, Richard Hull, Victor Vianu, Addison-Wesley.

Course Code: BTCS502-18	Course Title: Formal Language & Automata Theory	3L:1T:0P	3Credits

Detailed Contents
Module 1:   Introduction
Alphabet, languages and grammars, productions and derivation, Chomsky hierarchy of languages.
[3hrs] (CO1)
Module 2:   Regular languages and finite automata:
Regular expressions and languages, deterministic finite automata (DFA) and equivalence with regular expressions, nondeterministic finite automata (NFA) and equivalence with DFA, regular grammars and equivalence with finite automata, properties of regular languages, pumping lemma for regular languages, minimization of finite automata.   **[8hrs] (CO2)**

Module 3:   Context-free languages and pushdown automata
Context-free grammars (CFG) and languages (CFL), Chomsky and Greibach normal forms, nondeterministic pushdown automata (PDA) and equivalence with CFG, parse trees, ambiguity in CFG, pumping lemma for context-free languages, deterministic pushdown automata, closure properties of CFLs.
    [8hrs] (CO3)

Module 4:   Context-sensitive languages
Context-sensitive grammars (CSG) and languages, linear bounded automata and equivalence with CSG.

    [5hrs] (CO4)

Module 5:   Turing machines
The basic model for Turing machines (TM), Turing recognizable (recursively enumerable) and Turing-decidable (recursive) languages and their closure properties, variants of Turing machines, nondeterministic TMs and equivalence with deterministic TMs, unrestricted grammars and equivalence with Turing machines, TMs as enumerators. **[8hrs] (CO 5)**

Module 6:   Undecidability & Intractablity:
Church-Turing thesis, universal Turing machine, the universal and diagonalization languages, reduction between languages and Rice s theorem, undecidable problems about languages.
Intractablity: Notion of tractability/feasibility. The classes NP and co-NP, their importance. Polynomial time many-one reduction. Completeness under this reduction. Cook-Levin theorem: NP-completeness of propositional satisfiability, other variants of satisfiability. NP-complete problems from other domains:

graphs (clique, vertex cover, independent sets, Hamiltonian cycle), number problem (partition), set cover **[12hrs] (CO5)**

Course Outcomes: The student will be able to:
CO1: Write a formal notation for strings, languages and machines.
CO2: Design finite automata to accept a set of strings of a language.
CO3: Design context free grammars to generate strings of context free language .
CO4: Determine equivalence of languages accepted by Push Down Automata and languages generated by context free grammars
CO5: Distinguish between computability and non-computability and Decidability and undecidability.

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

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

| **Course Code:** BTCS503-18 | **Course Title:** Software Engineering | **3L:1T:0P** | **3Credits** |

Detailed Contents:
Module 1:
Evolution and impact of Software engineering, software life cycle models: Waterfall, prototyping, Evolutionary, and Spiral models. Feasibility study, Functional and Non-functional requirements, Requirements gathering, Requirements analysis and specification.

[10hrs] (CO 1)

Module 2:
Basic issues in software design, modularity, cohesion, coupling and layering, function-oriented software design: DFD and Structure chart, object modeling using UML, Object-oriented software development, user interface design. Coding standards and Code review techniques.

[8hrs] (CO2)
Module 3:
Fundamentals of testing, White-box, and black-box testing, Test coverage analysis and test case design techniques, mutation testing, Static and dynamic analysis, Software reliability metrics, reliability growth modeling.
[10hrs] (CO 3)
Module 4:
Software project management, Project planning and control, cost estimation, project scheduling using PERT and GANTT charts, cost-time relations: Rayleigh-Norden results, quality management

[8hrs] (CO4)
Module 5:

ISO and SEI CMMI, PSP and Six Sigma. Computer aided software engineering, software maintenance, software reuse, Component-based software development.

[6hrs] (CO5)

Text Books:
1. Roger Pressman, "Software Engineering: A Practitioners Approach,(6th Edition), McGraw Hill, 1997.

Reference Books:
1. Sommerville, "Software Engineering, 7th edition", Adison Wesley, 1996.
2. Watts Humphrey, "Managing software process", Pearson education, 2003.
3. James F. Peters and Witold Pedrycz, " Software Engineering – An Engineering Approach", Wiley.
4. Mouratidis and Giorgini. "Integrating Security and Software Engineering–Advances and Future", IGP. ISBN – 1-59904-148-0.

5. Pankaj Jalote, "An integrated approach to Software Engineering", Springer/Narosa.

6. Fundamentals of Software Engineering by Rajib Mall, – PHI-3rd Edition, 2009.

Course Outcomes:
At the end of the course the student should be able to:
CO 1: Students should be able to identify the need for engineering approach to software development and various processes of requirements analysis for software engineering problems.
CO 2: Analyse various software engineering models and apply methods for design and development of software projects.
CO 3: Work with various techniques, metrics and strategies for Testing software projects.
CO 4: Identify and apply the principles, processes and main knowledge areas for Software Project Management
CO 5: Proficiently apply standards, CASE tools and techniques for engineering software projects

----

Course Code: BTCS 504-18	Course Title: Computer Networks	3L:1T:0P	3Credits

Detaile d Contents:

Module 1: Data Communication Components
Representation of data and its flow Networks, Various Connection Topology, Protocols and Standards, OSI model, Transmission Media, LAN: Wired LAN, Wireless LANs, Connecting LAN and Virtual LAN, Techniques for Bandwidth utilization: Multiplexing -   Frequency division, Time division and Wave division, Concepts on spread spectrum.

[8hrs] (CO1)

Module 2: Data Link Layer and Medium Access Sub Layer
Error Detection and Error Correction - Fundamentals, Block coding, Hamming Distance, CRC; Flow Control and Error control protocols - Stop and Wait, Go back – N ARQ, Selective Repeat ARQ, Sliding Window, Piggybacking, Random Access, Multiple access protocols -Pure ALOHA, Slotted ALOHA, CSMA/CDCDMA/CA.

[10 hrs] (CO2)

Module 3: Network Layer
Switching, Logical addressing – IPV4, IPV6; Address mapping – ARP, RARP, BOOTP and DHCP–Delivery, Forwarding and Unicast Routing protocols.

[8 hrs] (CO3)

Module 4: Transport Layer
Process to Process Communication, User Datagram Protocol (UDP), Transmission Control Protocol (TCP), SCTP Congestion Control; Quality of Service, QoS improving techniques: Leaky Bucket and Token Bucket algorithm.

[8 hrs] (CO3)

Module 5:   Application Layer
Domain Name Space (DNS), DDNS, TELNET, EMAIL, File Transfer Protocol (FTP), WWW, HTTP, SNMP, Bluetooth, Firewalls, Basic concepts of Cryptography.

[8 hrs] (CO4)

Course Outcomes: The student will be able to:
CO1: Explain the functions of the different layer of the OSI Protocol;
CO2:. Describe the function of each block of wide-area networks (WANs), local area networks (LANs) and Wireless LANs (WLANs);
CO3: Develop the network programming for a given problem related TCP/IP protocol; &
CO4: Configure DNS DDNS, TELNET, EMAIL, File Transfer Protocol (FTP), WWW, HTTP, SNMP, Bluetooth, Firewalls using open source available software and tools.

Text Books:
1. Data Communication and Networking, 4th Edition, Behrouz A. Forouzan, McGraw- Hill.
2. Data and Computer Communication, 8th Edition, William Stallings, Pearson Prentice Hall India.

Reference Books:
1. Computer Networks, 8th Edition, Andrew S. Tanenbaum, Pearson New International Edition.
2. Internetworking with TCP/IP, Volume 1, 6th Edition Douglas Comer, Prentice Hall of India.
3. TCP/IP Illustrated, Volume 1, W. Richard Stevens, Addison-Wesley, United States of America.

Course Code: BTCS505-18	CourseTitle: Database management System lab	0L:0T:4P	2Credits

List of Experiments:
Task 1: Introduction to SQL and installation of SQL Server / Oracle.
Task 2: Data Types, Creating Tables, Retrieval of Rows using Select Statement, Conditional Retrieval of Rows, Alter and Drop Statements.
Task 3: Working with Null Values, Matching a Pattern from a Table, Ordering the Result of a Query, Aggregate Functions, Grouping the Result of a Query, Update and Delete Statements.
Task 4: Set Operators, Nested Queries, Joins, Sequences.
Task 5: Views, Indexes, Database Security and Privileges: Grant and Revoke Commands, Commit and Rollback Commands.
Task 6: PL/SQL Architecture, Assignments and Expressions, Writing PL/SQL Code, Referencing Non-SQL parameters.
Task 7: Stored Procedures and Exception Handling.
Task 8: Triggers and Cursor Management in PL/SQL.

Suggested Tools – MySQL, DB2, Oracle, SQL Server 2012, Postgre SQL, SQL lite

Course Outcomes:

CO1: This practical will enable students to retrieve data from relational databases using SQL.
CO2: students will be able to implement generation of tables using datatypes
CO3: Students will be able to design and execute the various data manipulation queries.
CO4: Students will also learn to execute triggers, cursors, stored procedures etc.

Course Code: BTCS506-18	Course Title: Software Engineering Lab	0L:0T:2P	1 Credits

List of Exper iments:

Task 1: Study and usage of OpenProj or similar software to draft a project plan
Task 2: Study and usage of OpenProj or similar software to track the progress of a project
Task 3: Preparation of Software Requirement Specification Document, Design Documents and Testing Phase
Task 4: related documents for some problems
Task 5: Preparation of Software Configuration Management and Risk Management related documents
Task 6: Study and usage of any Design phase CASE tool
Task 7: To perform unit testing and integration testing
Task 8: To perform various white box and black box testing techniques
Task 9: Testing of a web site

Suggested Tools - Visual Paradigm, Rational Software Architect. Visio, Argo UML, Rational Application Developer etc. platforms.

Course Code: BTCS507-18	Course Title: Computer Networks Lab	0L:0T:2P	1 Credits

List of Experiments:

Task 1: To study the different types of Network cables and network topologies.
Task 2: Practically implement and test the cross-wired cable and straight through cable using clamping tool and network lab cable tester.
Task 3: Study and familiarization with various network devices.
Task 4: Familiarization with Packet Tracer Simulation tool/any other related tool.
Task 5: Study and Implementation of IP Addressing Schemes
Task 6: Creation of Simple Networking topologies using hubs and switches
Task 7: Simulation of web traffic in Packet Tracer
Task 8: Study and implementation of various router configuration commands
Task 9: Creation of Networks using routers.
Task 10: Configuring networks using the concept of subnetting
Task 11: Practical implementation of basic network command and Network configuration commands

like ping, ipconfig, netstat, tracert etc. for troubleshooting network related problems.

Task 12: Configuration of networks using static and default routes.

Course Outcomes:

The students will be able to:

CO1: Know about the various networking devices, tools and also understand the implementation of network topologies;

CO2: Create various networking cables and know how to test these cables;

CO3: Create and configure networks in packet trace rtool using various network devices and topologies;

CO4: Understand IP addressing and configure networks using the subnet in;

CO5: Configure routers using various router configuration commands.

Suggested Tools **- NS2/3, Cisco packet tracer, Netsim etc.**.

ELECTIVES- I

Course Code: BTCOE 508-18	Course Title: Microprocessor & Assembly Language Programming	3L:0T:0P	3 Credits

Objective/s: The course is intended to give students good understanding of internal architectural details and functioning of microprocessors.

Detailed Contents:

Module 1:
Introduction: Introduction to Microprocessors, history, classification, recent microprocessors. [5]

Module 2:
Microprocessor Architecture: 8085 microprocessor Architecture. Bus structure, I/O, Memory & System buses, concept of address Bus, Data Bus & Control Bus, Synchronous & Asynchronous buses. Instruction execution sequence & Data Flow, Instruction cycle. [5]

Module 3:
I/O memory interface: Data transfer modes: Programmable, interrupt initiated and DMA. Serial & parallel interface, Detail study of 8251 I/O Processor & 8255 programmable peripheral interfaces. [6]

Module 4:
Instruction set & Assembly Languages Programming: Introduction, instruction & data formats, addressing modes, status flags, 8085 instructions, Data transfer operations, Arithmetic operations, Logical operations, Branch operations. [7]

Module 5:
Case structure & Microprocessor application: Interfacing of keyboards and seven segment LED display, Microprocessor controlled temperature system (MCTS), Study of traffic light system, stepper motor controller, Microprocessor based micro computers. [8]

Module 6:
Basic architecture of higher order microprocessors: Basic introduction to 8086 family, motorola 68000, Pentium processors.

Text Books:
1. 8085 Microprocessor by Ramesh Gaonkar, PHI Publications.
2. Daniel Tabak, Advanced Microprocessors, McGraw- Hill, Inc., Second Edition 1995.
3. Douglas V. Hall, Microprocessors and Interfacing: Programming and Hardware, Tata McGraw Hill Edition, 1986.
4. Charles M.Gilmore, Microprocessors: Principles and Applications, McGraw Hill

Course Code: BTCOE 511-18	Course Title: Microprocessor & Assembly Language Programming Lab	0L:0T:2P	1 Credits

List of Experiments:

1. Introduction to 8085 kit.
2. Addition of two 8 bit numbers, sum 8 bit.
3. Subtraction of two 8 bit numbers.
4. Find 1's complement of 8 bit number.
5. Find 2's complement of 8 bit number.
6. Shift an 8 bit no. by one bit.
7. Find Largest of two 8 bit numbers.
8. Find Largest among an array of ten numbers (8 bit).
9. Sum of series of 8 bit numbers.
10. Introduction to 8086 kit.
11. Addition of two 16 bit numbers, sum 16 bit.
12. Subtraction of two 16 bit numbers.
13. Find 1's complement of 16 bit number.
14. Find 2's complement of 16 bit number.
References: Microprocessor by B. Ram, Dhanpat Rai Publications.

Course Code: BTCOE509-	Course Title: Computer Peripherals and Interfaces	3L:0T:0P	3 Credits

OBJECTIVES: To learn the functional and operational details of various peripheral devices.

Detailed Contents:

Module 1:
SYSTEM RESOURCES: Interrupt, DMA Channel, I/O Port Addresses and resolving and resolving

the conflict of resources. I/O buses- ISA, EISA, Local bus, VESA Local bus, PCI bus, PCI Express, Accelerated graphics port bus.

Module 2:
IDE & SCSI Interfaces: IDE origin, IDE Interface ATA standards ATA1 to ATA7. ATA feature, ATA RAID and SCSI RAID, SCSI Cable and pin Connector pin outs SCSI V/s IDE Advantages and limitation.

Module 3:
Video Hardware: Video display technologies, DVI Digital signals for CRT Monitor, LCD Panels, Video adapter types, Integrated Video/ Motherboard chipset, Video RAM, Video driver and multiple Monitor, Graphic accelerators. Advanced 3D Technologies, TV Tuner and Video Capture upgrades troubleshooting Video Cards and Drivers.

Module 4:
I/O Interfaces: I/O Interfaces from USB and IEEE1394, I/O Interface from serial and Parallel to IEEE1394 and USB 961, Parallel to SCSI converter. Testing of serial and parallel port, USB Mouse/ Keyboard Interfaces.

Module 5:
Input/ Output Driver software aspects: Role of device driver DOS and UNIX/ LINUX device drivers.

Module 6:
Design & Integration of Peripheral devices to a computer system as a Case Study

Module 7:
Future Trends: Detailed Analysis of recent Progress in the Peripheral and Bus systems. Some aspects of cost Performance analysis while designing the system

Text Books
1. Douglas V. Hall ,"Microprocessors and Interfacing", Tata McGraw Hill 2006.
2. Barry B. Brey & C.R.Sarma" The intel microprocessors," Pearson 2003.
3. P. Pal Chandhari , "Computer Organization and design" Prentice Hall of India Pvt. Ltd, 1994. 4. Del Corso, H.Kirrman, JD Nicond "Microcomputer buses & links" Academic Press 1986.

Course Code: BTCOE512-18	Course Title: Computer Peripherals and Interfaces Lab	0L:0T:2P	1 Credits	List of

Experiments:

1. To study the construction and working of CRT, LCD, LED (coloured and black and white monitor) and its troubleshooting .
2. To Study the components and internal parts, working of hard disk and CDROM, DVD, Flash Drives
3. To study the operations and components and internal parts of Key Board, mouse and their troubleshooting
4. Study of components and internal parts and working of DMP, Inkjet printer and Laser printer and various installation of printers
5. To study the SMPS circuit and measure its various voltages. Connecting SMPS to motherboard and other devices.

6. Study the operation and maintenance of UPS.
7. Exercise on assembling a PC with peripherals and testing the same.
8. Setup and configuration of ROM BIOS
9. Visit to nearby industry

Course Code: BTCOE 510-18	Course Title: Graph Theory	3L:0T:0P	3 Credits

Detailed Contents:

Unit-1
Introduction to Graph Theory   **7 hours**
Definitions and Examples, Subgraphs, Complements, and Graph Isomorphism, Vertex Degree, Euler Trails and Circuits

Unit-2
Introduction to Graph Theory contd   **6 hours**
Planar Graphs, Hamilton Paths and Cycles, Graph Colouring, and Chromatic Polynomials

Unit-3
Trees   **6 hours**
Definitions, Properties, and Examples, Routed Trees, Trees and Sorting, Weighted Trees and Prefix Codes

Unit-4
Optimization and Matching   **7 hours**
Dijkstra's Shortest Path Algorithm, Minimal Spanning Trees – The algorithms of Kruskal and Prim, Transport Networks – Max-flow, Min-cut Theorem, Matching Theory

Unit-5
Fundamental Principles of Counting   **6 hours**
The Rules of Sum and Product, Permutations, Combinations – The Binomial Theorem, Combinations with Repetition, The Catalon Numbers
Unit-6
The Principle of Inclusion and Exclusion   **6 hours**
The Principle of Inclusion and Exclusion, Generalizations of the Principle, Derangements – Nothing is in its Right Place, Rook Polynomials
Unit-7
Generating Functions   **7 hours**
Introductory Examples, Definition and Examples – Calculational Techniques, Partitions of Integers, the Exponential Generating Function, the Summation Operator

Unit-8
Recurrence Relations 7 hours
First Order Linear Recurrence Relation, The Second Order Linear Homogeneous Recurrence Relation with Constant Coefficients, The Non-homogeneous Recurrence Relation, The Method of Generating Functions

Course Code: BTCOE 513-18	**Course Title:** Graph Theory Lab	**0L:0T:4P**	**2 Credits**

List of Experiments:

The tasks can be assigned as per the instructor.

Course Code: BTCS 510-18	**Course Title: Programming in Python**	**3L:0T:0P**	**3 Credits**

Detailed Syllabus:
UNIT - I Python Basics, Objects- Python Objects, Standard Types, Other Built-in Types, Internal Types, Standard Type Operators, Standard Type Built-in Functions, Categorizing the Standard Types, Unsupported Types Numbers - Introduction to Numbers, Integers, Floating Point Real Numbers, Complex Numbers, Operators, Built-in Functions, Related Modules Sequences - Strings, Lists, and Tuples, Mapping and Set Types

CO 1,2
UNIT - II FILES: File Objects, File Built-in Function [open()], File Built-in Methods, File Built-in Attributes, Standard Files, Command-line Arguments, File System, File Execution, Persistent Storage Modules, Related Modules Exceptions: Exceptions in Python, Detecting and Handling Exceptions, Context Management, *Exceptions as Strings, Raising Exceptions, Assertions, Standard Exceptions, *Creating Exceptions, Why Exceptions (Now)?, Why Exceptions at All?, Exceptions and the sys Module, Related Modules Modules: Modules and Files, Namespaces, Importing Modules, Importing Module Attributes, Module Built-in Functions, Packages, Other Features of Modules

CO 2,3
UNIT - III Regular Expressions: Introduction, Special Symbols and Characters, Res and Python Multithreaded Programming: Introduction, Threads and Processes, Python, Threads, and the Global Interpreter Lock, Thread Module, Threading Module, Related Modules

CO 3,4
UNIT - IV GUI Programming: Introduction, Tkinter and Python Programming, Brief Tour of Other GUIs, Related Modules and Other GUIs WEB Programming: Introduction, Wed Surfing with Python, Creating Simple Web Clients, Advanced Web Clients, CGI-Helping Servers Process Client Data,

Building CGI Application Advanced CGI, Web (HTTP) Servers

CO 4,5

UNIT – V Database Programming: Introduction, Python Database Application Programmer's Interface (DB-API), Object Relational Managers (ORMs), Related Modules

CO 5

Course Outcomes:

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

CO 1: Examine Python syntax and semantics and be fluent in the use of Python flow control and functions.

CO 2: Demonstrate proficiency in handling Strings and File Systems.

CO 3: Create, run and manipulate Python Programs using core data structures like Lists, Dictionaries and use Regular Expressions.

CO 4: Interpret the concepts of Object-Oriented Programming as used in Python.

CO 5: Implement exemplary applications related to Network Programming, Web Services and Databases in Python.

Suggested Readings/Books

1. Textbook 1**. Core Python Programming**, Wesley J. Chun, Second Edition, Pearson.

Course Code: BTCS 513-18	Course Title: Programming in Python Lab	0L:0T:2P	1Credits

Detailed List of Tasks:

1. Write a program to demonstrate different number data types in Python.
2. Write a program to perform different Arithmetic Operations on numbers in Python. 3. Write a program to create, concatenate and print a string and accessing sub-string from a given string.
3. Write a python script to print the current date in the following format "Sun May 29 02:26:23 IST 2017"
4. Write a program to create, append, and remove lists in python.
5. Write a program to demonstrate working with tuples in python.
6. Write a program to demonstrate working with dictionaries in python.
7. Write a python program to find largest of three numbers.
8. Write a Python program to convert temperatures to and from Celsius, Fahrenheit. [Formula: c/5 = f-32/9]

9. Write a Python program to construct the following pattern, using a nested for loop
   ```
    *
   * *
   * * *
   * * * *
   * * * * *
   * * * *
   * * *
   * *
   *
   ```
10. Write a Python script that prints prime numbers less than 20.

11. Write a python program to find factorial of a number using Recursion.
12. Write a program that accepts the lengths of three sides of a triangle as inputs. The program output should indicate whether or not the triangle is a right triangle (Recall from the Pythagorean Theorem that in a right triangle, the square of one side equals the sum of the squares of the other two sides).
13. Write a python program to define a module to find Fibonacci Numbers and import the module to another program.
14. Write a python program to define a module and import a specific function in that module to another program.
15. Write a script named copyfile.py. This script should prompt the user for the names of two text files. The contents of the first file should be input and written to the second file.
16. Write a program that inputs a text file. The program should print all of the unique words in the file in alphabetical order.
17. Write a Python class to convert an integer to a roman numeral.
18. Write a Python class to implement pow(x, n)
19. Write a Python class to reverse a string word by word.

Sixth Semester

Course Code: BTCS601-18	Course Title : Compiler Design	3L:0T:0P	3Credits

Detailed Contents:
UNIT 1: Unit I Introduction to Compilers:
Structure of a compiler – Lexical Analysis – Role of Lexical Analyzer – Input Buffering – Specification of Tokens – Recognition of Tokens – Lex – Finite Automata – Regular Expressions to Automata – Minimizing DFA.

[8 hrs., CO 1]

Unit II :Syntax Analysis:
Role of Parser – Grammars – Error Handling – Context-free grammars – Writing a grammar, Top-Down Parsing – General Strategies Recursive Descent Parser – Predictive Parser-LL(1) Parser-Shift Reduce Parser-LR Parser-LR (0) Item Construction of SLR Parsing Table -Introduction to LALR Parser – Error Handling and Recovery in Syntax Analyzer-YACC.

[8 hrs., CO 2]

Unit III : Intermediate Code Generation:
Syntax Directed Definitions, Evaluation Orders for Syntax Directed Definitions, Intermediate Languages: Syntax Tree, Three Address Code, Types and Declarations, Translation of Expressions, Type Checking.

[8 hrs., CO 3]

Unit IV: Run-Time Environment and Code Generation:

Storage Organization, Stack Allocation Space, Access to Non-local Data on the Stack, Heap Management – Issues in Code Generation – Design of a simple Code Generator.

[6 hrs., CO 4]

Unit V: Code Optimization:
Principal Sources of Optimization – Peep-hole optimization – DAG- Optimization of Basic Blocks-Global Data Flow Analysis – Efficient Data Flow Algorithm.

[6 hrs., CO 5]

Course Outcomes:
After undergoing this course, the students will be able to:
CO1: Build concepts on lexical analysis.
CO2: Understand strategies of syntax analysis.
CO3: Learn techniques of Intermediate code generation.
CO4: Undestand code design issues and design code generator.
CO5: Design and develop optimized codes.

Suggested Readings/ Books:
1. A.V. Aho, Monica, R.Sethi, J.D.Ullman, "Compilers, Principles, Techniques and Tools", Second Edition, Pearson Education/Addison Wesley, 2009.
2. Andrew W. Appel, "Modern Compiler Implementation in Java", Second Edition, 2009.
3. J.P. Tremblay and P.G. Sorrenson, "The Theory and Practice of Compiler Writing", McGraw Hill, 1985.

Course Code: BTCS604-18	Course Title: Compiler Design Lab	L:0;T:0; 2P	1Credits

Detailed Contents:

Sr. No.	No. List of Experiments
1	Design a lexical analyser for given language and the lexical analyser 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.
11	Convert the BNF rules into YACC form and write code to generate abstract syntax tree for the mini language specified in Note 1.
12	Write a C program to generate machine code from abstract syntax tree generated by the parser. The instruction set specified in Note 2 may be considered as the target code.

Lab Outcomes: At the end of this course student will:
1. Design Lexical analyzer for given language using C and LEX tools.
2. Design and convert BNF rules into YACC form to generate various parsers.
3. Generate machine code from the intermediate code forms
4. Implement Symbol table

Course Code:BTCS602-18	Course Title : Artificial Intelligence	3L:0T:0P	3Credits

Detailed Contents:

UNIT 1: Introduction (3 Hours)
Concept of AI, history, current status, scope, agents, environments, Problem Formulations, Review of tree and graph structures, State space representation, Search graph and Search tree.

[8hrs] (CO 1)

UNIT 2: Search Algorithms
Random search, Search with closed and open list, Depth first and Breadth first search, Heuristic search, Best first search, A* algorithm, Game Search.

[9hrs] (CO 2)

UNIT 3: Probabilistic Reasoning
Probability, conditional probability, Bayes Rule, Bayesian Networks- representation, construction and inference, temporal model, hidden Markov model.

[6hrs] (CO 3)

UNIT 4 Markov Decision process
MDP formulation, utility theory, utility functions, value iteration, policy iteration and partially observable MDPs. **[6hrs] (CO 4)**

UNIT 5 Reinforcement Learning
Passive reinforcement learning, direct utility estimation, adaptive dynamic programming, temporal difference learning, active reinforcement learning- Q learning**.**

[6hrs] (CO 5)

Course Outcomes:
After undergoing this course, the students will be able to:
CO1: Build intelligent agents for search and games
CO2: Solve AI problems by learning various algorithms and strategies
CO3: Understand probability as a tool to handle uncertainity
CO4: Learning optimization and inference algorithms for model learning
CO5: Design and develop programs for an reinforcement agent to learn and act in a structured environment

Suggested Readings/ Books:
1.Stuart Russell and Peter Norvig, "Artificial Intelligence: A Modern Approach" , 3rd Edition, Prentice Hall
2. Elaine Rich and Kevin Knight, "Artificial Intelligence", Tata McGraw Hill
3. Trivedi, M.C., "A Classical Approach to Artifical Intelligence", Khanna Publishing House, Delhi.
4. Saroj Kaushik, "Artificial Intelligence", Cengage Learning India,
5. David Poole and Alan Mackworth, "Artificial Intelligence: Foundations for Computational Agents", Cambridge University Press 2010

Course Code: BTCS 605-18	Course Title Artificial Intelligence Lab	L:0;T:0;P:2	1 Credits

Detailed List of Tasks:

1. Write a programme to conduct uninformed and informed search.
2. Write a programme to conduct game search.
3. Write a programme to construct a Bayesian network from given data.
4. Write a programme to infer from the Bayesian network.
5. Write a programme to run value and policy iteration in a grid world.
6. Write a programme to do reinforcement learning in a grid world

Lab Outcomes: At the end of the course, the students are able to:

1. Explain artificial intelligence, its characteristics and its application areas.
2. Formulate real-world problems as state space problems, optimization problems or constraint satisfaction problems.
3. Select and apply appropriate algorithms and AI techniques to solve complex problems.
4. Design and develop an expert system by using appropriate tools and techniques.

Elective-II

Course Code: BTCS714-18	Course Title: Parallel Computing	3L: 0T: 0P	Credits: 3

Detailed Contents:

Introduction: Paradigms of parallel computing: Synchronous - vector/array, SIMD, Systolic; Asynchronous - MIMD, reduction paradigm.

Hardware taxonomy: Flynn's classifications, Handler's classifications. Software taxonomy: Kung's taxonomy, SPMD.

Abstract parallel computational models: Combinational circuits, Sorting network, PRAM models, Interconnection RAMs. Parallelism approaches - data parallelism, control parallelism

Performance Metrices: Laws governing performance measurements. Metrices - speedups, efficiency, utilization, communication overheads, single/multiple program performances, bench marks.

Parallel Processors: Taxonomy and topology - shared memory mutliprocessors, distributed memory networks. Processor organization - Static and dynamic interconnections. Embeddings and simulations.

Parallel Programming: Shared memory programming, distributed memory programming, object oriented programming, data parallel programming, functional and dataflow programming.

Scheduling and Parallelization: Scheduling parallel programs. Loop scheduling. Parallelization of sequential programs. Parallel programming support environments.

Books and References:

1. M. J. Quinn. Parallel Computing: Theory and Practice , McGraw Hill, New York, 1994.
2. T. G. Lewis and H. El-Rewini. Introduction to Parallel Computing , Prentice Hall, New Jersey, 1992.
3. T. G. Lewis. Parallel Programming: A Machine-Independent Approach , IEEE Computer Society Press, Los Alamitos, 1994.

Course Outcomes: At the end of the course, students will be able to-
- CO1: To develop an understanding of various basic concepts associated with parallel computing environments.
- CO2: To understand the effects that issues of synchronization, latency and bandwidth have on the efficiency and effectiveness of parallel computing applications.
- CO3: To gain experience in a number of different parallel computing paradigms including memory passing, memory sharing, data-parallel and other approaches.
- CO4: To earn experience in designing and testing parallel computing solutions to programming problems.
- CO5: To develop improved communication and collaborative skills. Symbolic Logic & Logic Processing

Course Code: BTCS715-18	Course Title: Parallel Computing lab	0L: 0T: 2P	Credits: 1

List of Tasks:

1. Write a c program to create child process and print the process id and parent process id using fork function.

2. Write a 'c' header file for sharing and free the memory.

3. Shared memory program model
   a. Write program of array addition.
   b. Write a program of matrix multiplication.
   c. Write a program to calculate factorial of a given number.
   d. Write a program to calculate the sum of 1 to n numbers. 5. Shared memory program model.

4. a. Write a c program for summation of array elements using loop splitting.
   b. Write a c program for summation of array elements using self-scheduling.

5. a. Write a program to create and join thread.
   b. Write a program that calculates the sum of array with n elements.

6. Write a program for matrix multiplication using indirect scheduling.

Lab Outcomes: After successful completion of the lab, students can able to:

CO1:   The course includes individual as well as group projects and provide opportunities for the presentation and defense of designed solutions.

CO2:   The course encourages the development of professional communication skills and provides opportunities for collaborative project development.

Course Code: BTCOE 601	Course Title: Advance Computer Architecture	3L: 0T: 0P	Credits: 3

Detailed Contents:

Unit-1: Theory of Parallelism

Theory of Parallelism: Parallel Computer Models, The State of Computing, Multiprocessors and Multicomputer, Multivector and SIMD Computers, PRAM and VLSI Models, Program and Network Properties, Conditions of Parallelism, Program Partitioning and Scheduling, Program Flow Mechanisms, System Interconnect Architectures, Principles of Scalable Performance, Performance Metrics and Measures, Parallel Processing Applications, Speedup Performance Laws, Scalability Analysis and Approaches.

Unit-2: Hardware Technologies

Hardware Technologies: Processors and Memory Hierarchy, Advanced Processor Technology, Superscalar and Vector Processors, Memory Hierarchy Technology, Virtual Memory Technology.

Unit-3: Bus, Cache, and Shared Memory

Bus, Cache, and Shared Memory, Bus Systems ,Cache Memory Organizations ,Shared Memory Organizations ,Sequential and Weak Consistency Models ,Pipelining and Superscalar Techniques ,Linear Pipeline Processors ,Nonlinear Pipeline Processors , Instruction Pipeline Design ,Arithmetic Pipeline Design (Upto 6.4).

Unit-4: Parallel and Scalable Architectures

Parallel and Scalable Architectures: Multiprocessors and Multicomputers ,Multiprocessor System Interconnects, Cache Coherence and Synchronization Mechanisms, Three Generations of Multicomputers ,Message-Passing Mechanisms ,Multivector and SIMD Computers ,Vector Processing Principles ,Multivector Multiprocessors ,Compound Vector Processing ,SIMD Computer Organizations (Upto 8.4),Scalable, Multithreaded, and Dataflow Architectures, Latency-Hiding Techniques, Principles of Multithreading, Fine-Grain Multicomputers, Scalable and Multithreaded Architectures, Dataflow and Hybrid Architectures.

Unit-5: Software for parallel programming

Software for parallel programming: Parallel Models, Languages, and Compilers ,Parallel Programming Models, Parallel Languages and Compilers ,Dependence Analysis of Data Arrays ,Parallel Program Development and Environments, Synchronization and Multiprocessing Modes. Instruction and System Level Parallelism, Instruction Level Parallelism ,Computer Architecture ,Contents, Basic Design Issues ,Problem Definition ,Model of a Typical Processor ,Compiler-detected Instruction Level Parallelism, Operand Forwarding ,Reorder Buffer, Register Renaming , Tomasulo's Algorithm ,Branch Prediction, Limitations in Exploiting Instruction Level Parallelism ,Thread Level Parallelism.

Books and References:
1. Kai Hwang and Naresh Jotwani, Advanced Computer Architecture (SIE): Parallelism, Scalability, Programmability, McGraw Hill Education 3/e. 2015.
2. John L. Hennessy and David A. Patterson, Computer Architecture: A quantitative approach, 5th edition, Morgan Kaufmann Elseveir, 2013

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

CO1:	Demonstrate concepts of parallelism in hardware/software.
CO2:	Discuss memory organization and mapping techniques.
CO3:	Describe architectural features of advanced processors.
CO4:	Interpret performance of different pipelined processors.
CO5:	Explain data flow in arithmetic algorithms
CO6:	Development of software to solve computationally intensive problems.

Course Code: BTCOE 602	Course Title: Advance Computer Architecture Lab	0L: 0T: 2P	Credits: 1

List of Tasks:

1. HDL introduction
2. Basic digital logic base programming with HDL
3. 8-bit Addition, Multiplication, Division
4. 8-bit Register design
5. Memory unit design and perform memory operations.
6. 8-bit simple ALU design
7. 8-bit simple CPU design
8. Interfacing of CPU and Memory

Lab Outcomes: After successful completion of the lab, students can able to:

CO1: Identify the basic structure and functional units of a computer to analyze the merits and pitfalls in architecture and performance.

CO2: Demonstrate the impact of instruction set architecture, central processing unit on costperformance of computer design and apply it in assembly language programming.

CO3: Select and classify various interrupts used to implement I/O control and data transfers.

CO4: Describe the design process of a computer with arithmetic and logical operation and critical elements in each step.

CO5: Identify the pros and cons of different types of memory hierarchy, cache design and data transfer techniques in computer.

CO6: Analyze the interconnection networks, multiprocessors designs and use research based various design techniques to be employed.

| Course Code: BTCOE 603 | Course Title: Real Time Embedded Systems | 3L: 0T: 0P | Credits: 3 |

Detailed Contents:

Unit-1: Introduction to embedded system design
Complex systems and microprocessors– Embedded system design process –Design example: Model train controller- Design methodologies- Design flows - Requirement Analysis – Specifications-System analysis and architecture design – Quality Assurance techniques - Designing with computing platforms – consumer electronics architecture – platform-level performance analysis.

Unit-2: Arm processor and peripherals

ARM Architecture Versions – ARM Architecture – Instruction Set – Stacks and Subroutines – Features of the LPC 214X Family – Peripherals – The Timer Unit – Pulse Width Modulation Unit – UART – Block Diagram of ARM9 and ARM Cortex M3 MCU.

Unit-3: Embedded programming

Components for embedded programs- Models of programs- Assembly, linking and loading – compilation techniques- Program level performance analysis – Software performance optimization– Program level energy and power analysis and optimization – Analysis and optimization of program size- Program validation and testing.

Unit-4: Real time systems

Structure of a Real Time System – Estimating program run times – Task Assignment and Scheduling – Fault Tolerance Techniques – Reliability, Evaluation – Clock Synchronisation.

Unit-5: Processes and operating systems
Introduction – Multiple tasks and multiple processes – Multirate systems- Preemptive real- time operating systems- Priority based scheduling- Interprocess communication mechanisms – Evaluating operating system performance- power optimization strategies for processes – Example Real time operating systems-POSIX-Windows CE. - Distributed embedded systems – MPSoCs and shared memory multiprocessors. – Design Example - Audio player, Engine control unit – Video accelerator.

Books and References:
1. Marilyn Wolf, ―Computers as Components - Principles of Embedded Computing System Design‖, Third Edition ―Morgan Kaufmann Publisher (An imprint from Elsevier), 2012. (UNIT I, II, III, V).
2. Jane W.S.Liu,‖ Real Time Systems‖, Pearson Education, Third Indian Reprint, 2003.(UNIT IV).
3. Lyla B.Das, ―Embedded Systems : An Integrated Approach‖ Pearson Education, 2013.

2. Jonathan W.Valvano, ―Embedded Microcomputer Systems Real Time Interfacing‖, Third Edition Cengage Learning, 2012.

4. David. E. Simon, ―An Embedded Software Primer‖, 1st Edition, Fifth Impression, Addison- Wesley Professional, 2007.

5. Raymond J.A. Buhr, Donald L.Bailey, ―An Introduction to Real-Time Systems-From Design to Networking with C/C++‖, Prentice Hall, 1999.

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

CO1: The students can develop real time systems that are highly time bounded.

CO2: The students can apply various real time algorithms in building embedded systems.

CO3: The students can implement the RTOS development tools in building real time embedded systems.

CO4: Understand the concept of embedded system, microcontroller, different components of microcontroller and their interactions.

CO5: Get familiarized with programming environment to develop embedded solutions.

CO6: Program ARM microcontroller to perform various tasks.

CO7: Understand the key concepts of embedded systems such as I/O, timers, interrupts and interaction with peripheral devices.

Course Code: BTCOE 604	Course Title: Real Time Embedded Systems Lab	0L: 0T: 2P	Credits: 1

List of Tasks:

1. Digital FIR filter design and simulation.
2. Fixed point Implementation of Digital FIR Filter.
3. Interfacing of ADC and data trasnfer by software polling, study of aliasing.
4. Implementation of Digital FIR Filter on 8051 Microcontroller.
5. LCD - MCU interfacing and displaying a string.
6. Keyboard-MCU interfacing take a input from keypad and display on LCD.
7. Interface a LED matrix and display a number on the matrix.
8. Interfacing 4x4 switch matrix with the microcontroller.
9. Serial Communication between micro controller and PC.

Lab Outcomes: After successful completion of the lab, students can able to:

CO1: Correctly apply the terminology, and list applications, of real time systems
CO2: Translate requirements of real-time systems into forms that can be encoded.
CO3: Work within the constraints imposed by the real-time aspects of systems
CO4: Re-cast practical design problems into real time task models for the purpose of analysis, evaluation or implementation
CO5: Evaluate the implications of design choices on real time system implementation
CO6: Implement simple real time functions using a real time operating system and a programming language suitable for embedded real-time systems
CO7: Analyze and schedule real time task sets on a single processor
CO8: Apply real-time methodology to multiprocessor and distributed systems

Course Code: BTCS608-18	Course Title: Internet of Things	L:3; T:0; P:0	3Credits

DETAIL CONTENTS:

Unit 1. Introduction to IoT
Architectural Overview, Design principles and needed capabilities, IoT Applications, Sensing, Actuation, Basics of Networking, M2M and IoT Technology Fundamentals- Devices and gateways, Data management, Business processes in IoT, Everything as a Service(XaaS), Role of Cloud in IoT, Security aspects in IoT.

(8 Hours) , CO1

Unit 2. Elements of IoT
Hardware Components- Computing (Arduino, Raspberry Pi), Communication, Sensing, Actuation, I/O interfaces. Software Components- Programming API's (using Python /Node.js /Arduino) for Communication, Protocols-MQTT, ZigBee, Bluetooth, CoAP, UDP, TCP

(9 Hours), CO2

Unit 3. IoT Application Development
Solution framework for IoT applications- Implementation of Device integration, Data acquisition and integration, Device data storage- Unstructured data storage on cloud/local server, Authentication, authorization of devices.

(18 Hours) CO3

Unit 4. IoT Case Studies
IoT case studies and mini projects based on Industrial automation, Transportation, Agriculture, Healthcare, Home Automation

(10 Hours),CO4

Course Outcomes: After the completion of this course, the students will be able to:
- CO1: Understand internet of Things and its hardware and software components
- CO2: Interface I/O devices, sensors & communication modules
- CO3: Remotely monitor data and control devices
- CO4: Develop real life IoT based projects

List of Suggested Books :
1. Vijay Madisetti, Arshdeep Bahga, Ïnternet of Things, "A Hands on Approach", University Press
2. Dr. SRN Reddy, Rachit Thukral and Manasi Mishra, "Introduction to Internet of Things: A practical Approach", ETI Labs
3. Pethuru Raj and Anupama C. Raman, "The Internet of Things: Enabling Technologies, Platforms, and Use Cases", CRC Press
4. Jeeva Jose, "Internet of Things", Khanna Publishing House, Delhi
5. Adrian McEwen, "Designing the Internet of Things", Wiley
6. Raj Kamal, "Internet of Things: Architecture and Design", McGraw Hill
7. Cuno Pfister, "Getting Started with the Internet of Things", O Reilly Media

Course Code: BTCS609-18	Course Title: Internet of Things Lab	L:0; T:0; P:2	1Credits

LIST OF PRACTICALS

1. Familiarization with Arduino/Raspberry Pi and perform necessary software installation.

2. To interface LED/Buzzer with Arduino/Raspberry Pi and write a program to turn ON LED for 1 sec after every 2 seconds.

3. To interface Push button/Digital sensor (IR/LDR) with Arduino/Raspberry Pi and write a program to turn ON LED when push button is pressed or at sensor detection.

4. To interface DHT11 sensor with Arduino/Raspberry Pi and write a program to print temperature and humidity readings.

5. To interface motor using relay with Arduino/Raspberry Pi and write a program to turn ON motor when push button is pressed.

6. To interface OLED with Arduino/Raspberry Pi and write a program to print temperature and humidity readings on it.

7. To interface Bluetooth with Arduino/Raspberry Pi and write a program to send sensor data to smartphone using Bluetooth.

8. To interface Bluetooth with Arduino/Raspberry Pi and write a program to turn LED ON/OFF when '1'/'0' is received from smartphone using Bluetooth.

9. Write a program on Arduino/Raspberry Pi to upload temperature and humidity data to thingspeak cloud.

10. Write a program on Arduino/Raspberry Pi to retrieve temperature and humidity data from thingspeak cloud.

11. To install MySQL database on Raspberry Pi and perform basic SQL queries.

12. Write a program on Arduino/Raspberry Pi to publish temperature data to MQTT broker.

Lab Outcomes: On successful completion of the course, the student will:
   CO1:   Understand the concepts of Internet of Things
   CO2:   Analyze basic protocols in wireless sensor network
   CO3:   Design IoT applications in different domain and be able to analyze their performance
   CO4:   Implement basic IoT applications on embedded platform

Elective-III

Course Code: BTCS 616-18	Course Title : Data Science	3L:0T:0P	3Credits

Detailed Contents:

UNIT 1: Introduction
Introduction to Data Science, Evolution of Data Science, Data Science Roles, Stages in a Data Science Project, Applications of Data Science in various fields, Data Security Issues.

[8hrs] (CO 1)

UNIT 2: Data Collection and Data Pre-Processing
Data Collection Strategies, Data Pre-Processing Overview, Data Cleaning, Data Integration and Transformation, Data Reduction, Data Discretization.

[9hrs] (CO 2)

UNIT 3: Exploratory Data Analytics
Descriptive Statistics, Mean, Standard Deviation, Skewness and Kurtosis, Box Plots, Pivot Table, Heat Map, Correlation Statistics, ANOVA.

[6hrs] (CO 2)

UNIT 4: Model Development
Simple and Multiple Regression, Model Evaluation using Visualization, Residual Plot, Distribution Plot, Polynomial Regression and Pipelines, Measures for In-sample Evaluation, Prediction and Decision Making.

[6hrs] (CO 3)

UNIT 5 Model Evaluation
Generalization Error, Out-of-Sample Evaluation Metrics, Cross Validation, Overfitting, Under Fitting and Model Selection, Prediction by using Ridge Regression, Testing Multiple Parameters by using Grid Search.

[6hrs] (CO 4)

Course Outcomes:
After undergoing this course, the students will be able to:
CO1: understand of the basics of the ethical use of data science
CO2: Build skills in transformation and merging of data for use in analytic tools
CO3: Perform linear and multiple linear regression analyses.
CO4: Evaluate outcomes and make decisions based on data

Suggested Readings/ Books:
1. Jojo Moolayil, "Smarter Decisions: The Intersection of IoT and Data Science", PACKT, 2016.
2. Cathy O'Neil and Rachel Schutt, "Doing Data Science", O'Reilly, 2015.
3. David Dietrich, Barry Heller, Beibei Yang, "Data Science and Big Data Analytics", EMC 2013.
4. Raj, Pethuru, "Handbook of Research on Cloud Infrastructures for Big Data Analytics", IGI Global

Course Code: BTCS 617-18	Course Title : Data Science Lab	0L:0T:2P	1Credits

Detailed List of Tasks:
1. Creating and displaying Data. and Matrix manipulations
2. Creating and manipulating a List and an Array
3. Creating a Data Frame and Matrix-like Operations on a Data Frame
4. Merging two Data Frames and Applying functions to Data Frames
5. Visualization Effects
6. Plotting with Layers
7. Overriding Aesthetics
8. Histograms and Density Charts
9. Simple Linear Regression – Fitting, Evaluation and Visualization
10. Multiple Linear Regression, Lasso and Ridge Regression

Suggested Tools – Python, R

Lab Outcomes: After successful completion of the lab, students can able to:

CO1: Plan the projects in the domain of data science.

CO2: Use data analytics tools towards problem solving and solution analysis.

CO3: Apply Mathematical sciences and recent technologies in Computer Science to solve real life problems

CO4: Apply data science concepts and methods to solve problems in real-world context.

| Course Code: BTCOE 605 | Course Title : Wireless Sensor Network | 3L:0T:0P | 3Credits |

Detailed Contents:
Unit-1: Sensor networks overview
Introduction, Applications of WSN, Range of Applications, Design issues

Unit-2: Basic Wireless Sensor Technology
Sensor node architecture, Hardware and Software, Sensor Taxonomy, WSN Operating Environment, Trend

Unit-3: Wireless Transmission Technology and Systems
Introduction, Radio Technology Primer, Propagation & Propagation Impairments, Available Wireless Technologies

Unit-4: Fundamentals of MAC Protocols
Performance Requirements, Common Protocols, MAC Protocols for WSNs, Schedule-Based Protocols, Random Access-Based Protocols, Sensor-MAC Case Study, Protocol Overview, Periodic Listen and Sleep Operations, Schedule Selection and Coordination, Schedule Synchronization, Adaptive Listening, Access Control and Data Exchange.

Unit-5: Routing Protocols for Wireless Sensor Networks
Routing Challenges and Design Issues in Wireless, Sensor Networks, Network Scale and Time-Varying Characteristics, Resource Constraints, Sensor Applications Data Models, Routing Strategies in Wireless Sensor Networks, WSN Routing Techniques, Flooding and Its Variants,

Unit-6: Transport Control Protocols for Wireless Sensor Networks
Transport Protocol Design Issues, Examples of Existing Transport Control Protocols, CODA (Congestion Detection and Avoidance), ESRT (Event-to-Sink Reliable Transport), ATP (Ad Hoc Transport Protocol), Problems with Transport Control Protocols, Performance of Transport Control Protocols, Congestion, Packet Loss Recovery.

Unit-7: Middleware for Wireless Sensor Networks
Introduction, Network Management Requirements, Traditional Network Management Models, Simple Network Management Protocol, Telecom Operation Map, Network Management Design Issues, Example of Management Architecture: MANNA, Other Issues Related to Network Management, Naming, Localization.

Unit-8: Performance and Traffic Management
WSN Design Issues, MAC Protocols, Routing Protocols, Transport Protocols, Performance Modeling of WSNs, Performance Metrics, Basic Models, Network Models.

Books and References:

1. Wireless Sensor Networks: Technology, Protocols, and Applications by Kazem Sohraby/wiley.
2. Wireless Sensor Networks by Zhao Feng/ Elsevier India.
3. Security in Wireless Sensor Networks by Piotr Szczechowiak/ Lap Lambert Academic Publishing.
4. Wireless Sensor Networks by Raghavendra Sivalingam Znati/ Springer India 5. Building Wireless Sensor Networks by Robert Faludi/ O'reilly.

Course Outcomes: After completing this course the students should:

CO1:	Understand and explain common wireless sensor node architectures.
CO2:	Be able to carry out simple analysis and planning of WSNs.
CO3:	Demonstrate knowledge of MAC protocols developed for WSN.
CO4:	Demonstrate knowledge of routing protocols developed for WSN.
CO5:	Understand and explain mobile data-centric networking principles.
CO6:	Be familiar with WSN standards.

Course Code: BTCOE 606	Course Title : Wireless Sensor Network Lab	0L:0T:2P	1Credits

List of Tasks:

1. Introduction of Wireless sensor network applications and its simulation.
2. Network Simulator installation of wireless sensor network.
3. Write TCL script for transmission between mobile nodes.
4. Write TCL script for sensor nodes with different parameters.
5. Generate tcl script for udp and CBR traffic in WSN nodes.
6. Generate tcl script for TCP and CBR traffic in WSN nodes.
7. Implementation of routing protocol in NS2 for AODV protocol.
8. Implementation of routing protocol in NS2 for DSR protocol.
9. Implementation of routing protocol in NS2 for TORA protocol.
10. Study other wireless sensor network simulators (Mannasim. Contiki.)

Lab Outcomes: After successful completion of the lab, students can able to:
- **CO1:** Analyze and understand electronic Principles related to sensor
- **CO2:** Choose the application specific sensor and adapt the network or topology required
- **CO3:** Test the sensors
- **CO4:** Apply knowledge of wireless sensor network to various applications areas
- **CO5:** Build and implement Wireless Sensor Network

Course Code: BTCOE 607	Course Title : Hardware Security	3L:0T:0P	3Credits

Detailed Contents:

Unit-1: Preliminaries
Algebra of Finite Fields, Basics of the Mathematical Theory of Public Key Cryptography, Basics of Digital Design on Field-programmable Gate Array (FPGA), Classification using Support Vector Machines (SVMs)

Unit-2: Useful Hardware Security Primitives
Cryptographic Hardware and their Implementation, Optimization of Cryptographic Hardware on FPGA, Physically Unclonable Functions (PUFs), PUF Implementations, PUF Quality Evaluation, Design Techniques to Increase PUF Response Quality

Unit-3: Side-channel Attacks on Cryptographic Hardware
Basic Idea, Current-measurement based Side-channel Attacks (Case Study: Kocher's Attack on DES), Design Techniques to Prevent Side-channel Attacks, Improved Side-channel Attack Algorithms (Template Attack, etc.), Cache Attacks

Unit-4: Testability and Verification of Cryptographic Hardware
Fault-tolerance of Cryptographic Hardware, Fault Attacks, Verification of Finite-field Arithmetic Circuits

Unit-5: Modern IC Design and Manufacturing Practices and Their Implications
Hardware Intellectual Property (IP) Piracy and IC Piracy, Design Techniques to Prevent IP and IC Piracy, Using PUFs to prevent Hardware Piracy, Model Building Attacks on PUFs (Case Study: SVM Modeling of Arbiter PUFs, Genetic Programming based Modeling of Ring Oscillator PUF)

Unit-6: Hardware Trojans
Hardware Trojan Nomenclature and Operating Modes, Countermeasures Such as Design and Manufacturing Techniques to Prevent/Detect Hardware Trojans, Logic Testing and Side-channel Analysis based Techniques for Trojan Detection, Techniques to Increase Testing Sensitivity Infrastructure Security: Impact of Hardware Security Compromise on Public Infrastructure, Defense Techniques (Case Study: Smart-Grid Security)

Books and References:

1. Debdeep Mukhopadhyay and Rajat Subhra Chakraborty, "Hardware Security: Design, Threats, and Safeguards", CRC Press.
2. Ahmad-Reza Sadeghi and David Naccache (eds.): Towards Hardware-intrinsic Security: Theory and Practice, Springer.
3. Ted Huffmire et al: Handbook of FPGA Design Security, Springer.
4. Stefan Mangard, Elisabeth Oswald, Thomas Popp: Power analysis attacks - revealing the secrets of smart cards. Springer 2007.
5. Doug Stinson, Cryptography Theory and Practice, CRC Press.

Course Outcomes: Upon completion, students will be able to:

CO1: Describe state-of-the-art hardware security techniques. Justify their targeted applications and limitations. Describe how security is assured in an exemplary application.

CO2: Describe the threats to a system from the hardware perspective as well as available countermeasures. Apply the knowledge to select a suitable set of countermeasures for a specific threat scenario.

CO3: Analyze and critically assess trade-offs between system performance, cost, and security.

CO4: Explain the need for hardware security primitives. Justify pros and cons of different primitives and select a suitable one for a specific application.

CO5: Explain how the security is assured in the system.

Course Code: BTCOE 608	Course Title : Hardware Security Lab	0L:0T:2P	1Credits

List of Tasks:

1. Implementation of a basic morse-code transmitter and receiver on a 32-bit microcontroller (warm-up exercise for familiarization with ARM Cortex-M4 development, debugging and decompilation).
2. Preparation of a circuit diagram from high-resolution photographs and X-ray images of a target printed circuit board.
3. Extraction of the firmware and recording of a protocol exchange from a microcontroller PCB.
4. Partial decompilation (using Ghidra) of the firmware, along the execution path taken by the protocol exchange.
5. Reimplementation of the observed and decompiled elliptic-curve scalar multiplication (ECSM) operation in a high-level language (e.g., Python, Sage, Julia, Perl).

Lab Outcomes: This module's qualification aims are:
   CO1:   Thorough understanding of security relevant logic on modern integrated circuits.
   CO2:   Familiarity with different classes of hardware analysis techniques.
   CO3:   Design of purpose-built hardware analysis tools.
   CO4:   Hardware attack mitigation techniques and architectural improvements.

Course Code: BTCOE 609	Course Title : Human Computer Interactions	3L:0T:0P	3Credits

Detailed Contents:

Unit-1: Foundations of HCI
The Human: I/O channels – Memory – Reasoning and problem solving; The Computer: Devices – Memory – processing and networks; Interaction: Models – frameworks – Ergonomics – styles – elements – interactivity- Paradigms. - Case Studies

Unit-2: Design & software process
Interactive Design: Basics – process – scenarios – navigation – screen design – Iteration and prototyping. HCI in software process: Software life cycle – usability engineering – Prototyping in practice – design rationale. Design rules: principles, standards, guidelines, rules. Evaluation Techniques – Universal Design

Unit-3: Models and theories
HCI Models: Cognitive models: Socio-Organizational issues and stakeholder requirements – Communication and collaboration models-Hypertext, Multimedia and WWW.

Unit-4: Mobile HCI
Mobile Ecosystem: Platforms, Application frameworks- Types of Mobile Applications: Widgets, Applications, Games- Mobile Information Architecture, Mobile 2.0, Mobile Design: Elements of Mobile Design, Tools. - Case Studies

Unit-5: Web interface design
Designing Web Interfaces – Drag & Drop, Direct Selection, Contextual Tools, Overlays, Inlays and Virtual Pages, Process Flow - Case Studies

Books and References:
1. Alan Dix, Janet Finlay, Gregory Abowd, Russell Beale, ―Human Computer Interaction‖, 3rd Edition, Pearson Education, 2004 (UNIT I, II & III)
2. Brian Fling, ―Mobile Design and Development‖, First Edition, O‘Reilly Media Inc., 2009 (UNIT – IV)
3. Bill Scott and Theresa Neil, ―Designing Web Interfaces‖, First Edition, O‘Reilly, 2009. (UNIT-V)

Course Outcomes: Upon completion, students will be able to:

CO1: Analyze Human-Computer Interaction principle and designs in Information Systems.

CO2: Compare various HCI designs to gain knowledge on user-centric interfaces.

CO3: Evaluate the Internet sites considering; usability and user appreciation designs.

CO4:   Appraise social websites such as; facebook, linkedin, twitter or others from user-centric and HCI viewpoint.

CO5:   Construct conceptual basis to design HCI that includes: problems, goals, user interaction style, as well as user-centric interface design.

CO6:   Apply Information Systems tools to prototype the end-user design.

CO7:   Develop end-user interfaces incorporating problem solving solutions in HCI.

Course Code: BTCOE 610	Course Title : Human Computer Interactions Lab	0L:0T:2P	1Credits

List of Tasks:

1. Do a survey of existing user interfaces available for the selected topic/area and find out pros and cons of those interfaces.

2. Carry out the case study of user interface building and evaluation tools. Further, students will use one of these tools for implementing the prototype of the interface.

3. User Study- Create roles of various users of the system. Decide the role specific tasks of the selected problem. This can be carried out considering the aspects such as Contextual inquiry and representation- understand current environment, practices, use, and imagine how the interface will be used in the specific context.

4. Perform the task analysis by identifying various tasks objects and actions a user can perform on the interface and recommend appropriate interaction style.

5. GOMS and Keystroke Level Model of Proposed Design- Design and develop GOMS model for the selected problem. Develop 2-3 alternate Keystroke Level Model designs and analyze their time required by different types of users to execute a task using an interface and interaction method.

6. Prototype Designing- Design a prototype of the proposed user interface using any open source prototyping tool.

7. Evaluation of Design- Conduct an evaluation of the prototype developed in assignment 6 and describes a set of improvements to your design based on its user evaluation. Conduct the usability testing of developed interface by peers to find out the following usability measures- Time to learn, Speed of performance, Rate of errors by users, Retention over time, Subjective satisfaction.

Lab Outcomes: This module's qualification aims are:

CO1:   Analyze and identify usability issues in User interfaces.
CO2:   Design user interfaces according to the standards.
CO3:   Evaluate user interfaces using Heuristic Evaluation and Thinking aloud Test.
CO4:   Demonstrate skills to collaborate in a team for justifying identified problems and to write interface related reports as per the standards.

Seventh Semester

Course Code: BTCS 701-18	Course Title : Network Security and Cryptography	3L:0T:0P	3Credits

Detailed Contents:

UNIT 1: Introduction (3 Hours)

Introduction to Cryptography, Security Threats, Vulnerability, Active and Passive attacks, Security services and mechanism, Conventional Encryption Model, CIA model.

[5hrs] (CO 1)

UNIT 2: Math Background

Modular Arithmetic, Euclidean and Extended Euclidean algorithm, Prime numbers, Fermat and Euler's Theorem

[5hrs] (CO 1)

UNIT 3: Cryptography

Dimensions of Cryptography, Classical Cryptographic Techniques Block Ciphers (DES, AES) : Feistal Cipher Structure, Simplifies DES, DES, Double and Triple DES, Block Cipher design Principles, AES, Modes of Operations Public-Key Cryptography : Principles Of Public-Key Cryptography, RSA Algorithm, Key Management, Diffie-Hellman Key Exchange, Elgamal Algorithm, Elliptic Curve Cryptography

[12hrs] (CO 2)

UNIT 4 Hash and MAC Algorithms

Authentication Requirement, Functions, Message Authentication Code, Hash Functions, Security of Hash Functions and Macs, MD5 Message Digest Algorithm, Secure Hash Algorithm, Digital Signatures, Key Management: Key Distribution Techniques, Kerberos

[6hrs] (CO 3)

UNIT 5 Security in Networks

Threats in networks, Network Security Controls – Architecture, Encryption, Content Integrity, Strong Authentication, Access Controls, Wireless Security, Honeypots, Traffic flow security, Firewalls – Design and Types of Firewalls, Personal Firewalls, IDS, Email Security – PGP, S/MIME

[7hrs] (CO 4)

Course Outcomes:

After undergoing this course, the students will be able to:

CO1: Understand the fundamental principles of access control models and techniques, authentication and secure system design

CO2: Have a strong understanding of different cryptographic protocols and techniques and be able to use them.

CO3: Apply methods for authentication, access control, intrusion detection and prevention.

CO4: Identify and mitigate software security vulnerabilities in existing systems.

Suggested Readings/ Books:

1. Cryptography And Network Security Principles And Practice Fourth Edition, William Stallings, Pearson Education
2. Modern Cryptography: Theory and Practice, by Wenbo Mao, Prentice Hall PTR
3. Network Security Essentials: Applications and Standards, by William Stallings. Prentice Hall
4. Cryptography: Theory and Practice by Douglas R. Stinson, CRC press.

Course Code: BTCS 702-18	Course Title: Data Mining and Data Warehousing	3L: 0T: 0P	Credits: 3

Detailed Contents:

UNIT 1:

Data Warehousing Introduction: design guidelines for data warehouse implementation, Multidimensional Models; OLAP- introduction, Characteristics, Architecture, Multidimensional view Efficient processing of OLAP Queries, OLAP server Architecture ROLAP versus MOLAP Versus HOLAP and data cube, Data cube operations, data cube computation.

Data mining: What is data mining, Challenges, Data Mining Tasks, Data: Types of Data, Data Quality, Data Pre-processing, Measures of Similarity and Dissimilarity

[10hrs]

UNIT 2:

Data mining: Introduction, association rules mining, Naive algorithm, Apriori algorithm, direct hashing and pruning (DHP), Dynamic Item set counting (DIC), Mining frequent pattern without candidate generation (FP, growth), performance evaluation of algorithms

Classification: Introduction, decision tree, tree induction algorithms – split algorithm based on information theory, split algorithm based on Gini index; naïve Bayes method; estimating predictive accuracy of classification method

[10 hrs]

UNIT 3:

Cluster analysis: Introduction, partition methods, hierarchical methods, density based methods, dealing with large databases, cluster software

Search engines: Characteristics of Search engines, Search Engine Functionality, Search Engine Architecture, Ranking of web pages, The search engine history, Enterprise Search, Enterprise Search Engine Software.

[10 hrs]

UNIT 4:

Web data mining: Web Terminology and Characteristics, Locality and Hierarchy in the web, Web Content Mining, Web Usage Mining, Web Structure Mining, Web mining Software.

[8 hrs]

Suggested Readings / Books:

1. Carlo Vercellis, Business Intelligence: Data mining and Optimization for Decision Making, WILEY.
2. Han J., Kamber M. and Pei J. , b Data mining concepts and techniques, Morgan Kaufmann Publishers (2011) 3rd ed.
3. Pudi V., Krishana P.R., Data Mining, Oxford University press, (2009) 1st ed.
4. Adriaans P., Zantinge D., Data mining, Pearsoneducation press (1996), 1st ed.
5. Pooniah P. , Data Warehousing Fundamentals, Willey interscience Publication, (2001), 1st ed.

ELECTIVE IV

Course Code: BTCS 618-18	Course Title : Machine Learning	3L:0T:0P	3Credits

Detailed Contents:

UNIT 1: **Introduction:** Well-Posed learning problems, Basic concepts, Designing a learning system, Issues in machine learning. Types of machine learning: Learning associations, Supervised learning, Unsupervised learning and Reinforcement learning.

[4hrs] (CO 1)

UNIT 2: Data Pre-processing: Need of Data Pre-processing, Data Pre-processing Methods: Data Cleaning, Data Integration, Data Transformation, Data Reduction; Feature Scaling (Normalization and Standardization), Splitting dataset into Training and Testing set.

[4hrs] (CO 2)

UNIT 3: Regression: Need and Applications of Regression, Simple Linear Regression, Multiple Linear Regression and Polynomial Regression, Evaluating Regression Models Performance (RMSE, Mean Absolute Error, Correlation, RSquare, Accuracy with acceptable error, scatter plot, *etc*.)

[6hrs] (CO 3)

UNIT 4 Classification: Need and Applications of Classification, Logistic Regression, Decision tree, Tree induction algorithm – split algorithm based on information theory, split algorithm based on Gini index; Random forest classification, Naïve Bayes algorithm; K-Nearest Neighbours (K-NN), Support Vector Machine (SVM), Evaluating Classification Models Performance (Sensitivity, Specificity, Precision, Recall, *etc*.). **Clustering**: Need and Applications of Clustering, Partitioned methods, Hierarchical methods, Density-based methods.

[12hrs] (CO 4)

UNIT 5 Association Rules Learning: Need and Application of Association Rules Learning, Basic concepts of Association Rule Mining, Naïve algorithm, Apriori algorithm. **Artificial Neural Network:** Need and Application of Artificial Neural Network, Neural network representation and working, Activation Functions. **Genetic Algorithms:** Basic concepts, Gene Representation and Fitness Function, Selection, Recombination, Mutation and Elitism.

[14hrs] (CO 5)

Course Outcomes:
After undergoing this course, the students will be able to:
CO1: Analyse methods and theories in the field of machine learning
CO2: Analyse and extract features of complex datasets
CO3: Deploy techniques to comment for the Regression
CO4: Comprehend and apply different classification and clustering techniques
CO5: Understand the concept of Neural Networks and Genetic Algorithm

Suggested Readings/ Books:
Text Books:

1.   Mitchell M., T., Machine Learning, McGraw Hill (1997) 1stEdition.

2.   Alpaydin E., Introduction to Machine Learning, MIT Press (2014) 3rdEdition.
3.   Vijayvargia Abhishek, Machine Learning with Python, BPB Publication (2018)

Reference Books:

1.   Bishop M., C., Pattern Recognition and Machine Learning, Springer-Verlag (2011) 2ndEdition.
2.   Michie D., Spiegelhalter J. D., Taylor C. C., Campbell, J., Machine Learning, Neural and Statistical Classification. Overseas Press (1994).

……………………………………………………………………………………………………

Course Code: BTCS619-18	Course Title: machine Learning Lab	L:0;T:0;2 P:1Credits

Detailed List of Tasks:

1.   Implement data pre-processing
2.   Deploy Simple Linear Regression
3.   Simulate Multiple Linear Regression
4.   Implement Decision Tree
5.   Deploy Random forest classification
6.   Simulate Naïve Bayes algorithm
7.   Implement K-Nearest Neighbors (K-NN), k-Means
8.   Deploy Support Vector Machine, Apriori algorithm
9.   Simulate Artificial Neural Network
10.   Implement the Genetic Algorithm code

Suggested Tools Python/R/MATLAB

Course Code: BTCS620-18	Course Title: Mobile Application Development	L:3; T:0; P:0	3Credits

Details of course:
Unit-1
Introduction to Android: The Android Developing environment, Android SDK, Introduction to Open Handset Alliance, Development Framework, Application Fundamentals; Device Compatibility, System permissions, Understanding Anatomy of Android Application, Android Development Tools
6 hrs. ,CO 1

Unit-II
Getting started with Mobility: Mobility Landscape, Mobile Platforms, Mobile apps development, Android terminologies, Application Context, Activities, Services, Intents, Receiving and Broadcasting Intents, Setting up the mobile apps development environment with emulator
6 hrs., CO1

Unit-III

Building block of Mobile apps: App user Interface Designing, Layout, User Interface elements, VUIs and Mobile Apps, Text to Speech Techniques, Designing the Right UI, Activity states and lifecycle, Interaction among activities

6 hrs., CO 2

Unit-IV

Sprucing up Mobile apps: App functionality beyond user interface- Threads, sync task, Services-states and life cycle, Notifications, Broadcast receivers, Telephony and SMS APIs Native data handling: on device file I/O, shared preferences, mobile databases such as SQLite, Working with a content provider

8 hrs., CO 3,4

Unit-V

Factors in Developing Mobile Applications: Mobile Software Engineering, Frameworks and Tools, Generic UI Development, Android User

Graphics and Multimedia: Performance and Multithreading, Graphics and UI Performance, Android Graphics, Mobile Agents and Peer-to-Peer Architecture, Android Multimedia

8 hrs., CO 4,5

Unit-VI

Platforms and Additional Issues: Development Process, Architecture, Design, Technology Selection, Testing, Security and Hacking, Active Transactions, More on Security

8hrs., CO 5

Course Outcomes:

CO 1: Describe those aspects of mobile programming that make it unique from programming for other platforms,

CO 2: Critique mobile applications on their design pros and cons,

CO 3: Utilize rapid prototyping techniques to design and develop sophisticated mobile interfaces,

CO 4: Program mobile applications for the Android operating system that use basic and advanced phone features, and

CO 5: Deploy applications to the Android marketplace for distribution

References:

1. Rick Rogers, John Lombardo, Meike Blake, "Android application development", Ist Edition, O'Reilly, 2010

2. Lauren Darcey and Shane Conder,"Android Wireless Application Development", 2nd ed. Pearson Education, 2011

3. Wei-Meng Lee , Beginning Android 4 development ,2012 by John Wiley & Sons

4. Jeff Mewherter, Scott Gowell,Wrox Publisher,"Professional Mobile Application Development", Ist Edition, 2012

5. Reto Meier, "Professional Android 4 Application Development", Wrox, 2012

……………..……………………………………….……………………………………….....
.

Course Code: BTCS621-18	Course Title: Mobile Application Development Lab	L:0; T:0; P:2	1Credits

LIST OF PRACTICALS

1**.** Introduction to Android platform. Introduction to the tools used in the lab. Create a simple application
2. Understand the app idea and design user interface/wireframes of mobile app
3. Set up mobile app development environment
4. Write a program using activity class to show different events.
5. Write a program to convert text to speech.
6. Develop and debug mobile app components – User interface, services, notifications, broadcast receivers, data components
7. Using emulator to deploy and run mobile apps
8. Testing mobile app- unit testing, black box testing and test automation

……….…………………………………………………………………….....................

Course Code: BTCS 704-18	Course Title : Deep Learning	3L:0T:0P	3Credits

Detailed Contents:

UNIT 1: **Machine Learning Basics:** Learning, Under-fitting, Overfitting, Estimators, Bias, Variance, Maximum Likelihood Estimation, Bayesian Statistics, Supervised Learning, Unsupervised Learning and Stochastic Gradient Decent.

[4hrs] (CO 1)
UNIT 2: Deep Feedforward Network: Feed-forward Networks, Gradient-based Learning, Hidden Units, Architecture Design, Computational Graphs, Back-Propagation, Regularization, Parameter Penalties, Data Augmentation, Multi-task Learning, Bagging, Dropout and Adversarial Training and Optimization.

[4hrs] (CO 2)
UNIT 3: Convolution Networks: Convolution Operation, Pooling, Basic Convolution Function, Convolution Algorithm, Unsupervised Features and Neuroscientific for convolution Network.
[6hrs] (CO 3)
UNIT 4: Sequence Modelling: Recurrent Neural Networks (RNNs), Bidirectional RNNs, Encoder-Decoder Sequence-to-Sequence Architectures, Deep Recurrent Network, Recursive Neural Networks and Echo State networks.
[12hrs] (CO 4)
UNIT 5: Deep Generative Models: Boltzmann Machines, Restricted Boltzmann Machines, Deep Belief Networks, Deep Boltzmann Machines, Sigmoid Belief Networks, Directed Generative Net, Drawing Samples from Auto –encoders.

[14hrs] (CO 5)

Course Outcomes:
After undergoing this course, the students will be able to:
CO1: Comprehend the advancements in learning techniques
CO2: Compare and explain various deep learning architectures and algorithms.
CO3: Demonstrate the applications of Convolution Networks
CO4: Apply Recurrent Network for Sequence Modelling
CO5: Deploy the Deep Generative Models

Suggested Readings/ Books:

Text Books:

1.   *Goodfellow L., Bengio Y. and Courville A., Deep Learning, MIT Press (2016).*
2.   *Patterson J. and Gibson A., Deep Learning: A Practitioner's Approach, O'Reilly (2017), 1st ed.*

Reference Books:

1.   *Haykin S., Neural Network and Machine Learning, Prentice Hall Pearson (2009), 3rd ed.*
2.   *Geron A., Hands-on Machine Learning with Sci-kit and TensorFlow, O'Reilly Media (2017)*

……………………………………..…………………………………………………………………………

Course Code: BTCS 705-18	Course Title: Deep Learning Lab	L:0;T:0; 2P:	Credits;1

Detailed List of Tasks:

- Creating a basic network and analyze its performance
- Deploy the Confusion matrix and simulate for Overfitting
- Visualizing a neural network
- Demo: Object Detection with pre-trained RetinaNet with Keras
- Neural Recommender Systems with Explicit Feedback
- Backpropagation in Neural Networks using Numpy
- Neural Recommender Systems with Implicit Feedback and the Triplet Loss
- Fully Convolutional Neural Networks
- ConvNets for Classification and Localization
- Text Classification and Word Vectors
- Character Level Language Model (GPU required)

Suggested Tools Python/R/MATLAB
…….

Course Code: BTCS721-18	Course Title: Block Chain Technology	3L:0 T: 0P	Credits: 3

Detailed Contents:

INTRODUCTION TO BLOCKCHAIN
Blockchain- Public Ledgers, Blockchain as Public Ledgers -Bitcoin, Blockchain 2.0, Smart Contracts, Block in a Blockchain, Transactions-Distributed Consensus, The Chain and the Longest Chain - Cryptocurrency to Blockchain 2.0 - Permissioned Model of Blockchain, Cryptographic -Hash Function, Properties of a hash Function- Hash pointer and Merkle tree

BITCOIN AND CRYPTOCURRENCY
A basic crypto currency, Creation of coins, Payments and double spending, FORTH – the precursor for Bitcoin scripting, Bitcoin Scripts, Bitcoin P2P Network, Transaction in Bitcoin Network, Block Mining, Block propagation and block relay, Consensus introduction, Distributed consensus in open Environments-Consensus in a Bitcoin network

BITCOIN CONSENSUS
Bitcoin Consensus, Proof of Work (PoW)- Hashcash PoW , Bitcoin PoW, Attacks on PoW ,monopoly problem- Proof of Stake- Proof of Burn - Proof of Elapsed Time - Bitcoin Miner, Mining Difficulty, Mining Pool- Permissioned model and use cases, Design issues for Permissioned Blockchains, Execute contracts- Consensus models for permissioned blockchain-Distributed consensus in closed environment Paxos

DISTRIBUTED CONSENSUS
RAFT Consensus-Byzantine general problem, Byzantine fault tolerant system-Agreement Protocol, Lamport-Shostak-Pease BFT Algorithm-BFT over Asynchronous systems, Practical Byzantine Fault Tolerance

HYPER LEDGER FABRIC & ETHERUM
Architecture of Hyperledger fabric v1.1-Introduction to hyperledger fabric v1.1, chain code- Ethereum: Ethereum network, EVM, Transaction fee, Mist Browser, Ether, Gas, Solidity, Smart contracts, Truffle Design and issue Crypto currency, Mining, DApps, DAO

BLOCKCHAIN APPLICATIONS
Internet of Things-Medical Record Management System-Block chain in Government and Block chain Security-Block chain Use Cases –Finance

COURSE OUTCOMES
CO1: Understand emerging abstract models for Block chain Technology.
CO2: Identify major research challenges and technical gaps existing between theory and practice in crypto currency domain.
CO3: It provides conceptual understanding of the function of Blockchain as a method of securing distributed ledgers, how consensus on their contents is achieved, and the new applications that they enable.
CO4: Apply hyperledger Fabric and Etherum platform to implement the Block chain Application.

REFERENCES
1. Mastering Blockchain: Deeper insights into decentralization, cryptography, Bitcoin, and popular Blockchain frameworks by Bashir, Imran,2017.
2. Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller, and Steven Goldfeder. Bitcoin and cryptocurrency technologies: a comprehensive introduction. Princeton University Press, 2016.
 3. Joseph Bonneau et al, SoK: Research perspectives and challenges for Bitcoin and cryptocurrency, IEEE Symposium on security and Privacy, 2015.

…………………………………………………………………………………………..………………………….

Course Code: 713-18	Course Title: Block chain Technology lab	L: T: 2P	Credits:1

Detailed List of Tasks:

1. To Develop Naive Block chain construction.

2. Design Memory Hard algorithm and its Implementation

3. Design Toy application using Blockchain

5. Program to Solve a Mining puzzles using Block chain

6. The ability to formulate mathematical models and problem-solving skills through programming techniques for addressing real-time problems using appropriate data structures and algorithms.

7. The ability to provide design, build, and deploy a distributed application and provide solutions using block chain applications to enhance business measures by sharing information safely and effectively.

8. The ability to create crypto currencies and give a strong technical understanding of Block chain technologies with an in-depth understanding of applications, open research challenges, and future directions.

ELECTIVE V

Course Code: BTCS 612-18	Course Title: Cloud Computing	3L:0T:0P	3Credits

Detailed Contents:

UNIT1: **Introduction:** Definition of cloud, characteristics of cloud, historical developments & challenges ahead, the vision of cloud computing, Driving factors towards cloud, Comparing grid with utility computing, cloud computing and other computing systems, types of workload patterns for the cloud, IT as a service, Applications of cloud computing.

[8hrs] (CO1)

UNIT2: Cloud computing concepts: Introduction to virtualization techniques, Characteristics of virtualization, Pros and Cons of virtualization Technology, Hypervisors, Types of hypervisors, Multitenancy, Application programming interfaces (API), Elasticity and scalability.

[9hrs] (CO2)

UNIT 3: Cloud service models: Cloud service models, Infrastructure as a service (IaaS) architecture-details and example, Platform as a service (PaaS) architecture- details and example, Software as a service (SaaS) architecture-- details and example, Comparison of cloud service delivery models.

[6hrs] (CO3)

UNIT 4: Cloud deployment models: Introduction to cloud deployment models, Public clouds, Private clouds, Hybrid clouds, Community clouds, Migration paths for cloud, Selection criteria for cloud deployment.

[6hrs] (CO4)

UNIT 5: Security in cloud computing: Understanding security risks, Principal security dangers to cloud computing, Internal security breaches, User account and service hijacking, measures to reduce cloud security breaches
Case Studies: Comparison of existing Cloud platforms /Web Services.

[6hrs] (CO5)

Course Outcomes:
After undergoing this course, the students will be able to:
CO1: Understand the core concepts of the cloud computing paradigm
CO2: Understanding importance of virtualization along with their technologies
CO3: Analyze various cloud computing service and deployment models and apply them to solve problems on the cloud.
CO4: Implementation of various security strategies for different cloud platform

Suggested Readings/ Books:
1. Raj Kumar Buyya, James Broberg, Andrezei M.Goscinski, "Cloud Computing: Principles and Paradigms",Wiley 2011
2. Anthony T. Velte, Toby J. Velte and Robert Elsenpeter, "Cloud Computing: A practical Approach", McGraw Hill, 2010.
3. Barrie Sosinsky, "Cloud Computing Bible", Wiley, 2011.
4. Judith Hurwitz, Robin Bllor, Marcia Kaufman, Fern Halper, "Cloud Computing for dummies", 2009.

Reference Books
1. Rajkumar Buyya, Christian Vecchiola, S.Thamarai Selvi, "Mastering Cloud Computing"   TMH 2013.
2. George Reese "Cloud Application Architectures", First Edition, O"Reilly Media 2009.
3. Dr. Kumar Saurabh "Cloud Computing" 2nd Edition, Wiley India 2012.

………………………………………………………………………………………………….

Course Code: BTCS 612-18	Course Title: Cloud Computing Lab	L:0;T:0; P:2	1 Credits

Detailed List of Tasks:

1. Install VirtualBox/VMware Workstation on different OS.

2. Install different operating systems in VMware.

3. Simulate a cloud scenario using simulator.

4. Implement scheduling algorithms.

5. To study cloud security management.

6. To study and implementation of identity management

7. Case Study - Amazon Web Services/Microsoft Azure/Google cloud services.

Suggested Tools –Matlab, Cloudsim

Course Code: BTCS706-18	Course Title: Distributed Databases	3L: 0T: 0P	Credits: 3

Detailed Contents:

Unit 1:
INTRODUCTION: Distributed data processing; What is a DDBS; Advantages and disadvantages of DDBS; Problem areas; Overview of database and computer network concepts
DISTRIBUTED DATABASE MANAGEMENT SYSTEM ARCHITECTURE: Transparencies in a distributed DBMS; Distributed DBMS architecture; Global directory issues.

6 hrs., CO1

Unit 2:
DISTRIBUTED DATABASE DESIGN: Alternative design strategies; Distributed design issues; Fragmentation; Data allocation.
SEMANTICS DATA CONTROL: View management; Data security; Semantic Integrity Control.
QUERY PROCESSING ISSUES: Objectives of query processing; Characterization of query processors; Layers of query processing; Query decomposition; Localization of distributed data.

10 hrs., CO1

Unit 3:

DISTRIBUTED QUERY OPTIMIZATION: Factors governing query optimization; Centralized query optimization; Ordering of fragment queries; Distributed query optimization algorithms.

TRANSACTION MANAGEMENT: The transaction concept; Goals of transaction management; Characteristics of transactions; Taxonomy of transaction models.

CONCURRENCY CONTROL: Concurrency control in centralized database systems; Concurrency control in DDBSs; Distributed concurrency control algorithms; Deadlock management.

10 hrs., CO2

Unit 4:

RELIABILITY: Reliability issues in DDBSs; Types of failures; Reliability techniques; Commit protocols; Recovery protocols.

PARALLEL DATABASE SYSTEMS: Parallel architectures; parallel query processing and optimization; load balancing.

ADVANCED TOPICS: Databases, Distributed Object Management, Multi-databases.

10 hrs., CO2,3

COURSE OUTCOMES

After completion of course, students would be able to:

CO1: Design trends in distributed systems.

CO2: Apply network virtualization in distributed environment.

CO3: Apply remote method invocation and objects.

References:

1. Principles of Distributed Database Systems, M.T. Ozsu and P. Valduriez, Prentice-Hall, 1991.
2. Distributed Database Systems, D. Bell and J. Grimson, Addison-Wesley, 1992.

………

Course Code: BTCS707-18	Course Title: Distributed Databases lab	L: T: 2P	Credits: 1

Detailed list of Tasks:

Programs may be implemented using any open source tool

Expt. 1: Installation and configuration of database packages.

Expt. 2: Creating and managing database objects (Tables, views, indexes etc.)

Expt. 3: Creating and managing database security through user management.

Expt. 4: Creating and maintaining database links.

Expt. 5: Implement Partitioning on the database tables.

Expt. 6: Implement various Transaction concurrency control methods [i.e. lock's] by executing multiple update and queries.

Expt. 7: Performance tuning of SQL queries.

Mini Project: Student has to do a project assigned from course contents in a group of two or three students. The team will have to demonstrate as well as have to give a presentation of the same.

| Course Code: BTCOE701-19 | Course Title: Big Data Analytics | L:3 T:0 P:0 | 3 Credits |

Detailed Contents:

Module I

Introduction: Big Data Overview, The rising and importance of data sciences, Big data analytics in industry verticals

Hadoop Architecture: Hadoop Architecture, Hadoop ecosystem components, Hadoop Storage: HDFS, Hadoop Processing: MapReduce Framework, Hadoop Server Roles

Module II

Data Analytics Lifecycle and methodology: Business Understanding, Data Understanding, Data Preparation, Modeling, Evaluation, Communicating results, Deployment, Data exploration & preprocessing

Module III

Data Analytics - Theory & Methods: Measures and evaluation, Supervised learning, Linear/Logistic regression, o Decision trees, Naïve Bayes, Unsupervised learning, K-means clustering, Association rules, Unstructured Data Analytics, Technologies & tools, Text mining, Web mining

Module IV

The Endgame: Opertionalizing an Analytics project, Data Visualization Techniques, Creating final deliverables

Course Outcomes
1. Describe Big Data and its importance with its applications
2. Differentiate various big data technologies like Hadoop MapReduce, Pig, Hive, Hbase and No-SQL.
3. Apply tools and techniques to analyze Big Data.
4. Design a solution for a given problem using suitable Big Data Techniques

Text Books:
1. Hadoop: The Definitive Guide by Tom White
2. Big Data Analytics: From Strategic Planning to Enterprise Integration with Tools, Techniques, NoSQL, and Graph by David Loshin
3. Machine Learning by Tom M. Mitchell

Course Code: BTCOE702-19	Course Title: Big Data Analytics Lab	L:0 T:0 P:2	1 Credits

List of Experiments:
1. Hands-on with Map Reduce: Hadoop, Hive, MapR
2. Hands-on with NoSQL Databases: S3, Hadoop Distributed File System(HDFS)
3. Hands-on with Statistical Packages
4. Hands-on with Visual Data Analysis tools

Lab Outcomes:

CO1: Perform data gathering of large data from a range of data sources.

CO2: Critically analyse existing Big Data datasets and implementations, taking practicality, and usefulness metrics into consideration.

CO3: Select and apply suitable statistical measures and analyses techniques for data of various structure and content and present summary statistics

CO4: Employ advanced statistical analytical skills to test assumptions, and to generate and present new information and insights from large datasets

……

Course Code: BTCOE703-19	Course Title : Software Defined Networks	3L:0T:0P	3Credits

Course Prerequisites:

Computer networks or related courses., C, C++, Java, or Python programming skills, Basic Linux operating system skills.

Course Objectives:

This course introduces software defined networking, an emerging paradigm in computer networking that allows a logically centralized software program to control the behavior of an entire network. Recently, SDN is being deployed in data center networks, campus networks, enterprise networks, operator networks, and is expected to play a very important role in Internet of Things (IoT) networks and 5G cellular networks. In this course, we will first introduce the concepts of SDN, its use cases and its applications to give the students an overall picture of this new technology.

Course Outcomes:
I. To define and understand terminology involved in the field of software defined networking (SDN).
II. To describe software defined architecture and open flow protocol for communication between controller and switches.
III. To provide an overview and comparison of various SDN controllers.

IV.   To design topologies using Mininet and various APIs.
V.   To develop various applications and protocols for SDN architecture.
VI.   To identify and analyse various security threats in SDN based networks.

Detailed Contents:

UNIT I:
Overview of Software Defined Networking: History and Evaluation of SDN**,** Introduction to SDN, Advantages of SDN over Traditional Network Architecture, Separation of Control and Data Plane, Use Cases of SDN.

[6hrs] (CO 1)

UNIT II:
SDN Components
How SDN Works - SDN Architecture: Data plane, Control plane, Application Plane, Southbound Interface, Northbound Interface, Pure and Hybrid openflow switches, Software and Hardware based Openflow switches, Programmable Network Hardware.

[6hrs] (CO 2)

SDN Controllers: Overview, Centralized & Distributed Controllers, Open source SDN Controllers: POX , Ryu, Floodlight, OpenDaylight, Advantages and Disadvantages of each controller**.**

[5hrs] (CO 3)

UNIT III:
OpenFlow Protocol
OpenFlow Overview- OpenFlow 1.0 and OpenFlow Basics- , OpenFlow 1.1 Additions, OpenFlow 1.2 Additions, OpenFlow 1.3 Additions, Flow table components: matching rules, Actions, Counters, OpenFlow security,   Proactive and reactive approach to insert flow table entries, Comparison of Openflow with other Southbound interfaces, OpenFlow Limitations.

[6hrs] (CO 2)

UNIT IV:
Mininet Emulation Tool
Creating Default & Custom topologies in Mininet using low level API, mid-level API, high level API, Developing Switching and Firewall Applications in Mininet.

[6hrs] (CO 4)

UNIT V:
 Programming SDN: Northbound Application Programming Interface, Current Languages and Tools, Composition of SDNs.

[5hrs] (CO 5)

SDN Security Challenges and Issues Characteristics of SDN, Security Analysis and Potential attacks in SDN, Solutions to the security issues in SDN.

[5hrs] (CO 6)

UNIT VI:
Software Defined Networks with Network Function Virtualization (NFV)
Introduction to Network Function Virtualization, History and Evaluation of NFV, NFV Architecture and its relation with SDN, Similarities and differences in SDN and NFV, NFV use cases.

[6hrs] (CO 1)

Text Books:

1. SDN: Software Defined Networks, An Authoritative Review of Network Programmability Technologies. By Thomas D. Nadeau, Gray Publisher: O'Reilly Media, August 2013, ISBN: 97814493-4230-2, ISBN 10:1-4493-4230-2.
2. Software Defined Networks: A Comprehensive Approach by Paul Goransson and Chuck Black, Morgan Kaufmann, June 2014, Print Book ISBN: 9780124166752, eBook ISBN : 9780124166844.

Suggested Books:

1. Software Defined Networking with OpenFlow., Siamak Azodolmolky, Packt Publishing, 2013.
2. Software Networks: Virtualization, SDN, 5G, Security., Guy Pujolle, Wiley, 2015.
3. Foundations of Modern Networking: SDN, NFV, QoE, IoT, and Cloud, William Stallings Publisher: Addison-Wesley 2015 ISBN: 9780134175393, William Stallings Publisher.

………………………………………………………………………………………………………

Course Code: BTCOE704-19	Course Title: Software Defined Networks Lab	L:0 T:0 P:2	1 Credits

Detailed List of Tasks:

1. Installation of Mininet emulation tool in Virtual Box.
2. Build different topologies (single, reversed, linear, tree) using the Mininet CLI.
3. Insert, delete and modify flow entries in sdn switches using ovs-ofctl.
3. Create custom toplogies in mininet using low-level API.
4. Create custom toplogies in mininet using mid-level API.
5. Create custom toplogies in mininet using high-level API.
6. Connect mininet network with a remote controller.
7. Create and run hub application using pox controller.
8. Create and run switch application using pox controller