Study Scheme & Syllabus
of

Bachelor of Technology
Computer Science & Engineering

B. Tech (CSE)

Batch 2018 onwards

By

Department of Academics

IK Gujral Punjab Technical University

Bachelor of Technology in Computer Science & 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
1BTHU102-18	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		
BTES301-18	Engineering Science Course	Digital Electronics	3	0	0	40	60	100	3
BTCS301-18	Professional Core Courses	Data structure & Algorithms	3	1	0	40	60	100	3
BTCS302-18	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
HSMC101/102 -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/Unsatisfactory
Total			**15**	**3**	**12**	**380**	**420**	**800**	**21**

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**

Fifth Semester

Course Code	Course Type	Course Title	Load Allocations			Marks Distribution		Total Marks	Credits
			L	T	P	Internal	External		
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	1	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
BTCS YYY-18	Professional Elective Courses	Elective-II	3	0	0	40	60	100	3
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
BTCS YYY-18	Professional Elective Courses	Elective-II lab	0	0	2	30	20	50	1
		Industrial Training	0	0	0	60	40	100	Satisfactory/Unsatisfactory
Total			**18**	**1**	**12**	**450**	**500**	**950**	**24**

Sixth Semester

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

Seventh Semester

Course Code	Course Type	Course Title	Load Allocations			Marks Distribution		Total Marks	Credits
			L	T	P	Internal	External		
BTCS VVV-18	Professional Elective	Elective-V	3	0	0	40	60	100	3
BTCS TTT-18	Professional Elective Courses	Elective-VI	3	0	0	40	60	100	3
BTOE ***	Open Elective Courses	Open Elective-II	3	0	0	40	60	100	3
BTOE ***	Open Elective Courses	Open Elective- III	3	0	0	40	60	100	3
BTCS 701-18	Professional Core Courses	Machine Learning	3	0	0	40	60	100	3
BTCS 702-18	Project	Project-II	0	0	12	120	80	200	6
BTCS 703-18	Professional Core Courses	Machine Learning Lab	0	0	2	30	20	50	1
BTCS VVV-18	Professional Elective	Elective-V lab	0	0	2	30	20	50	1
BTCS TTT-18	Professional Elective Courses	Elective-VI lab	0	0	2	30	20	50	1
Total			**15**	**0**	**18**	**410**	**440**	**850**	**24**

Eighth Semester

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

Course Code	Course Type	Course Title	Load Allocations			Marks Distribution		Total Marks	Credits
			L	T	P	Internal	External		
BTCS 801-18	Professional Core Courses	Social Network Analysis	2	0	2	40	60	100	3
BTCS 802-18	Professional Core Courses	Cyber Attacks	2	0	4	40	60	100	3
BTCS 803-18	Professional Core Courses	Deep Learning	3	0	2	40	60	100	4
BTCS 804-18	Project	Project-III	0	0	12	120	80	200	6
Total			**7**	**0**	**20**	**240**	**260**	**500**	**16**

LIST OF ELECTIVES

Elective-I

BTCS 508-18 Programming in Java
BTCS 509-18 Web and Open Source Technologies
BTCS 510-18 Programming in Python
BTCS 511-18 Programming in Java lab
BTCS 512-18 Web and Open Source Technologies lab
BTCS 513-18 Programming in Python Lab

Elective-II

BTCS 514-18 Mobile Application Development
BTCS 515-18 Computer Graphics
BTCS 516-18 Internet of Things
BTCS 517-18 Mobile Application Development lab
BTCS 518-18 Computer Graphics Lab
BTCS 519-18 Internet of Things Lab

Elective-III

BTCS 604-18 Cyber Security
BTCS 605-18 Data Mining
BTCS 606-18 Cloud Computing
BTCS 607-18 Cyber Security Lab
BTCS 608-18 Data Mining lab
BTCS 609-18 Cloud Computing lab

Elective-IV

BTCS 610-18 Information Theory and Coding
BTCS 611-18 Data Science
BTCS 612-18 Soft Computing
BTCS 613-18 Information Theory and Coding lab
BTCS 614-18 Data Science Lab
BTCS 615-18 Soft Computing lab

Elective-V

BTCS 703-18 Quantum Computing
BTCS 704-18 Ad-Hoc and Sensor Networks
BTCS 705-18 Speech and Natural Language Processing
BTCS 706-18 Quantum Computing lab
BTCS 707-18 Ad-Hoc and Sensor Networks lab
BTCS 708-18 Speech and Natural Language Processing lab

Elective-VI

BTCS 709-18 Block Chain Technologies
BTCS 710-18 Software Defined Networking
BTCS 711-18 Image Processing
BTCS 712-18 Block Chain Technologies lab
BTCS 713-18 Software Defined Networking Lab
BTCS 714-18 Image Processing lab

Open electives offered by the department:

Courses of odd semesters:

BTCS301-18	**Enterprise Resource Planning**
BTCS302-18	**Cyber laws and IPR**
BTCS501-18	**Database Management System**
BTCS504-18	**Computer Networks**
BTCS701-18	**Artificial Intelligence**
BTCS709-18	**Block Chain Technologies**
BTCS710-18	**Data Science**

Courses of even semesters:

BTES401-18	**Computer Organisation & Architecture**
BTCS402-18	**Operating System**
BTCS601-18	**Internet of Things**
BTCS604-18	**Cyber Security**

LIST 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 Systems	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

MINOR DEGREE IN COMPUTER SCIENCE ENGG.(Credits required 20 from Core+Electives/MOOCS*)

List of Core Courses: Minimum of 2 courses must be opted, other than studied in regular course.

Course Code	Type of Course	Course Title	Hours per Week			Marks Distribution		Total Marks	Credits
			L	T	P	Internal	External		
BTCS301-18 & BTCS303- 18	PCC	Data structure Theory & Lab	3	0	4	40T+30 P	60T+20 P	150	5
BTCS302-18 & BTCS304- 18	PCC	Object Oriented Programming Theory & Lab	3	0	4	40T+30 P	60T+20 P	150	5
BTCS602-18 & BTCS605- 18	PCC	Computer networks Theory & Lab	3	0	2	40T+30 P	60T+20 P	150	4
BTCS402-18 & BTCS404- 18	PCC	Operating system Theory & Lab	3	0	4	40T+30 P	60T+20 P	150	5
BTES401-18 & BTCS402- 18	ESC	Computer Organisation and architecture Theory & Lab	3	0	2	40T+30 P	60T+20 P	150	4
BTCS501-18 & BTCS504-18	PCC	Database Management system Theory & Lab	3	0	4	40T+30 P	60T+20 P	150	5

*List of Courses through MOOCS will be provided every six months through BOS/ MOOCS Coordinator; each course must be of minimum 12 weeks and of 4 credits after submission of successful exam in that course.

List of Electives: 3 courses can be opted, other than studied in regular course.

Course Code	Type of Course	Course Title	Hours per Week			Marks Distribution		Total Marks	Credits
			L	T	P	Internal	External		
BTCS507-18 & BTCS511- 18	ELECTIVE	Web Technologies Theory & Lab	3	0	2	40T+30 P	60T+20 P	150	4
BTCS608-18 & BTCS616-18	ELECTIVE	Machine Learning Theory & Lab	3	0	2	40T+30 P	60T+20 P	150	4
BTCS613-18 & BTCS621- 18	ELECTIVE	Cloud computing Theory & Lab	3	0	2	40T+30 P	60T+20 P	150	4
BTCS707-18 & BTCS715-18	ELECTIVE	Adhoc and Sensor network Theory & Lab	3	0	2	40T+30 P	60T+20 P	150	4
BTCS804-18 & BTCS808- 18	ELECTIVE	Data Analysis Theory & Lab	3	0	2	40T+30 P	60T+20 P	150	4
BTCS509-18 & BTCS513- 18	ELECTIVE	Computer Graphics Theory & Lab	3	0	2	40T+30 P	60T+20 P	150	4
BTCS606-18 & BTCS614- 18	ELECTIVE	Mobile Application Development Theory & Lab	3	0	2	40T+30 P	60T+20 P	150	4
BTCS612-18 & BTCS620- 18	ELECTIVE	Data Mining Theory & Lab	3	0	2	40T+30 P	60T+20 P	150	4
BTCS702-18 & BTCS710- 18	ELECTIVE	Information Theory & Coding Theory & Lab	3	0	2	40T+30 P	60T+20 P	150	4
BTCS704-18 & BTCS712- 18	ELECTIVE	Soft Computing Theory & Lab	3	0	2	40T+30 P	60T+20 P	150	4

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:

Section-A

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: BTCS301-18	**Course Title:** Data Structure & Algorithms	**3L:0T:P**	**3Credits**

Detailed Contents:

Module 1: 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)

Module 2: 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)

Module 3: **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)

Module 4: 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)

Module 4: 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, SartajSahni, 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: BTCS302-18	**Course Title:** Object Oriented Programming	**3L:0T:0P**	**3Credits**

Pre-requisites: Programming in C

Detailed Contents:

Module 1: Introduction
Overview of C++, Sample C++ program, Different data types, operators, expressions, and statements, arrays and strings, pointers & function components, recursive functions, user - defined types, function overloading, inline functions, Classes & Objects – I: classes, Scope resolution operator, passing objects as arguments, returning objects, and object assignment.

[8 hrs] (CO1)

Module 2: Classes & Objects –II
Constructors, Destructors, friend functions, Parameterized constructors, Static data members, Functions, Arrays of objects, Pointers to objects, this pointer, and reference parameter, Dynamic allocation of objects, Copy constructors, Operator overloading using friend functions, overloading.

[8 hrs] (CO1, CO2)

Module 3: Inheritance
Base Class, Inheritance and protected members, Protected base class inheritance, Inheriting multiple base classes, Constructors, Destructors and Inheritance, Passing parameters to base class constructors, Granting access, Virtual base classes.

[8 hrs] (CO3, CO4)

Module 4: Virtual functions, Polymorphism
Virtual function, calling a Virtual function through a base class reference, Virtual attribute is inherited, Virtual functions are hierarchical, pure virtual functions, Abstract classes, Using

virtual functions, Early and late binding.

[8 hrs] (CO3, CO4)

Module 5: Exception Handling

Basics of exception handling, exception handling mechanism, throwing mechanism, catching mechanism, I/O System Basics, File I/0: Exception handling fundamentals, Exception handling options. C++ stream classes, Formatted I/O, fstream and the File classes, Opening and closing a file, Reading and writing text files.

[10 hrs] (CO5)

Course Outcomes:

The student will be able to:

1. Identify classes, objects, members of a class and the relationships among them needed to solve a specific problem;
2. Demonstrate the concept of constructors and destructors. And create new definitions for some of the operators;
3. Create function templates, overload function templates;
4. Understand and demonstrate the concept of data encapsulation, inheritance, polymorphism with virtual functions; &
5. Demonstrate the concept of file operations, streams in C++ and various I/O manipulators.

Suggested Books:

1. E. Balagurusamy, Object Oriented Programming with C++, Tata McGraw Hill.

Reference Books:

1. Stanley B.Lippmann, JoseeLajoie: C++ Primer, 4th Edition, Addison Wesley, 2012.
2. Herbert Schildt: The Complete Reference C++, 4th Edition, Tata McGraw Hill, 2011.

Course Code: BTCS303-18	**Course Title:** Data Structure & Algorithms Lab	**0L:0T:4P**	**2Credits**

List of Experiment:

Task 1: Write a program to insert a new element at end as well as at a given position in an array.

Task 2: Write a program to delete an element from a given whose value is given or whose position is given.

Task 3: Write a program to find the location of a given element using Linear Search.

Task 4: 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.

Task 9: Write a program to implement insertion and deletion operations in a queue using linear

array.

Task 10: 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

Task 11: 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.
Task 14: Program to sort an array of integers in ascending order using bubble sort.
Task 15: 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.
Task 17: 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.

Reference Books:

1. "Data Structures with C (Schaum's Outline Series)", Seymour Lipschutz, 1st edition, McGraw Hill Education.

Course Code: BTCS304-18	**Course Title:** Object Oriented Programming Lab	**0L:0T:4P**	**2Credits**

List of Experiment:

Task 1: Write a program that uses a class where the member functions are defined inside a class.
Task 2: Write a program that uses a class where the member functions are defined outside a class.
Task 3: Write a program to demonstrate the use of static data members.

Task 4: Write a program to demonstrate the use of const data members.
Task 5: Write a program to demonstrate the use of zero argument and parameterized constructors.
Task 6: Write a program to demonstrate the use of dynamic constructor.
Task 7: Write a program to demonstrate the use of explicit constructor.
Task 8: Write a program to demonstrate the use of initializer list.
Task 9: Write a program to demonstrate the overloading of increment and decrement operators.
Task 10: Write a program to demonstrate the overloading of memory management operators.
Task 11: Write a program to demonstrate the typecasting of basic type to class type.
Task 12: Write a program to demonstrate the typecasting of class type to basic type.
Task 13: Write a program to demonstrate the typecasting of class type to class type.
Task 14: Write a program to demonstrate the multiple inheritances.
Task 15: Write a program to demonstrate the runtime polymorphism.
Task 16: Write a program to demonstrate the exception handling.
Task 17: Write a program to demonstrate the use of class template.
Task 18: Write a program to demonstrate the reading and writing of mixed type of data.

Lab Outcomes:

The student will be able to:

1. Develop classes incorporating object-oriented techniques;
2. Design and implement object-oriented concepts of inheritance and polymorphism;
3. Illustrate and implement STL class of containers and need for exceptions to handle errors for object oriented programs; &
4. Design and implement any real world based problem involving GUI interface using object-oriented concepts.

Reference Books:

1. Stanley B. Lippmann, JoseeLajoie: C++ Primer, 4th Edition, Addison Wesley, 2012.
2. E. Balagurusamy, Object Oriented Programming with C++, Tata McGraw Hill.

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

Detailed Contents:

Module 1:

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

[CO1, CO2] **(12Hrs)**

Module 2:

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.)**

Module 3:

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

[CO4**] (12 hrs.)**

Module 4:

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] **(12 hrs.)**

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

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.

Textbooks/References:

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, Prentice Hall India, 1995.

Development of Societies
Course code: HSMC101-18

Credits: 3

COURSE TOPIC

2.1 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

2.2 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

2.3 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.

3. READINGS

3.1 TEXTBOOK:
3.2 *REFERENCE BOOKS:

4. OTHER SESSIONS

4.1 *TUTORIALS:
4.2 *LABORATORY:
4.3 *PROJECT: Possible projects in this course could be

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

PHILOSOPHY Course
code: HSMC102-18

Credits: 3

COURSE TOPICS:

2.1 Unit 1:

The difference between knowledge (Vidya) and Ignorance (Avidya):
a. Upanishads;
b. Six systems orthodox and Heterodox Schools of Indian Philosophy.
c. Greek Philosophy:

2.2 Unit 2:

Origin of the Universe:

- NasidiyaSukta: "Who really knows?"

- Brhadaranyaka Upanishad; Chandogya Upanishad: Non-self, Self, real and unreal.
- Taittiriya Upanishad: SikshaValli.
- 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.

2.3 Unit 3:
Knowledge as Power: Francis Bacon. Knowledge as both power and self-realization in Bagavad Gita.

2.4 Unit 4:
Knowledge as oppression: M. Foucault. Discrimination between Rtam and Satyam in Indian Philosophy.

2.5 Unit 5:
Knowledge as invention: Modern definition of creativity; scientific activity in the claim that science invents new things at least through technology.

2.6 Unit 6:
Knowledge about the self, transcendental self; knowledge about society, polity and nature.

2.7 Unit 7:
Knowledge about moral and ethics codes.

2.8 Unit 8:
Tools of acquiring knowledge: Tantrayuktis, a system of inquiry (Caraka, Sushruta, Kautilya, Vyasa)

3. 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 Press.
6. Plato, Symposium, Hamilton Press.
7. Kautilya Artha Sastra. Penguin Books, New Delhi.
8. Bacon, Nova Orgum
9. Arnold, Edwin. The Song Celestial.
10. Foucault, Knowledge/Power.
11. Wildon, Anthony, System of Structure.
12. Lele, W.K. The Doctrine of Tantrayukti. Varanasi: Chowkamba Series.
13. Dasgupta, S. N. History of Indian Philosophy, MotilalBanasidas, Delhi.
14. Passmore, John, Hundred Years of Philosophy, Penguin.

4. OTHER SESSIONS:
4.1 Mode of Conduct

5. 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.

6. 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:BTES301-18	**Course Title:** Digital Electronics	**3L:0T:0P**	**3Credits**

Detailed Contents:

Module 1:
NUMBER SYSTEMS: Binary, Octal, Decimal, Hexadecimal. Number base conversions, 1's, 2's complements, signed Binary numbers. Binary Arithmetic, Binary codes: Weighted BCD, Gray code, Excess 3 code, 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, Karnaugh map 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 into 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 of 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 McGraw HillPublishing CompanyLimited, New Delhi, 2003.
3. R.P.Jain, **Modern Digital Electronics**, 3 ed., Tata McGraw–Hill publishing company limited, New Delhi, 2003.
4. Thomas L. Floyd, **Digital Fundamentals**, Pearson Education, Inc, New Delhi, 2003
5. Ronald J. Tocci, Neal S. Widmer, Gregory L. Moss, **Digital System - Principles and Applications**, PearsonEducation.
6. Ghosal, Digital **Electronics**, Cengage Learning.

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

List of Experiments:

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

Fourth Semester

Course Code: BTES401-18	**Course Title:** Computer Organization & Architecture	**3L:0T:0P**	**3Credits**

Pre-requisites: Digital Electronics

Detailed Contents:

Module 1: 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 of 8085 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)

Module 2: 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)

Module 3: Pipelining

Basic concepts of pipelining, throughput and speedup, pipeline hazards.

Parallel Processors: Introduction to parallel processors, Concurrent access to memory and cache coherency.

[10 hrs] (CO5)

Module 4: 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 analyze its 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 CarlHamacher, 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.

Course Code: BTCS402-18	**Course Title:** Operating Systems	**3L:0T:0P**	**3Credits**

Detailed Contents:

Module 1: 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)

Module 2: 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)

Module 3: 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)

Module 4: Deadlocks

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

[8 hrs] (CO3)

Module 5: 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)

Module 6: 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:0T:0P**	**3Credits**

Pre-requisites: Data Structures

Detailed Contents:

Module 1: Introduction
Characteristics of algorithm. Analysis of algorithm: Asymptotic analysis of complexity bounds – best, average and worst-case behavior; Performance measurements of Algorithm, Time and space trade-offs, Analysis of recursive algorithms through recurrence relations: Substitution method, Recursion tree method and Masters' theorem.

[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 problems 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 writethe 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. Introduction to Algorithms, 4TH Edition, Thomas H Cormen, Charles E Lieserson, Ronald

L Rivest and Clifford Stein, MIT Press/McGraw-Hill.
2. Data Structures and Algorithms in C++, Weiss, 4th edition, Pearson.
3. Fundamentals of Computer Algorithms – E. Horowitz, Sartaj Saini, Galgota Publications.

Reference Books

1. Algorithm Design, 1stEdition, Jon Kleinberg and ÉvaTardos, 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: BTES402-18	**Course Title:** Computer Organization & Architecture Lab	**0L:0T:2P**	**1Credits**

List of Experiment:

Task 1: Computer Anatomy- Memory, Ports, Motherboard and add-on cards.
Task 2: Dismantling and assembling PC.
Task 3: Introduction to 8085 kit.
Task 4: 2. 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 of ten 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's complement 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 Publications.

Course Code: BTCS404-18	**Course Title:** Operating Systems Lab	**0L:0T:4P**	**2Credits**

List of Experiment:

Task 1: 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.

Task 6: 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. Woodhull and Andrew S. Tanenbaum, Pearson Education.

Course Code: BTCS405-18	**Course Title:** Design and Analysis of Algorithms Lab	**0L:0T:4P**	**2Credit**

List of Experiment:

Task 1: Code and analyze solutions to following problem with given strategies:
i. Knap Sack using greedy approach
ii. Knap Sack using dynamic approach

Task 2: Code and analyze to find an optimal solution to matrix chain multiplication using dynamic programming.

Task 3: Code and analyze to find an optimal solution to TSP using dynamic programming.

Task 4: 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 a maze.

Task 5: 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.

Task 6: Code and analyze to find shortest paths in a graph with positive edge weights using Dijkstra's algorithm.

Task 7: Code and analyze to find shortest paths in a graph with arbitrary edge weights using Bellman-Ford algorithm.

Task 8: Code and analyze to find shortest paths in a graph with arbitrary edge weights using Flyods' algorithm.

Task 9: Code and analyze to find the minimum spanning tree in a weighted, undirected graph using Prims' algorithm

Task 10: Code and analyze to find the minimum spanning tree in a weighted, undirected graph using Kruskals' algorithm.

Task 11: 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, 4th edition, Pearson
2. Data Structures and Algorithms using Python and C++, David M. Reed and John Zelle, 2009 edition (available as e book), Franklin Beedle & Associates.

UNIVERSAL HUMAN VALUES 2: UNDERSTANDING HARMONY

Course code: HSMC122-18

Credits: 3

COURSE TOPICS:

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

Module 1: 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.

Module 2: Understanding Harmony in the Human Being - Harmony in Myself!

7. Understanding human being as a co-existence of the sentient 'I' and the material 'Body'
8. Understanding the needs of Self ('I') and 'Body' - happiness and physical facility
9. Understanding the Body as an instrument of 'I' (I being the doer, seer and enjoyer)
10. Understanding the characteristics and activities of 'I' and harmony in 'I'
11. Understanding the harmony of I with the Body: Sanyam and Health; correct appraisal of Physical needs, meaning of Prosperity in detail
12. Programs to ensure Sanyam and Health.

Include practice sessions to discuss the role others have played in making material goods available to me. Identifying from one's own life. Differentiate between prosperity and accumulation. Discuss program for ensuring health vs dealing with disease.

Module 3: Understanding Harmony in the Family and Society- Harmony in Human-Human Relationship

13. 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.

14. Understanding the meaning of Trust; Difference between intention and competence
15. Understanding the meaning of Respect, Difference between respect and differentiation; the other salient values in relationship.
16. Understanding the harmony in the society (society being an extension of family): Resolution, Prosperity, fearlessness (trust) and co-existence as comprehensive Human Goals.
17. 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

18. Understanding the harmony in the Nature
19. Interconnectedness and mutual fulfilment among the four orders of nature - recyclability and self-regulation in nature
20. Understanding Existence as Co-existence of mutually interacting units in all- pervasive space
21. 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.

Module 5: Implications of the above Holistic Understanding of Harmony on Professional Ethics

22. Natural acceptance of human values
23. Definitiveness of Ethical Human Conduct
24. Basis for Humanistic Education, Humanistic Constitution and Humanistic Universal Order
25. 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.
26. Case studies of typical holistic technologies, management models and production systems.
27. 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.
28. 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 Intl. Publishers, New Delhi, 2004.
3. The Story of Stuff (Book).
4. The Story of My Experiments with Truth - by Mohandas Karamchand Gandhi
5. Small is Beautiful - E. F Schumacher.
6. Slow is Beautiful - Cecile Andrews
7. Economy of Permanence - J C Kumarappa
8. Bharat Mein Angreji Raj –Pandit Sunderlal
9. Rediscovering India - by Dharampal
10. Hind Swaraj or Indian Home Rule - by Mohandas K. Gandhi
11. India Wins Freedom - Maulana Abdul Kalam Azad
12. Vivekananda - Romain Rolland (English)
13. 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-	**L:2; T:0; P:0**	**0Credits**

Detailed Contents:

Module 1: 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 ground water, floods, drought, conflicts over water, dams-benefits and problems.
c) Mineral resources: Use and exploitation, environmental effects of extracting and using mineral resources, case studies.
d) Food resources: World food problems, changes caused by agriculture and overgrazing, effects of modern agriculture, fertilizer-pesticide problems, water logging, salinity, case studies.
e) Energy resources: Growing energy needs, renewable and non-renewable energy sources, use of alternate energy sources. Case studies.
f) Land resources: Land as a resource, land degradation, man induced landslides, soil erosion and desertification.
    - Role of an individual in conservation of natural resources.
    - Equitable use of resources for sustainable lifestyles.

Module 2: Ecosystems

Concept of an ecosystem. Structure and function of an ecosystem. Food chains, food webs 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)

Module 3: Biodiversity and its conservation

a) Introduction – Definition: genetic, species and ecosystem diversity.
b) Biodiversity at global, National and local levels.
c) India as a mega-diversity nation
d) Hot-sports of biodiversity.
e) Threats to biodiversity: habitat loss, poaching of wildlife, man-wildlife conflicts.
f) Endangered and endemic species of India

Module 4: Social Issues and the Environment

a) From Unsustainable to Sustainable development
b) Resettlement and rehabilitation of people; its problems and concerns.
c) Environmental ethics: Issues and possible solutions.
d) Climate change, global warming, acid rain, ozone layer depletion, Nuclear accidents and holocaust. Case Studies.
e) Public awareness.

*ACTIVITIES

Nature club (bird watching, 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 be encouraged 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
i) To live with some eminent environmentalist for a week or so to understand his work
j) To work in kitchen garden for mess
k) To know about the different varieties of plants
l) Shutting down the fans and ACs of the campus for an hour or so
m) Visit to a local area to document environmental assets river/forest/grassland/hill/mountain/lake/Estuary/Wet lands
n) Visit to a local polluted site-
Urban/Rural/Industrial/Agricultural n) Visit to a Wildlife sanctuary, National Park or Biosphere Reserve

Suggested Readings

1. Agarwal, K.C. 2001 Environmental Biology, Nidi Publ. Ltd. Bikaner.
2. BharuchaErach, The Biodiversity of India, Mapin Publishing Pvt. Ltd., Ahmedabad – 380 013, India, Email:mapin@icenet.net (R)
3. Brunner R.C., 1989, Hazardous Waste Incineration, McGraw Hill Inc. 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. Press 1140p.
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 Stadards, 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,

USA 499p

Course Code: HSMC101- 18	**Course Title:** Development of Societies	**3L:0T: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.
5. Gandhian idea of development. Swaraj and Decentralization.

PROJECT: Possible projects in this course could be

a) Interact with local communities and understand their issues.
b) Study local cottage industry and agricultural practices. Role of engineering and specialized knowledge.
c) 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	**3L:0T:0P**	**3Credits**

Detailed Contents:

Unit 1:

The difference between knowledge (Vidya) and Ignorance (Avidya):

a. Upanishads;
b. Six systems orthodox and Heterodox Schools of Indian Philosophy.
c. Greek Philosophy:

Unit 2:

Origin of the Universe:

- Nasidiya Sukta: "Who really knows?"
- Brhadaranyaka Upanishad; Chandogya Upanishad: Non-self, Self, real and unreal.

- Taittiriya Upanishad: SikshaValli.
- 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.

Unit 3:
Knowledge as Power: Francis Bacon. Knowledge as both power and self-realization in Bagavad Gita.

Unit 4:
Knowledge as oppression: M. Foucault. Discrimination between Rtam and Satyam in Indian Philosophy.

Unit 5:
Knowledge as invention: Modern definition of creativity; scientific activity in the claim that science invents new things at least through technology.

Unit 6:
Knowledge about the self, transcendental self; knowledge about society, polity and nature.

Unit 7:
Knowledge about moral and ethics codes.

Unit 8:
Tools of acquiring knowledge: Tantrayuktis, a system of inquiry (Caraka, Sushruta, Kautilya, Vyasa)

READINGS

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

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 livedexperience can be considered.

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

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. **CO1, CO2**

Module 2:

Basic counting techniques-inclusion and exclusion, pigeon-hole principle, permutation and combination. **CO3**

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. **CO3, CO4**

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 **CO4**

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. CO5

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 Its Application to Computer Science", TMG Edition, Tata Mcgraw-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. To be able to express logical sentence in terms of predicates, quantifiers, and logical connectives
2. To derive the solution for a given problem using deductive logic and prove the solution based on logical inference
3. For a given a mathematical problem, classify its algebraic structure
4. To evaluate Boolean functions and simplify expressions using the properties of Boolean algebra
5. To develop the given problem as graph networks and solve with techniques of graph theory.

Fifth Semester

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

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] (CO 1, 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] (CO 2, 4)

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

[3hrs] (CO 3)

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] (CO 5, 6)

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 4, 6)

Course Outcomes:

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

CO 1: For a given query write relational algebra expressions for that query and optimize the developed expressions

CO 2: For a given specification of the requirement design the databases using ER method and normalization.

CO 3: For a given specification construct the SQL queries for Open source and **Commercial**

DBMS - MYSQL, ORACLE, and DB2.

CO 4: For a given query optimize its execution using Query optimization algorithms

CO 5: For a given transaction-processing system, determine the transaction atomicity, consistency, isolation, and durability.

CO 6: Implement the isolation property, including locking, time stamping based on concurrency control and Serializability of scheduling.

Suggested books:

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

Suggested reference books

1 "Principles of Database and Knowledge – Base Systems", Vol 1 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 : BTCS 502-18	**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. 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. 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. Context-sensitive languages: Context-sensitive grammars (CSG) and languages, linear bounded automata and equivalence with CSG. 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. Undecidability:

Church-Turing thesis, universal Turing machine, the universal and diagonalization languages, reduction between languages and Rice s theorem, undecidable problems about languages.

[4hrs] (CO 1)

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 off finite automata.

[8hrs] (CO 2, 3)

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] (CO 4, 5)

Module 4:

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

[6hrs] (CO 5)

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 5.6)

Module6:

Undecidability Church-Turing thesis, universal Turing machine, the universal and diagonalization languages, reduction between languages and Rices theorem, undecidable problems about languages.

[8hrs] (CO 7)

Course Outcomes:

At the end of the course the student should be able to:
CO 1: Write a formal notation for strings, languages and machines.
CO 2: Design finite automata to accept a set of strings of a language.
CO 3: For a given language determine whether the given language is regular or not.
CO 4: Design context free grammars to generate strings of context free language.

CO 5: Determine equivalence of languages accepted by Push Down Automata and languages generated by context free grammars
CO 6: Write the hierarchy of formal languages, grammars and machines.
CO 7: Distinguish between computability and non-computability and Decidability and undecidability.

Suggested books

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

Suggested 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: BTCS 503-18	**Course Title :** Software Engineering	**3L:0T:0P**	**3 Credits**

Detailed Contents:

UNIT 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.

[8hrs] (CO 1)

UNIT 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.

[6hrs] (CO 2)

UNIT 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.

[8 hrs] (CO 3)

UNIT 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

[8 hrs] (CO 4)

UNIT 5:

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

[6 hrs] (CO 5)

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: Analyze 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

Suggested Readings/ Books:

1. Roger Pressman, "Software Engineering: A Practitioners Approach, (6th Edition), McGraw Hill,
1. 1997.
2. Sommerville, "Software Engineering, 7th edition", Adison Wesley, 1996.
3. Watts Humphrey, "Managing software process", Pearson education, 2003.
4. James F. Peters and Witold Pedrycz, "Software Engineering – An Engineering Approach", Wiley.
5. Mouratidis and Giorgini. "Integrating Security and Software Engineering–Advances and Future", IGP. ISBN – 1-59904-148-0.
6. Pankaj Jalote, "An integrated approach to Software Engineering", Springer/Narosa.

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

Detailed 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.

[8 hrs] (CO 1)

Module 2: Data Link Layer and Medium Access SubLayer

Error Detection and Error Correction- Fundamentals, Block coding, Hamming Distance, CRC; Flow Control and Error control protocols-Stop and Wait, Goback–NARQ, Selective Repeat ARQ, Sliding Window, Piggy backing, Random Access, Multiple access protocols-Pure ALOHA, Slotted ALOHA, CSMA/CDCDMA/CA.

[10 hrs] (CO 2)

Module 3: Network Layer

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

[8 hrs] (CO 3)

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] (CO 3)

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] (CO 4)

Course Outcomes:

The student will be able to:

CO 1: Explain the functions of the different layer of the OSI Protocol

CO 2: Describe the function of each block of wide-area networks (WANs), local area networks (LANs) and Wireless LANs (WLANs);

CO 3: Develop the network programming for a given problem related TCP/IP protocol

CO 4: Configure DNS DDNS, TELNET, EMAIL, File Transfer Protocol (FTP), WWW, HTTP, SNMP, Bluetooth, Firewalls using open source available software and tools.

SuggestedBooks

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. Internet working with TCP/IP, Volume1, 6th Edition Douglas Comer, Prentice Hall of India.
3. TCP/IP Illustrated, Volume1, W. Richard Stevens, Addison-Wesley, United States of America.

Elective-I

Course Code: BTCS508-18	**Course Title:** Programming in JAVA	**L:3 ;T:0; P:0**	**3 Credits**

Detailed Contents:

Unit 1:
Overview: Object oriented programming principles, Java essentials, java virtual machine, program structure in java
Java class libraries, Data types, Variables and Arrays, Data types and casting, automatic type promotion in expressions, arrays.
Operators and Control Statements: Arithmetic operators, bit wise operators, relational operators, Boolean logical operators, the? Operator, operator precedence
Java's selection statements, iteration statements, jump statements.

CO 1

UNIT 2:

Introduction to Classes: Class fundamentals, declaring class, creating objects
Introducing methods: method declaration, overloading, using objects as parameters, recursion Constructors, this keyword, garbage collection, the finalization

CO 1

UNIT 3:

Inheritance: Inheritance basics, using super and final, method overriding, dynamic method dispatch, Abstract Class
Interface: variables and extending Interfaces
Package: Creating and importing packages, Package access protection,
Exception Handling: Exception handling fundamentals, Exception types, Uncaught Exceptions Using try and catch, multiple catch clauses, nested try statements, throw, Java's built-in exceptions.

CO 1,2

UNIT 4:
Multithreaded Programming: The Java thread model, the main thread, creating thread, creating multiple threads, using is Alive () and join (), Thread priorities, synchronization, Inter thread communications, suspending resuming and stopping threads.

CO 3

UNIT5:
I/O: I/O Basics, Reading Console Input, Writing Console Output, Reading and Writing Files
Applets: Applet Fundamentals, Applet Architecture, The HTML Applet tag, Passing parameters to Applets.
Networking: Networking basics, Java and the Net, TCP/IP Client Sockets URL, URL Connection, TCP/IP Server Sockets, Database connectivity.

CO 4

Course Outcomes:

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

CO1: Understand the features of Java such as operators, classes, objects, inheritance, packages and exception handling

CO2: Learn latest features of Java like garbage collection, Console class, Network interface, APIs

CO3: Acquire competence in Java through the use of multithreading, applets

CO4: Get exposure to advance concepts like socket and database connectivity.

Suggested Readings/Books

1. Herbert Schildt, The Complete Reference Java 2, McGraw-Hill.
2. Joyce Farrell, Java for Beginners, Cengage Learning.
3. Deitel and Deitel, Java: How to Program, 6th Edition, Pearson Education.
4. James Edward Keogh, Jim Keogh, J2EE: The complete Reference, Mc Graw Hill
5. Khalid A. Mughal, Torill Hamre, Rolf W. Rasmussen, Java Actually, Cengage Learning.
6. Shirish Chavan, Java for Beginners, 2nd Edition, Shroff Publishers.

Course Code: BTCS 509-18	**Course Title:** Web and Open Source Technologies	**L:3; T:0; P:0**	**3 Credits**

Detailed Contents:

Introduction to WWW: Protocols and programs, secure connections, application and development tools, the web browser, Server, choices, setting up UNIX and Linux web servers, Logging users, dynamic IP

Web Design: Web site design principles, planning the site and navigation

Introduction to HTML: The development process, Html tags and simple HTML forms, web site structure

Introduction to XHTML: XML, Move to XHTML, Meta tags, Character entities, frames and frame sets, inside browser.

Style sheets: Need for CSS, introduction to CSS, basic syntax and structure, using CSS, background images, colors and properties, manipulating texts, using fonts, borders and boxes, margins, padding lists, positioning using CSS, CSS2

JavaScript: Client side scripting, Javascript, how to develop Javascript, simple Javascript, variables, functions, conditions, loops and repetition.

Advance script, Javascript and objects, Javascript own objects, the DOM and web browser environments, forms and validations

DHTML: Combining HTML, CSS and Javascript, events and buttons, controlling your browser

CO 1

Ajax: Introduction, HTTP request, XHttpResponse, AJAX Server Script, AJAX Database, Advantages & disadvantages, Purpose of it, Ajax based web application, alternatives of Ajax

XML: Introduction to XML, uses of XML, simple XML and XML key components, DTD and Schemas, well formed, using XML with application.XML, XSL and XSLT. Introduction to XSL, XML transformed simple example, XSL elements, transforming with XSLT

CO 2

PHP: Starting to script on server side, syntax, statements, operators, Arrays, function and forms sessions, E-mail, PHP and AJAX, advance PHP

MySQL Databases: Basic command with PHP examples, Connection to server, creating database, selecting a database, listing database, listing table names creating a table, inserting data, altering tables, queries, deleting database, deleting data and tables, PHPmyadmin and database bugs.

JavaScript Library & Web-Framework:

Jquery: Introduction, Why jQuery, jQuery methods for DOM manipulation, jQuery methods for CSS manipulation, jQuery AJAX Methods (Asynchronous JavaScript and XML)

AngularJS: Fundamental structural concepts of AngularJS, AngularJS Directives, AngularJS Expressions, Use of custom attributes in HTML, introduction to modules and controllers, form validation using validation rules, Server Communication & Data Binding techniques.

CO 3

Course Outcomes:

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

CO 1: Students are able to develop a dynamic webpage by the use of java script and DHTML.

CO 2: Students will be able to write a well formed / valid XML document.

CO 3: Students will be able to write a server side java application called JSP to catch form data sent from client and store it on database

Suggested Readings/Books:

1. Deitel, Deitel, Nieto, and Sandhu: XML How to Program, Pearson Education.
2. Herbert Schildt: Java 2: The Complete Reference, Fifth Edition, TMH.
3. Ivan Bayross: Web Enabled Commercial Application.
4. Schafer: Development, BPB.
5. HTML, CSS, Java Script,Perl, Python and PHP, Wiley India Textbooks.
6. R. Peterson, 2007, Linux: The Complete Reference, Sixth Edition, TMH.

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

Detailed Contents:

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. Core Python Programming, Wesley J. Chun, Second Edition, Pearson.

Elective-II

Course Code: BTCS 514-18	**Course Title:** Mobile Application Development	**L:3;T:0; P:0**	**3 Credits**

Detailed Contents:

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

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

CO 1, 2

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

CO 2,3

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

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

CO 4,5

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

CO 4

Unit-VII
Deployment of apps: Versioning, signing and packaging mobile apps, distributing apps on market place.

CO 5

Course Outcomes:

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

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. T1.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,WroxPublisher,"Professional Mobile Application Development", Ist Edition, 2012.
5. Reto Meier, "Professional Android 4 Application Development", Wrox, 2012.

Course Code: BTCS 515-18	**Course Title:** Computer Graphics	**L:3; T:0; P:0**	**3 Credits**

Detailed Syllabus:

UNIT–I

Overview of Computer Graphics: Basics of Computer Graphics, Applications, Video Display devices, Raster–Scan displays, Random–Scan displays, Color CRT Monitors, Flat–Panel Displays; Video Controller, Display Processor, Common Graphic Input and Output devices, Graphic File Formats, Graphics Software's.

CO 1

Unit- II

Output Primitives: Line Drawing, DDA, Bresenham Line Algorithm; Mid-Point Line Algorithm, Bresenham Circle Algorithm, Midpoint Circle drawing algorithms; Midpoint Ellipse Algorithm; Flood and Boundary Filling.

CO 2

Unit- III

Two-Dimensional Geometric Transformation: Translation, Rotation, Scaling, Reflection, Shearing, Matrix representations; Composite transformations.

CO 2

UNIT–IV

Two-Dimensional Viewing: Viewing coordinate reference frame; Window to Viewport coordinate transformation. Point Clipping, Line Clipping, text Clipping; Cohen–Sutherland and Liang– Barskey Algorithms for line clipping; Sutherland–Hodgeman algorithm for polygon clipping.

CO 3, 4

Unit- V

Three Dimensional Transformations & Viewing: Translation, Rotation, Scaling, Reflection and composite transformations. Parallel and Perspective Projections, Viewing Transformation: View Plan, View Volumes and Clipping.

CO 4, 5

Unit- VI

3D Graphics and Visibility: Plane projections and its types, Vanishing points, Specification of a 3D view. Image and object precision, Hidden edge/surface removal or visible edge/surface determination techniques; z buffer algorithms, Depth sort algorithm, Scan line algorithm and Floating horizon technique.

CO 5

Unit –VII

Color Models: Properties of Light, Intuitive Color Concepts, concepts of chromaticity, RGB Color Model, CMY Color Model, HLS and HSV Color Models, Conversion between RGB and CMY color Models, Conversion between HSV and RGB color models, Color Selection and Applications.

CO 5, 6

UNIT–VIII

Animation: Graphics Design of Animation sequences, General Computer Animation Functions Introduction to Rendering, Raytracing, Antialiasing, Fractals, Gourard and Phong shading.

CO 6

Course Outcomes:

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

CO 1. To list the concepts used in computer graphics.

CO 2. To implement various algorithms to scan, convert the basic geometrical primitives, transformations, Area filling, clipping.

CO 3. To describe the importance of viewing and projections.

CO 4. To define the fundamentals of animation, virtual reality and its related technologies.

CO 5. To understand a typical graphics pipeline

CO 6. To design an application with the principles of virtual reality

References:

1. D. Hearn and M.P. Baker, Computer Graphics: C version, 2nd Edition, PHI, 2004.
2. D.F. Rogers, Procedural Elements for Computer Graphics, 2nd Edition, Addison Wasley, 2004.
3. D.F. Rogers, Mathematical Elements for Graphics, 2nd Edition. McGraw Hill, 2004.
4. J.D. Foley et al, Computer Graphics, Principles and Practices, 2nd Edition, Addison Wasley, 2004.
5. Roy A. Plastock, Gordon Kalley, Computer Graphics, Schaum's Outline Series, 1986.

Course Code: BTCS 516-18	**Course Title:** Internet of Things	**L:3; T:0; P:0**	**3 Credits**

Detailed Syllabus:

1. **Introduction to IoT (8 Hours)**
   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.

   CO 1

2. **Elements of IoT (9 Hours)**
   Hardware Components- Computing (Arduino, Raspberry Pi), Communication, Sensing, Actuation, I/O interfaces.
   Software Components- Programming API's (using Python/Node.js/Arduino) for Communication CO2
   Protocols-MQTT, ZigBee, Bluetooth, CoAP, UDP, TCP.

   CO 2

3. **IoT Application Development (18 Hours)**
   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.

   CO 3

4. **IoT Case Studies (10 Hours)**
   IoT case studies and mini projects based on Industrial automation, Transportation, Agriculture, Healthcare, Home Automation

   CO 4

Course Outcomes:

At the end of the course the student should be able to:
CO 1. To understand internet of Things and its hardware and software components
CO 2. To develop an Interface, I/O devices, sensors & communication modules
CO 3. To remotely monitor data and control devices
CO 4. To develop real life IoT based projects

List of Suggested Books

1. Vijay Madisetti, Arshdeep Bahga, İnternet 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: BTCS 505-18	**Course Title:** Database management System lab	**0L:0T:4P**	**2 Credits**

Detailed List of Tasks:

1. Introduction to SQL and installation of SQL Server / Oracle.
2. Data Types, Creating Tables, Retrieval of Rows using Select Statement, Conditional Retrieval of Rows, Alter and Drop Statements.
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 Statemets.
4. Set Operators, Nested Queries, Joins, Sequences.
5. Views, Indexes, Database Security and Privileges: Grant and Revoke Commands, Commit and Rollback Commands.
6. PL/SQL Architecture, Assignments and Expressions, Writing PL/SQL Code, Referencing Non- SQL parameters.
7. Stored Procedures and Exception Handling.
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	**L:0;T:0; P:2**	**1 Credits**

Detailed List of Tasks:

1. Study and usage of OpenProj or similar software to draft a project plan
2. Study and usage of OpenProj or similar software to track the progress of a project
3. Preparation of Software Requirement Specification Document, Design Documents and Testing Phase
4. related documents for some problems
5. Preparation of Software Configuration Management and Risk Management related documents
6. Study and usage of any Design phase CASE tool
7. To perform unit testing and integration testing
8. To perform various white box and black box testing techniques
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	**L:0;T:0; P:2**	**1 Credits**

Detailed List of Tasks:

Task1: To study the different types of Network cables and network topologies
Task2: Practically implement and test the cross-wired cable and straight through cable using clamping tool and network lab cable tester.
Task3: Study and familiarization with various network devices.
Task4: Familiarization with Packet Tracer Simulation tool/any other related tool.
Task5: Study and Implementation of IP Addressing Schemes
Task6: Creation of Simple Networking topologies using hubs and switches
Task7: Simulation of web traffic in Packet Tracer
Task8: Study and implementation of various router configuration commands
Task9: Creation of Networks using routers.
Task10: Configuring networks using the concept of subnetting
Task11: Practical implementation of basic network command and Network configuration commands like ping, ipconfig, netstat, tracert etc. for trouble shooting network related problems.
Task12: Configuration of networks using static and default routes.

Course Outcomes:

The students will be able to

1. Know about the various networking devices, tools and also understand the implementation of network topologies.
2. Create various networking cables and know how to test these cables.
3. Create and configure networks in packet tracer tool using various network devices and topologies.
4. Understand IP addressing and configure networks using the subnettin.
5. Configure routers using various router configuration commands.
6. Trouble shoot the networks by using various networking commands. Graphics Software's.

Elective-I Lab

Course Code: BTCS511-18	**Course Title:** Programming in Java Lab	**L:0;T:0;P:2**	**1Credits**

To accomplish CO1;

1. WAP in Java to show implementation of classes.
2. WAP in Java to show implementation of inheritance.
3. WAP in Java to show Implementation of packages and interfaces.

To accomplish CO2;

4. WAP in Java to show Implementation of threads.
5. WAP in Java Using exception handling mechanisms.
6. WAP in Java to show Implementation of Applets.

To accomplish CO3;

7. WAP in Java to show Implementation of mouse events, and keyboard events.
8. WAP in Java to show Implementing basic file reading and writing methods.
9. Using basic networking features, WAP in Java

To accomplish CO4;

10. WAP in Java to show Connecting to Database using JDBC.

Project work: A desktop based application project should be designed and implemented in java**.**

Course Outcomes:

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

CO1. Implement the features of Java such as opeartors, classes, objects, inheritance, packages and exception handling

CO2. Design problems using latest features of Java like garbage collection, Console class, Network interface, APIs

CO3. Develop competence in Java through the use of multithreading, Applets etc

CO4. Apply advance concepts like socket and database connectivity, and develop project based on industry orientation.

Suggested Readings/Books

1. Herbert Schildt, The Complete Reference Java2, McGraw-Hill.
2. Deitel and Deitel, Java: How to Program, 6th Edition, Pearson Education.
3. James Edward Keogh, Jim Keogh, J2EE: The complete Reference, Mc Graw Hill

---

Course Code: BTCS 512-18	**Course Title:** Web and Open Source Technologies Laboratory	**L:0;T:0; P:2**	**1Credits**

Detailed List of Tasks:

1. Write an HTML page including javascript that takes a given set of integer numbers and shows them after sorting in descending order.
2. Write an HTML page that has one input, which can take multi-line text and a submit button. Once the user clicks the submit button, it should show the number of characters, words and lines in the text entered using an alert message. Words are separated with white space and lines are separated with new line character.
3. Write an HTML page that contains a selection box with a list of 5 countries. When the user selects a country, its capital should be printed next to the list. Add CSS to customize the properties of the font of the capital (color, bold and font size).
4. Create an XML document that contains 10 user's information.
5. Using jQuery find all children in a specified class of a division
6. Find all elements of a form that are disabled
7. Create an input form and validate using jQuery. Highlight inputs elements if errors occur
8. Build a Single Page Application (SPA) using AngularJS.

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

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.

Elective-II Lab

Course Code: BTCS 517- 18	**Course Title:** Mobile Application Development Lab	**L:0;T:0; P:2**	**1Credits**

Detailed List of Tasks:

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: BTCS518-18	**Course Title:** Computer Graphics Lab	**L:0;T:0; P:2**	**1Credits**

Detailed List of Tasks:

1. WAP to draw different geometric structures using different functions.
2. Implement DDA line generating algorithm.
3. Implement Bresenham's line generating algorithm.
4. Implement Mid-point circle line generating algorithm.
5. Implementation of Bresenham's circle drawing algorithm.
6. Implementation of mid-point circle generating Algorithm.
7. Implementation of ellipse generating Algorithm.
8. WAP of color filling the polygon using Boundary fill and Flood fill algorithm.
9. To translate an object with translation parameters in X and Y directions.
10. To scale an object with scaling factors along X and Y directions.
11. Program of line clipping using Cohen-Sutherland algorithm.
12. To perform composite transformations of an object.
13. To perform the reflection of an object about major

Course Code: BTCS 519-18	**Course Title:** Internet of Things Laboratory Lab	**L:0;T:0; P:2**	**1Credits**

Detailed List of Tasks:

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 things peak cloud.
10. Write a program on Arduino/Raspberry Pi to retrieve temperature and humidity data from thing speak 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.
