# Study Scheme & Syllabus
# of

## Bachelor of Technology
## Software Engineering

## B. Tech (SE)

# Batch 2019 onwards

**By**

**Board of Study CSE on 16 July 2019**

## IK Gujral Punjab Technical University
**(Under MOU with Thompson Rivers University)**

## Bachelor of Technology in Software 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 | | |
| **SE 1110** | Engineering Science Course | Programming for Engineers 1 | 3 | 0 | 2 | 60+30 | 40+20 | 150 | 3 |
| **EP 1150** | Basic Science Course | Physics for Engineers 1 | 3 | 0 | 2 | 60+30 | 40+20 | 150 | 3 |
| **EP 1700** | Engineering Science Course | Engineering Mechanics 1 | 3 | 1 | 0 | 60 | 40 | 100 | 3 |
| **EP 1990** | Engineering Science Course | Introduction to Engineering Measurements | 3 | 1 | 0 | 60 | 40 | 100 | 2 |
| **MA 1130** | Basic Science Course | Enriched Calculus 1 | 3 | 2 | 0 | 60 | 40 | 100 | 3 |
| **MA 1300** | Basic Science Course | Linear Algebra for Engineers | 3 | 2 | 0 | 60 | 40 | 100 | 3 |
| **Total** | | | **18** | **6** | **4** | **420** | **280** | **700** | **17** |

## Second Semester

| Course Code | Course Type | Course Title | Load Allocations | | | Marks Distribution | | Total Marks | Credits |
|---|---|---|---|---|---|---|---|---|---|
| | | | L | T | P | Internal | External | | |
| **SE 1210** | Engineering Science Course | Programming for Engineers 2 | 3 | 0 | 2 | 60+30 | 40+20 | 150 | 3 |
| **EP 1250** | Basic Science Course | Physics for Engineers 2 | 3 | 0 | 3 | 60+30 | 40+20 | 150 | 3 |
| **MA 1230** | Basic Science Course | Enriched Calculus 2 | 3 | 2 | 0 | 60 | 40 | 100 | 3 |
| **MA 1700** | Engineering Science Course | Discrete Mathematics | 3 | 1 | 0 | 60 | 40 | 100 | 3 |
| **CH 1520** | Basic Science Course | Principles of Chemistry | 3 | 0 | 3 | 60 | 40 | 100 | 3 |
| **DR 1520** | Engineering Science Course | Engineering Graphics | 2 | 0 | 3 | 60 | 40 | 100 | 3 |
| **Total** | | | **17** | **3** | **11** | **420** | **280** | **700** | **18** |

## 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 |
| **SE301-19** | Professional Core Courses | Data structure & Algorithms | 3 | 1 | 0 | 40 | 60 | 100 | 3 |
| **SE302-19** | Professional Core Courses | Computer Networks | 3 | 1 | 0 | 40 | 60 | 100 | 3 |
| **SE303-19** | Professional Core Courses | Software Engineering | 3 | 0 | 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 |
| **SE304-19** | Professional Core Courses | Data structure & Algorithms Lab | 0 | 0 | 4 | 30 | 20 | 50 | 2 |
| **SE305-19** | Professional Core Courses | Computer Networks lab. | 0 | 0 | 2 | 30 | 20 | 50 | 1 |
| **SE306-19** | Professional Core Courses | Software Engineering lab. | 0 | 0 | 2 | 30 | 20 | 50 | 1 |
| **SE307-19** | Professional Core Courses | IT Workshop* | 0 | 0 | 2 | 30 | 20 | 50 | 1 |
| | | Summer Institutional Training | 0 | 0 | 0 | 60 | 40 | 100 | Satisfactory/ Unsatisfactory |
| | | **Total** | **14** | **3** | **12** | **410** | **440** | **850** | **21** |

## Fourth Semester

| Course Code | Course Type | Course Title | Load Allocations | | | Marks Distribution | | Total Marks | Credits |
|---|---|---|---|---|---|---|---|---|---|
| | | | L | T | P | Internal | External | | |
| **SE401-19** | Professional Core Courses | Database Management System | 3 | 0 | 0 | 40 | 60 | 100 | 3 |
| **SE402-19** | Engineering Science Course | Computer Organization & Architecture | 3 | 1 | 0 | 40 | 60 | 100 | 3 |
| **SE403-19** | Professional Core Courses | Operating Systems | 3 | 1 | 0 | 40 | 60 | 100 | 3 |
| **SE404-19** | 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 2 | 2 | 1 | 0 | 40 | 60 | 100 | 3 |
| **SE405-19** | Engineering Science Course | Database Management System Lab | 0 | 0 | 4 | 30 | 20 | 50 | 2 |
| **SE406-19** | Engineering Science Course | Computer Organization & Architecture Lab | 0 | 0 | 2 | 30 | 20 | 50 | 1 |
| **SE407-19** | Professional Core Courses | Operating Systems Lab | 0 | 0 | 2 | 30 | 20 | 50 | 1 |
| **SE408-19** | Professional Core Courses | Design & Analysis of Algorithms Lab | 0 | 0 | 4 | 30 | 20 | 50 | 2 |
| **Total** | | | **14** | **4** | **12** | **320** | **380** | **700** | **21** |

# *Semester-1st*

| SE 1110 | Programming for Engineers 1 | 3L:0T:2P | 3 Credits |
|---|---|---|---|

**Calendar Description:** Students are introduced to the concepts of computer programming with specific emphasis on engineering problems and applications. Students learn computer programming as a part of engineering process. Students conceptualize the programming approach in line with engineering profession by following design, implement and testing using specifications. Students explore C++ programming basics, statements, syntax, control structures, functions, and types of arrays.

**Prerequisites:** Admission to either Electrical, Computer or Software Engineering Program or Engineering Program Advisor's permission.

**Course Vectoring:** (3, 0, 2)

**Educational Objectives/Outcomes**:

After completing this course, the students will have the ability to:

1. Understand the design, coding and debugging of computer program.
2. Translate the specification into a computer program to solve small engineering problems.
3. Describe the fundamental programming constructs and use them to develop program.
4. Apply the fundamental principles, concepts and constructs of modern computer programming to solve small engineering problems.
5. Demonstrate the control structures, selections and execution flow.
6. Write functions to demonstrate the knowledge of modular programming.

**Texts/Materials**

B1: Y. Daniel Liang, Introduction to Programming with C++, 3/E, Prentice Hall, 2012, ISBN-10: 0133252817.

| Course Topics | Textbook Mapping | |
|---|---|---|
| | Chapter # | Book |
| Introduction to Computers, Programming, and C++ | 1 | B1 |
| Elementary Programming | 2 | B1 |
| Selections | 3 | B1 |
| Mathematical Functions, Characters, and Strings | 4 | B1 |
| Loops | 5 | B1 |
| Functions | 6 | B1 |
| Single-Dimensional Arrays and C-Strings | 7 | B1 |
| Multidimensional Arrays | 8 | B1 |

| EP 1150 | Physics for Engineers 1 | 3L:0T:2P | 3 Credits |
|---|---|---|---|

**Calendar Description:** This course is similar to PHYS 1150: Mechanics and Waves except that Engineering students do not complete the laboratory portion. Students are introduced to and apply calculus to physical concepts. Topics include mechanics, simple harmonic motion, mechanical waves, sound, wave optics and geometric optics.

**Prerequisites:** Admission to the Engineering Transfer Program.

**Corse Vectoring:** (3, 0, 2)

**Educational Objectives/Outcomes:**

After completing this course the students will have the ability to:

1. Explain and write about the physical concepts for waves, wave optics, geometric optics, kinematics, and dynamics
2. Apply the physical concepts to complex multiple unknown problems by solving such problems using calculus

**Texts/Materials**

B1: Serway, R.; Jewett, J.; Physics for Scientists and Engineers, Brooks Cole; 9 edition, 2013, ISBN-13: 978-1133947271.

| Course Topics | Textbook Mapping | |
|---|---|---|
| | Chapter | Book |
| Optics, Electromagnetic Waves, Plane EM Waves, EM Spectrum | 34 | B1 |
| Geometric Optics, The Ray Approximation, Reflection and Refraction, Dispersion and Prisms, Total Internal Reflection, Optics - Spherical Mirrors, Refraction, Lenses | 35, 36 | B1 |
| Wave Optics, Conditions for Interference, Young's Double-slit Experiment, Intensity Distribution, Interference in Thin Films | 37 | B1 |
| Kinematics and Vectors, Review of Motion in One-Dimension, Displacement, Velocity, Speed, Acceleration, Kinematic Equations, Freely Falling Bodies | 2 | B1 |
| Vectors Review, Properties of Vectors, Components and Unit Vectors | 3 | B1 |
| Mechanics (with Calculus), Two-dimensional Motion, Displacement, Velocity and Position Vectors, 2-D Motion with Constant Acceleration, Uniform Circular Motion | 4 | B1 |
| Force and Motion, Concept of Force, Newton's Laws, Friction | 5 | B1 |
| Work and Energy, The Scalar (Dot) Product, Work Done by Constant & Varying Forces, Kinetic Energy and the Work- Energy Theorem, Potential Energy, Conservative and Non- Conservative Forces, Conservation of Energy, Change in Energy when the Force is Non-conservative, Power | 7, 8 | B1 |
| Momentum and Collisions (covered in seminar) | 9 | B1 |
| Static Equilibrium and Elasticity (covered in seminar) | 12 | B1 |
| Wave Phenomena, Oscillatory Motion, Simple Harmonic Motion, Mass Attached to a Spring, Energy of the Simple Harmonic Oscillator | 15 | B1 |

| | | |
|---|---|---|
| Wave Motion, Introduction, Types of Waves, One-D Travelling Waves, Speed of Waves on Strings, Reflection and Transmission, Sinusoidal Waves, Energy Transmitted by Waves on Strings | 16 | B1 |
| Sound Waves, Speed of Sound Waves, Periodic Sound Waves, Intensity of Periodic Sound Waves, The Doppler Effect | 17 | B1 |
| Superposition and Standing Waves, Superposition and Interference of Sinusoidal Waves, Standing Waves in Strings and Air Columns | 18 | B1 |

| EP 1700 | Engineering Mechanics 1 | 3L:1T:0P | 3 Credits |
|---|---|---|---|

**Calendar Description:** Study and apply the principles required to solve engineering mechanics problems. Students explore statics, including equilibrium of particles, equilibrium of rigid bodies, trusses and dry frictions. Further topics of study in dynamics include rectilinear motion, curvilinear motion, Newton's second law, equations of motion in different coordinates, impulse, momentum, impact, work and energy.

**Prerequisites:** EPHY 1150 or PHYS 1150, MATH 1130 or MATH 1140 or MATH 1300 or MATH 1540

**Corse Vectoring:** (3, 1, 0)

**Educational Objectives/Outcomes:**

After completing this course the students will have the ability to:

1. Solve statical equilibrium problems for particles subjected to concurrent forces.
2. Solve statical equilibrium problems for rigid bodies subjected to a system of forces and moments.
3. Solve problems involving dry friction.
4. Solve kinematic problems for particles and systems of particles in motion.
5. Use Newton's 3rd Law to solve kinetics problems for particles using rectangular, normal- tangential and polar/cylindrical coordinate systems.
6. Solve kinetics problems for particles and systems of particles using work and energy.
7. Solve kinetics problems for particles and systems of particles using impulse and momentum.

**Texts/Materials**

B1: Hibbeler, R.C.; Engineering Mechanics, Statics and Mechanics, Prentice Hall, 2015, ISBN-13: 978-0133915426.
M1: Engineering Design Notepaper (Available at the Bookstore), Scientific Calculator.

| Course Topics | Textbook Mapping | |
|---|---|---|
| | Chapter | Book |
| Introduction, Fundamental Concepts, Units of Measurement, The International System of Units, Numerical Calculations, General Procedure for Analysis | 1.1 to 1.6 | B1 |
| Scalars and Vectors, Vector Operations, Vector Addition of Forces, Addition of a System of Coplanar Forces, Cartesian Vectors | 2.1 to 2.5 | B1 |
| Addition of Cartesian Vectors, Position Vectors, Force Vector Directed Along a Line, Dot Product | 2.6 to 2.9 | B1 |
| Condition for the Equilibrium of a Particle, The Free-Body Diagram, Coplanar Force Systems | 3.1 to 3.3 | B1 |
| Three-Dimensional Force Systems | 3.4 | B1 |
| Moment of a Force – Scalar Formulation, Cross Product Moment of a Force – Vector Formulation, Principle of Moments, Moment of a Force About a Specified Axis | 4.1 to 4.5 | B1 |

| | | |
|---|---|---|
| Moment of a Couple, Simplification of a Force and Couple System, Further Simplification of a Force and Couple | 4.6 to 4.8 | B1 |
| Reduction of a Simple Distributed Loading | 4.9 | B1 |
| Conditions for Rigid-Body Equilibrium | 5.1 | B1 |
| Free-Body Diagrams, Equations of Equilibrium | 5.2 to 5.3 | B1 |
| Two- and Three-Force Members, | 5.4 | B1 |
| Free-Body Diagrams, Equations of Equilibrium. Constraints and Statical Determinacy | to 5.7 | |
| Simple Trusses, The Method of Joints | 6.1 to 6.2 | B1 |
| Zero-Force Members, The Method of Sections | 6.3 to 6.4 | B1 |
| Characteristics of Dry Friction, Problems Involving Dry Friction, Wedges | 8.1 to 8.3 | B1 |
| Introduction, Rectilinear Kinematics: Continuous Motion Rectilinear Kinematics: Erratic Motion, General Curvilinear Motion, Curvilinear Motion: Rectangular Components, Motion of a Projectile | 12.1 to 12.6 | B1 |
| Curvilinear Motion: Normal and Tangential Components Curvilinear Motion: Cylindrical Components, Absolute Dependent Motion Analysis of Two Particles, Relative- Motion of Two Particles Using Translating Axes | 12.7 to 12.10 | B1 |
| Newton's Second Law of Motion, The Equation of Motion The Equation of Motion for a System of Particles, Equations of Motion: Rectangular Coordinates, Equations of Motion: Normal and Tangential Coordinates | 13.1 to 13.5 | B1 |
| Equations of Motion: Cylindrical Coordinates | 13.6 | B1 |
| The Work of A Force, Principle of Work and Energy Principle of Work and Energy for a System of Particles, Power and Efficiency | 14.1 to 14.4 | B1 |
| Conservative Forces and Potential Energy, Conservation of Energy | 14.5 to 14.6 | B1 |
| Principle of Linear Impulse and Momentum, Principle of Linear Impulse and Momentum for a System of Particles, Conservation of Linear Momentum for a System of Particles | 15.1 to 15.2 | B1 |
| Impact, Angular Momentum, Relation Between Moment of a Force and Angular Momentum | 15.3 to 15.7 | B1 |

| EP 1990 | Introduction to Engineering Measurements | 3L:1T:0P | 2 Credits |
|---|---|---|---|

**Calendar Description:** Students are introduced to the measurement and control of physical quantities of interest in engineering and scientific applications. Issues and methods relevant to the real-time measurement and control of parameters such as force, displacement, acceleration, temperature, level, pressure, and flow are considered. Students apply the principles developed in the course during seminars, and in discussions of case studies that are relevant to various engineering or scientific disciplines.

**Prerequisites:** MATH 1130 or 1140, and EPHY 1150 or PHYS 1150, or permission of the instructor.

**Corse Vectoring: (**3, 1, 0)

**Educational Objectives/Outcomes:**

After completing this course the students will have the ability to:

1. Understand the operating principles of various sensors and instruments used to measure and control physical quantities in engineering applications.
2. Understand the linearity, gain, range, quantization error, and speed of measurement systems.
3. Understand the principles of real-time data acquisition, and measurement systems software development.
4. Use data-acquisition hardware and software to interface measurement sensors, and create software to capture, analyze, and display, and process measured quantities.
5. Assemble and apply elementary sensor signal-conditioning circuits.
6. Use electronic test equipment to test and configure electronic measurement circuits.
7. Apply curve fitting methods for the application of non-linear sensors in measurement.
8. Familiar with case studies and engineering applications involving the measurement of temperature, strain, force, pressure, angular and linear position and speed, and flow.

**Texts/Materials**

Reference materials will be provided via the course website.

| Course Topics |
|---|
| Electrical Measurement fundamentals, Current, voltage, power, resistance, capacitance, inductance, Multimeters, function generators, oscilloscopes, Ohm's and Kirchhoff's laws, Electro-mechanical energy conversion principles, Rotating DC machine principles and measurements |
| Data Acquisition Principles, Introduction to data acquisition hardware, Sensors and signal conditioning, Linearity, gain, span, sampling rate, and quantization errors, Non- linear sensors, Introduction to data acquisition software applications. |
| Sensing Technologies and their Applications, Temperature, Strain, Force, Torque, and Pressure, Flow and Level, Displacement, Position, and Proximity |
| Principles of Surveying, Coordinate Systems; Electronic surveying transits; Electronic distance measurements; Leveling and positioning a transit; Measuring horizontal angles; Measuring vertical angles; Triangulation with controlled points; |

| MA 1130 | Enriched Calculus 1 | 3L:2T:0P | 3 Credits |
|---|---|---|---|

**Course Description:** This course includes an in-depth study of single-variable differential calculus and its applications, and provides a strong foundation for further study in mathematical disciplines. This is a required course for students in the Engineering Transfer program.

**Prerequisites:** "A" in Pre-Calculus Math 12 or equivalent (British Columbia graduates of 2013 onwards) or "A" in Principles of Math 12 or equivalent (British Columbia graduates prior to 2013) or admission to the Engineering Program Note: Students who already have credit for MATH 1140, MATH 1150, or MATH 1170 may not take MATH 1130 for further credit.

**Course Vectoring:** (3, 2, 0)

**Educational Objectives/Outcomes:**

After completing this course, the students will have the ability to:

1. Understand the mathematical concept of a limit at a basic level.
2. Understand the derivative as a limit describing instantaneous rates of change.
3. Display reasonable proficiency in working with elementary mathematical functions and their derivatives.
4. Interpret and apply derivatives in applications.
5. Use the computer mathematics system MAPLE when appropriate.

**Texts/Materials**

B1: W. Briggs, L. Cochran, B. Gillett, Calculus: Early Transcendentals, Single Variable, 2nd Edition, Pearson Education, 2015.

| Course Topics | Textbook Mapping | |
|---|---|---|
| | Chapter | Book |
| Functions, domains, ranges, and intervals, graphs, combinations of functions models | 1 | B1 |
| Tangent lines and slope predictors, limits, techniques for computing limits, infinite limits, limits at infinity, continuity | 2 | B1 |
| The derivative and rates of change, basic differentiation rules, power rule, product and quotient rules, derivative of trigonometric functions, derivatives as rates of change, the chain rule, implicit differentiation, derivatives of exponential and logarithmic functions, derivatives of inverse trig functions, related rates, maxima and minima on closed intervals, applied optimization problems | 3 | B1 |
| Maxima and minima, increasing/decreasing functions and the mean value theorem, first derivative test, curve sketching, second derivative test and concavity, graphing functions, applied optimization problems, linear approximation, L'Hopital's rule, antiderivatives | 4 | B1 |

| MA 1300 | Linear Algebra for Engineers | 3L:2T:0P | 3 Credits |
|---|---|---|---|

**Course Description:** This course is designed for students in the first year Engineering Transfer program and covers vectors in n- dimensional space, linear transformations, matrices, Jordan-Gauss elimination method, eigenvalues and eigenvectors and their application to engineering problems. A computer lab component will be used to explore applications.

**Prerequisites:** Admission to the Engineering program.

**Course Vectoring:** (3, 2, 0)

**Educational Objectives/Outcomes:**

After completing this course, the students will have the ability to:

1. Understand the properties of the vector spaces,
2. Understand the relationship between a linear transformation and the associated matrix,
3. Find eigenvalues and eigenvectors and apply these concepts to solving differential equations.

Texts/Materials

B1: Lay, D., Lay, S., McDonald, J., Linear Algebra and its Applications, Pearson, Fifth Edition (2016).

| Course Topics |
|---|
| Linear Equations in Linear Algebra: Systems of Linear Equations, Row Reduction and Echelon Forms, Vector Equations, The Matrix Equation $AA\mathbf{xx}=\mathbf{bb}$, Solution Sets of Linear Systems, Applications of Linear Systems, Linear Independence, Introduction to Linear Transformation, The Matrix of a Linear Transformation |
| Matrix Algebra: Matrix Operations, The Inverse of a Matrix, Characterization of Invertible Matrices, Subspaces of $\mathbb{R}nn$, Dimension and Rank |
| Determinants: Properties of Determinants, Cramer's Rule, Volume and Linear Transformations |
| Eigenvalues and Eigenvectors: Eigenvalues and Eigenvectors, The Characteristic Equation, Similarity, Diagonalization, complex numbers, Discrete Dynamical Systems, Applications Differential Equations |
| Orthogonality: Orthogonality, Orthogonal Complement, Orthogonal Projection, The Gram-Schmidt Process |

# *Semester-2nd*

| SE 1210 | Programming for Engineers-2 | 3L:0T:2P | 3 Credits |
|---|---|---|---|

**Calendar Description:** Students are introduced to the concepts of object-oriented programming in designing, implementing and testing engineering problems. Students learn the principles of inheritance and polymorphism in designing of methods and classes in object-oriented approach. Students explore the techniques of reading and writing data to file, exceptional handling, pointers, and dynamic memory management, vectors, stacks and recursion.

**Prerequisites:** A minimum of grade "C" or better in SENG 1110.

**Course Vectoring:** (3, 0, 2)

**Educational Objectives/Outcomes:**

After completing this course, the students will have the ability to:

1. Identify classes, objects, members of a class and the relationships among to model a problem specification using C++.
2. Describe the principles of inheritance and polymorphism and demonstrate their relation to the design of methods and classes in C++.
3. Use pointers and describe the rationale in small engineering applications.
4. Demonstrate the use of reading and writing data to file.
5. Discuss some conceptual use of vectors, stacks and recursion.
6. Understand error-handling techniques using exception-handling techniques.

**Texts/Materials**

B1: Y. Daniel Liang, Introduction to Programming with C++, 3/E, Prentice Hall, 2012, ISBN-10: 0133252817.

| Course Topics | Textbook Mapping | |
|---|---|---|
| | Chapter # | Book |
| Objects and Classes | 9 | B1 |
| Object-Oriented Thinking | 10 | B1 |
| Pointers and Dynamic Memory Management | 11 | B1 |
| Templates, Vectors, and Stacks | 12 | B1 |
| File Input and Output | 13 | B1 |
| Operator Overloading | 14 | B1 |
| Inheritance and Polymorphism | 15 | B1 |
| Exception Handling | 16 | B1 |
| Recursion | 17 | B1 |

| EP 1250 | Physics for Engineers 2 | 3L:0T:3P | 3 Credits |
|---|---|---|---|

**Calendar Description:** This course is a continuation of EPHY 1150. Topics include thermodynamics, kinetic theory of gases, electricity and magnetism.

**Prerequisites:** A minimum of grade "C" in EPHY 1150.

**Corse Vectoring:** (3, 0, 3)

**Educational Objectives/Outcomes:**

After completing this course the students will have the ability to:

1. Solve problems in electrostatics involving electric forces and electric fields due to point charges by means of vector additions and Coulomb's law.
2. Calculate electric potentials and potential differences for point charges and uniformly charged conductors.
3. Solve kinematic problems involving point charges moving within a uniform electric/magnetic field.
4. Calculate magnetic forces and torques due to current-carrying conductors inside a uniform magnetic field.
5. Determine magnetic fields around current-carrying conductors by means of vector additions and Biot-Savart law.
6. Use Ampere's Law to determine the magnetic fields around current-carrying conductors and inside an ideal solenoid.
7. Explain the concepts of magnetic flux and eddy current, and apply Faraday's and Lenz's Laws to solve problems in electromagnetic induction and the concepts of thermal equilibrium and absolute temperature, and solve problems involving the thermal properties of solids, liquids and ideal gases.
8. Apply the Laws of Thermodynamics and explain them.

**Texts/Materials**

B1: R.A. Serway and J.W. Jewett, Physics for Scientists and Engineers, Thomson Learning Inc., 8th ed. (2010) or 9th ed. (2014)

| Course Topics | Textbook Mapping | |
|---|---|---|
| | Chapter | Book |
| Electric Fields: Properties of Charge; Insulators and Conductors; Coulomb's Law; Electric Fields; Electric Field Lines; Motion of a Charged Particle in a Uniform Electric Field. | 23 | B1 |
| Gauss's Law: Electric Flux; Gauss's Law and Applications. | 24 | B1 |
| Electric Potential: Potential Difference and Electric Potential; Potential Differences in Uniform Electric Fields; Electric Potential and Potential Energy Due to Point Charges; Obtaining Electric Fields from Electric Potentials; Potential of a Charged Conductor. | 25 | B1 |
| Magnetic Fields: The Magnetic Field and Vector Products; Motion of a Charged Particle in a Uniform Magnetic Field; Magnetic Force on a Current-Carrying Conductor; Torque on a Current Loop in a Uniform Magnetic Field. | 29 | B1 |

| | | |
|---|---|---|
| Sources of the Magnetic Field: The Biot-Savart Law; Ampere's Law; Magnetic Field of a Solenoid; Magnetic Flux. | 30 | B1 |
| Faraday's Law: Faraday's Law of Induction; Motional EMF; Lenz's Law. | 31 | B1 |
| Temperature: Law of Thermodynamics and Temperature; Thermometers and Temperature Scales; The Constant-Volume Gas Thermometer and the Absolute Temperature Scale; Thermal Expansion of Solids and Liquids; The Ideal Gas Law. | 19 | B1 |
| The 1st Law of Thermodynamics: Heat and Internal Energy; Heat Capacity; Latent Heat; Work and Heat; The 1st Law and Energy Conservation; Energy Transfer Mechanisms. | 20 | B1 |
| The Kinetic Theory of Gases: Molecular Model of Ideal Gas; Specific Heat of Ideal Gas; Adiabatic Processes for Ideal Gas; The Equipartition of Energy. | 21 | B1 |
| The 2nd Law of Thermodynamics: Heat Engines; Reversible and Irreversible Processes; Entropy and the 2nd Law; Entropy Changes in Irreversible Processes. | 22 | B1 |

| MA 1230 | Enriched Calculus 2 | 3L:2T:0P | 3 Credits |
|---|---|---|---|

**Course Description:** This course offers an in-depth study of single-variable integral calculus and its applications to provide a strong foundation for further study in mathematical disciplines. It is a required course for students in the engineering transfer program.

**Prerequisites:** A minimum of grade "C" or better in MATH 1130.

**Course Vectoring:** (3, 2, 0)

**Educational Objectives/Outcomes:**

After completing this course, the students will have the ability to:

1. Compute integrals of a number of elementary functions by hand
2. Compute integrals to any desired accuracy by numerical methods;
3. Have an understanding of the fundamental nature of integration;
4. Be able to solve problems involving areas, volumes, lengths of curves, applied problems as centres of mass, work, fluid pressure and moments of inertia. etc., in which integration is used.

**Texts/Materials**

B1: W. Briggs, L. Cochran, B. Gillett, Calculus: Early Transcendentals, Single Variable, 2nd Edition, Pearson Education, 2015.

| Course Topics | Textbook Mapping | |
|---|---|---|
| | Chapter | Book |
| Antiderivatives and Initial Value Problems, Velocity and Displacement, Sigma Notation and Reimann Sums, Evaluation of Integrals, Fundamental Theorem of Calculus and Area, Average value of a Function, Integration by Substitution, Area Between Curves, Volumes, Arc Length and Surface Area, force and Work, Moments, centroids, Centres of Mass, Logarithms and Exponential Models | 5 & 6 | B1 |
| Basic Approaches, Integration by Parts, Trigonometric Integrals including Inverse Trig, Trig Substitution, Rational Functions and Partial Fractions, Other Integration Strategies, Numerical integration, Improper Integrals | 7 | B1 |
| Basic Idea of Differential Equations, Direction Fields, Separable Differential Equations, First-Order Linear, Differential Equations, Modeling with Differential Equations | 8 | B1 |
| Sequences and Series | 9 & 10 | B1 |
| Polar Coordinates | 11 | B1 |

| MA 1700 | Discrete Mathematics | 3L:1T:0P | 3 Credits |
|---|---|---|---|

**Course Description:** This course is an introduction to the foundation of modern mathematics including basic set theory; counting; solution to recurrence relations; logic and quantifiers; properties of integers; mathematical induction; asymptotic notation; introduction to graphs and trees; finite state machines and formal languages; Boolean algebra.

**Prerequisites:** A minimum grade of C+ in Principles of Math 12 or MATH 100 or a minimum grade of C+ in MATH 061 within the last two years at least or permission of instructor.

**Course Vectoring:** (3, 1, 0)

**Educational Objectives/Outcomes:**

After completing this course, the students will have the ability to:
Introduce students to discrete mathematics including formal proof.
Express logical statements formally.
Construct a variety of mathematical proofs.
Evaluate functions and relations.
Use graphs to model and solve problems.
Analyze algorithms using appropriate notation to determine their efficiency.

**Texts/Materials**

B1: Epp, Susanna S., Discrete Mathematics with Applications, 4th ed, 2004.

| Course Topics | Textbook Mapping | |
|---|---|---|
| | Chapter | Book |
| The Logic of Compound Statements | 2 | B1 |
| The Logic of Quantified Statements | 3 | B1 |
| Elementary Number Theory and Methods of Proof | 4 | B1 |
| Mathematical Induction and Recursion | 5 | B1 |
| Set Theory | 6 | B1 |
| Functions | 7 | B1 |
| Relations | 8 | B1 |
| Counting and Probability | 9 | B1 |
| Graphs and Trees | 10 | B1 |
| Analysis of Algorithm Efficiency (if time permits) | 11 | B1 |
| Regular Expressions and Finite-State Automata | 12 | B1 |

| CH 1520 | Principles of Chemistry | 3L:0T:3P | 3 Credits |
|---|---|---|---|

**Course Description:** The topics include gas laws, thermochemistry, equilibrium, entropy and free energy, redox reactions, and electrochemistry. Students are expected to become familiar with all these topics during the course. The laboratory stresses basic precision techniques in quantitative analytical chemistry as well as the use of analytical instrumentation and experiments in physical chemistry.

**Prerequisites:** Chemistry 12 or CHEM 0600 (a grade of B or better is recommended) and CHEM 1500 (C-minimum) or acceptance into the TRU Engineering program.

**Course Vectoring:** (3, 0, 3)

**Educational Objectives/Outcomes:**

After completing this course, the students will have the ability to:

1. Understand, as well as be able to explain and apply the fundamental qualitative and quantitative properties of gases, including how gases are described by the Ideal gas equation, the origins of the ideal gas equation, the concepts of mole fraction, gaseous diffusion, non-ideal behavior of gasses and the assumptions used in kinetic molecular theory.
2. Understand, as well as be able to explain and apply the fundamental qualitative and quantitative aspects of thermodynamics including the first law of thermodynamics, various thermodynamic systems, calculations involving specific heat, heat transfer in calorimetry experiments, Hess's law and heats of solution.
3. Understand, as well as be able to explain and apply the fundamental qualitative and quantitative aspects of acid-base equilibria including dynamic equilibria, the equilibrium constant, various definitions of acids and bases, the meaning of Kw, calculations involving weak acids and bases, the qualitative and quantitative properties of strong acids, as well as those of polyprotic acids.
4. Understand, as well as be able to explain and apply the fundamental qualitative and quantitative aspects of buffers, titrations and solubility equilibria.
5. Understand, as well as be able to explain and apply the fundamental qualitative and quantitative aspects of entropy, free energy and electrochemistry as related to chemical reactions.
6. Understand, as well as be able to explain and apply the fundamental qualitative and quantitative aspects of kinetics as related to chemical reactions.

**Texts/Materials**

B1: R. Chang and K. Goldsby, Principles of Chemistry, McGraw-Hill, 2013.

| Course Topics | Textbook Mapping | |
|---|---|---|
| | Chapter | Book |
| Section 1. Gases (6 lectures) | 5 | B1 |
| Gas properties and pressure, The gas laws, The ideal gas law: Applications, Dalton's law of partial pressures, The kinetic molecular theory of gases, Diffusion and effusion, Real gases | | |
| Section 2: Thermochemistry (5 lectures) | 6 | B1 |

| | | |
|---|---|---|
| Basic concepts, Enthalpy of chemical reactions, Calorimetry, Hess' law: Applications, Examples of enthalpy changes that refer to specific processes | | |
| Section 3: Chemical equilibrium (2 lectures; Review) | 14 | B1 |
| Principles of chemical equilibrium, Relationships involving equilibrium constants, Le Chatelier's Principle | | |
| Section 4: Acid-Base equilibrium (5 lectures; Review) | 15 | B1 |
| Acid-Base concepts, Acidity of a solution, Acid-base strength and equilibrium, Problems involving weak-acid and weak-base equilibria, Acid- base properties of salt solutions (Hydrolysis), Lewis theory of acids and bases | | |
| Section 5: Buffers, Titrations & Solubility Equilibria (5 lectures) | 16 | B1 |
| Buffer solutions: pH calculations, Acid-base titrations, Solubility equilibria | | |
| Section 6: Entropy, Free Energy & Electrochemistry (8 lectures) | 17, 18 | B1 |
| Spontaneous and non-spontaneous processes, Entropy and entropy changes, The Gibbs free energy, Thermodynamics of redox reactions | | |

| DR 1520 | Engineering Graphics | 2L:0T:3P | 3 Credits |
|---|---|---|---|

**Course Description:** This course offers an introduction to learning and applying the principles of engineering graphical visualization and representation. The topics of this course include orthographic projection, technical sketching, development of the ability to visualize in three dimensions, engineering graphic standards and conventions, graphical presentation of engineering data, computer based graphics aids, engineering geometry and the solution of space problems.

**Prerequisites:** Admission to Engineering Program or written consent of Program Coordinator.

**Course Vectoring:** (2, 0, 3)

**Educational Objectives/Outcomes:**
After completing this course, the students will have the ability to:

1. Create orthographic, isometric and oblique sketches of mechanical parts;
2. Create fully dimensioned orthographic computer-aided two-dimensional drawings of mechanical parts;
3. Create sectional drawings of mechanical parts;
4. Create auxiliary views of mechanical parts;
5. Create three dimensional solid models of mechanical parts and extract two- dimensional orthographic views;
6. Solve engineering graphics problems using both descriptive geometry and the AutoCAD geometric calculator;
7. Form the concept of technical drawings for further engineering study and practice.

**Texts/Materials**
B1: Giesecke, et al., Technical Drawing with Engineering Graphics, 15th Edition (ISBN 0134306414)

| Course Topics |
|---|
| Fundamental AutoCAD Commands, getting started with AutoCAD, system and dimensioning variables, lines, circles, arcs, polylines, and text entity creation, grid, snap and other drawing aids, object snap modes, editing commands, display commands, inquiry commands, blocks, plotting |
| Sketching and Text, alphabet of lines and line precedence sketching techniques, pictorial views, isometric sketches oblique sketches, perspective sketches, lettering |
| Orthographic Views, principle orthographic views, 3rd angle projection, missing views - visualization |
| Sectional Views, extracting section views - types of sections hatch patterns, conventional breaks |
| Auxiliary Views, extracting auxiliary views, primary auxiliary views, secondary auxiliary views |
| Dimensioning and Tolerances, dimensioning practice, location dimensions, size dimensions, finish symbols, dimensioning system variables, dimension styles, tolerances |
| Threads and Fasteners, thread terminology, thread call-ups bolts and nuts, other types of fasteners |
| Fundamentals of 3D Drawing, paper space and model space UCS and WCS, display controls |
| Solid Modeling, wire frame, surface modeling, solid modeling solid entity creation, Boolean operations, |

| |
|---|
| regions, editing commands, display commands, mass properties |
| Descriptive Geometry, points in space, lines in space, planes in space, key views |
| Problem Solving, graphical problem solving techniques, civil problems, iterative solutions to engineering problems, solving and back solving formulas. |

# *Semester-3rd*

| Course Code: BTES 301-18 | 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, Graycode, Excess 3code, ASCII.

**LOGIC GATES:** AND, OR, NOT, NAND, NOR, Exclusive-OR and Exclusive-NOR. Implementations of Logic Functions using gates, NAND-NOR implementations.

**Module2:**

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

**Module 3:**

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

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

**Module 4:**

**MEMORY DEVICES**: Classification of memories, RAM organization, Write operation, Read operation, Memory cycle. ROM organization, PROM, EPROM, EEPROM, Programmable logic array, Programmable array logic, complex Programmable logic devices (CPLDS), Field Programmable Gate Array (FPGA).

**A/D & D/A CONVERTORS:** Analog & Digital signals. Sample and hold circuit, A/D and D/A conversion techniques (Weighted type, R-2R Ladder type, Counter Type, Dual Slope type, Successive Approximation type).

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

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

**Suggested Readings/Books:**

1. Morris Mano, **Digital Design**, Prentice Hall of India Pvt. Ltd
2. Donald P.Leach and Albert Paul Malvino, **Digital Principles and Applications**, 5 ed., Tata McGraw Hill Publishing Company Limited, New Delhi, 2003.
3. R.P.Jain, **Modern Digital Electronics**, 3ed., 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**, Pearson Education.
6. Ghosal, **Digital Electronics**, Cengage Learning.

| **Course Code:** SE301-19 | **Course Title:** Data Structure & Algorithms | **3L:1T:0P** | **3 credits** |
|---|---|---|---|

**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)**

**Module2: Stacks and Queues**

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

**[10 hrs] (CO2, CO4, CO5)**

**Module3**: **Linked Lists**

Singly linked lists: Representation in memory, Algorithms of several operations: Traversing, Searching, Insertion into, Deletion from linked list; Linked representation of Stack and Queue, Header nodes, Doubly linked list: operations on it and algorithmic analysis; Circular Linked Lists: All operations their algorithms and the complexity analysis.

**Trees:** Basic Tree Terminologies, Different types of Trees: Binary Tree, Threaded Binary Tree, Binary Search Tree, AVL Tree; Tree operations on each of the trees and their algorithms with complexity analysis. Applications of Binary Trees. B Tree, B+ Tree: definitions, algorithms and analysis.

**[10 hrs] (CO2, CO4, CO5)**

**Module4: Sorting and Hashing**

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

**[10 hrs] (CO3)**

**Module4: Graph**

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

**[6 hrs] (CO2, CO4)**

**Course Outcomes:**

The student will be able to:

1. For a given algorithm student will able to analyze the algorithms to determine the time and computation complexity and justify the correctness;
2. Handle operation like searching, insertion, deletion, traversing on various Data Structures and determine time and computational complexity;
3. 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. 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 Structu**res", Samanta and Debasis, 2nd edition, PHI publishers.
2. "**Fundamentals of Data Structures**", Illustrated Edition by Ellis Horowitz, Sartaj Sahni, 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:** SE302-19 | **Course Title:** Computer Networks | **3L:1T: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] (CO1)**

**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] (CO2)**

**Module 3: Network Layer**

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

**[8 hrs] (CO3)**

**Module 4: Transport Layer**

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

**[8 hrs] (CO3)**

**Module 5: Application Layer**

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

**[8 hrs] (CO4)**

## Course Outcomes:

The student will be able to:

1. Explain the functions of the different layer of the OSI Protocol
2. Describe the function of each block of wide-area networks (WANs), local area networks (LANs) and Wireless LANs (WLANs);
3. Develop the network programming for a given problem related TCP/IP protocol
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

| Course Code: SE 303-18 | Course Title: Software Engineering | 3L:0T:0P | 3Credits |
|---|---|---|---|

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

| 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. Gandhian idea of development. Swaraj and Decentralization.

**PROJECT: Possible projects in this course could be**

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

| Course Code: HSMC 102- 18 | Course Title: PHILOSOPHY | 3L:0T:0P | 3Credits |
|---|---|---|---|

## Detailed Contents:

**Unit1:**

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

- Upanishads;
- Six systems orthodox and Heterodox Schools of Indian Philosophy.
- Greek Philosophy:

**Unit2** the Origin of Universe:

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

**Unit3:**
Knowledge as Power: Francis Bacon. Knowledge as both power and self-realization in Bagavad Gita.

**Unit4:**
Knowledge as oppression: M. Foucault. Discrimination between Rtam and Satyam in Indian Philosophy.

**Unit5:**
Knowledge as invention: Modern definition of creativity; scientific activity in the claim that Science invents new things at least through technology.

**Unit6:**
Knowledge about the self, transcendental self; knowledge about society, polity and nature.

**Unit7:**
Knowledge about moral and ethics codes.

**Unit8:**
Tools of acquiring knowledge: Tantrayuktis, asystem ofinquiry(Caraka, Sushruta, Kautilya, Vyasa)

**READINGS**

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

**ASSESSMENT (indicative only):**

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

**OUTCOME OF THE COURSE:**

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

| Course Code: BTES 302-18 | 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 subtract or 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

| Course Code:SE304-19 | Course Title: Data Structure & Algorithms Lab | 0L:0T:4P | 2Credits |
|---|---|---|---|

**List of Experiment:**

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

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

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

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

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

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

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

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

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

**Task10:** Write a menu driven program to perform following insertion operations in a single linked list:

i. Insertion at beginning
ii. Insertion at end
iii. Insertion after a given node
iv. Traversing a linked list

**Task11:** Write a menu driven program to perform following deletion operations in as single linked list:

i. Deletion at beginning
ii. Deletion at end
iii. Deletion after a given node

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

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

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

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

**Task16:** Program to sort an array of integers in ascending order using insertion sort.

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

**Task18:** Program to traverse a Binary search tree in Pre-order, In-order and Post-order.

**Task19:** Program to traverse graphs using BFS.

**Task20:** 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.
6. “Data Structures with C (Schaum's Outline Series)”, Seymour Lipschutz,1st edition, McGraw Hill Education.

| Course Code:SE305-19 | Course Title: Computer Networks Lab | 0L:0T:2P | 1Credits |
|---|---|---|---|

**List of Experiments:**

**Task 1:** To study the different types of Network cables and network topologies

**Task 2:** Practically implement and test the cross-wired cable and straight through cable using clamping tool and network lab cable tester.

**Task 3:** Study and familiarization with various network devices.

**Task 4:** Familiarization with Packet Tracer Simulation tool/any other related tool.

**Task 5:** Study and Implementation of IP Addressing Schemes

**Task 6:** Creation of Simple Networking topologies using hubs and switches

**Task 7:** Simulation of web traffic in Packet Tracer

**Task 8:** Study and implementation of various router configuration commands

**Task 9:** Creation of Networks using routers.

**Task 10:** Configuring networks using the concept of subnetting

**Task 11:** Practical implementation of basic network command and Network configuration commands like ping, ipconfig, netstat, tracert etc. for trouble shooting network related problems.

**Task 12:** 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 sub netting.
5. Configure routers using various router configuration commands.
6. Troubleshoot the networks by using various networking commands.

| Course Code: SE306-18 | Course Title: Software Engineering lab | 0:0T:2P | 3Credits |
|---|---|---|---|

**Detailed List of Tasks:**

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

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

# *Semester-4th*

| Course Code: SE401-19 | Course Title: Database management Systems | L:3;T:0; P:0 | 3Credits |
|---|---|---|---|

**Detailed contents:**

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

**[7hrs] (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:

**CO1:** For a given query write relational algebra expressions for that query and optimize the developed expressions

**CO2:** For a given specification of the requirement design the databases using ER method and normalization.

**CO3:** For a given specification construct the SQL queries for Open source and Commercial DBMS - MYSQL, ORACLE, and DB2.

**CO4:** For a given query optimize its execution using Query optimization algorithms

**CO5:** For a given transaction-processing system, determine the transaction atomicity, consistency, isolation, and durability.

**CO6:** 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: SE402-19 | Course Title: Computer Organization and Architecture | 3L:1T:0P | 3Credits |
|---|---|---|---|

**Pre-requisites:** Digital Electronics

## Detailed Contents:

**Module1: Functional blocks of a computer**

CPU, memory, input-output subsystems, control unit. Instruction set architecture of a CPU– registers, instruction execution cycle, RTL interpretation of instructions, addressing modes, instruction set. Case study– instruction set of8085 processor.

**Data representation**: signed number representation, fixed and floating point representations, character representation. Computer arithmetic–integer addition and subtraction, ripple carry adder, carry look-ahead adder, etc. multiplication– shift-and add, Booth multiplier, carry save multiplier, etc. Division restoring and non-restoring techniques, floating point arithmetic.

**[10 hrs] (CO1, CO2)**

**Module2: Introduction to x86 architecture.**

**CPU control unit design**: Hardwired and micro-programmed design approaches, Case study– design of a simple hypothetical CPU.

**Memory system design**: semiconductor memory technologies, memory organization. **Peripheral devices and their characteristics**: Input-output subsystems, I/O device interface, I/O transfers– program controlled, interrupt driven and DMA, privileged and non-privileged instructions, software interrupts and exceptions. Programs and processes–role of interrupts in process state transitions, I/O device interfaces–SCII, USB.

**[12 hrs] (CO2, CO4)**

**Module3: Pipelining**

Basic concepts of pipelining, throughput and speed up, pipeline hazards.

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

**[10 hrs] (CO5)**

**Module4: Memory Organization**

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

**[10 hrs] (CO3)**

## Course Outcomes:

The student will be able to:

1. Understand functional block diagram of microprocessor;
2. Apply instruction set for writing assembly language programs;
3. Design a memory module and analyse 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 Carl Hamacher, McGraw Hill Higher Education.

**Reference Books:**

1. "**Computer Architecture and Organization**", 3rd Edition by John P. Hayes, WCB/McGraw-Hill
2. "**Computer Organization and Architecture: Designing for Performance**", 10th Edition by William Stallings, Pearson Education.
3. "**Computer System Design and Architecture**", 2nd Edition by Vincent P. Heuring and Harry F. Jordan, Pearson Education.

| Course Code: SE 403-19 | Course Title: Operating Systems | 3L:1T:0P | 3Credits |
|---|---|---|---|

**Detailed Contents:**

**Module1: Introduction**

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

**[6 hrs] (CO1)**

**Module2: Processes**

Definition, Process Relationship, Different states of a Process, Process State transitions, Process Control Block (PCB), Context switching

**Thread:** Definition, Various states, Benefits of threads, Types of threads, Concept of multithreads,

**Process Scheduling**: Foundation and Scheduling objectives, Types of Schedulers, Scheduling criteria: CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time; Scheduling algorithms: Pre- emptive and Non-pre-emptive, FCFS, SJF, RR; Multiprocessor scheduling: Real Time scheduling: RM and EDF.

**[10 hrs] (CO2, CO3)**

**Module3: Inter-process Communication**

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

**[8 hrs] (CO2)**

**Module4: Deadlocks**

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

**[8 hrs] (CO2)**

**Module5: Memory Management**

Basic concept, Logical and Physical address map, Memory allocation: Contiguous Memory allocation –Fixed and variable partition– Internal and External fragmentation and Compaction; Paging: Principle of operation– Page allocation–Hardware support for paging, Protection and sharing, Disadvantages of paging.

**Virtual Memory**: Basics of Virtual Memory– Hardware and control structures–Locality of reference, Page fault, Working Set, Dirty page/ Dirty bit–Demand paging, Page Replacement algorithms: Optimal, First in First Out (FIFO), Second Chance(SC), Not recently used (NRU) and Least Recently used(LRU).

**[10 hrs] (CO4)**

**Module6: I/O Hardware**

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

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

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

**[8 hrs] (CO5, CO6)**

**Course Outcomes:**

The student will be able to:

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

**Suggested Books:**

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

**Reference Books:**

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

| **Course Code:** BTCS 404-18 | **Course Title:** Design and Analysis of Algorithms | **3L:0T:0P** | **3Credits** |
|---|---|---|---|

**Pre-requisites:** Data Structures

## Detailed Contents:

**Module1: Introduction**

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

**[8 hrs] (CO1)**

**Module 2: Fundamental Algorithmic Strategies**

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

**[10 hrs] (CO1, CO2)**

**Module 3: Graph and Tree Algorithms**

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

**[10 hrs] (CO3)**

**Module 4: Tractable and Intractable Problems**

Computability of Algorithms, Computability classes – P, NP, NP-complete and NP-hard. Cook's theorem, Standard NP-complete 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
3. (greedy/ divide and conquer/backtrack etc.);
4. Explain model for a given engineering problem, using tree or graph, and write the corresponding algorithm to solve the problems;
5. Demonstrate the ways to analyze approximation/randomized algorithms (expected running time, probability of error); &
6. Examine the necessity for NP class based problems and explain the use of heuristic techniques.

## Suggested Books:

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

## Reference Books

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

## UNIVERSAL HUMAN VALUES 2: UNDERSTANDING HARMONY

**Course code: HSMC 122-18** Credits:3

**COURSE TOPICS:**

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

### Module1: Course Introduction- Need, Basic Guidelines, Content and Process for Value Education

1. Purpose and motivation for the course, recapitulation from Universal Human Values-I
2. Self-Exploration–what is it? – Its content and process; 'Natural Acceptance' and Experiential Validation- as the process for self-exploration.
3. Continuous Happiness and Prosperity- A look at basic Human Aspirations
4. Right understanding, Relationship and Physical Facility- the basic requirements for fulfilment of aspirations of every human being with their correct priority
5. Understanding Happiness and Prosperity correctly-A critical appraisal of the Current scenario.
6. Method to fulfil the above human aspirations: understanding and living in harmony at various levels.

Include practice sessions to discuss natural acceptance in human being as the innate acceptance for living with responsibility (living in relationship, harmony and co-existence) rather than as arbitrariness in choice based on liking-disliking.

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

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

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

1. Understanding values in human- human relationship; meaning of Justice (nine Universal values in relationships) and program for its fulfilment to ensure mutual happiness; Trust and Respect as the foundational values of relationship.
2. Understanding t h e meaning of Trust; Difference between intention and competence
3. Understanding t h e meaning of Respect, Difference between respect and differentiation; the other salient values in relationship.
4. Understanding the harmony in the society (society being an extension of family): Resolution, Prosperity, fearlessness (trust) and co-existence as comprehensive Human Goals.
5. Visualizing a universal harmonious order in society- Undivided Society, Universal Order-from family to world family.

Include practice sessions to reflect on relationships in family, hostel and institute as extended family,

real life examples, teacher-student relationship, goal of education etc. Gratitude as a universal value in relationships. Discuss with scenarios. Elicit examples from students' lives.

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

1. Understanding the harmony in the Nature
2. Interconnectedness and mutual fulfilment among the four orders of nature- recyclability and self-regulation in nature
3. Understanding Existence as Co-existence of mutually interacting units in all- pervasive space
4. Holistic perception of harmony at all levels of existence.

Include practice sessions to discuss human being as cause of imbalance in nature (film "Home" can be used), pollution, depletion of resources and role of technology etc.

**Module5: Implications of the above Holistic Understanding of Harmony on**

**Professional Ethics**

1. Natural acceptance of human values
2. Definitiveness of Ethical Human Conduct
3. Basis for Humanistic Education, Humanistic Constitution and Humanistic Universal Order
4. Competence in professional ethics:
    a. Ability to utilize the professional competence for augmenting universal human order
    b. Ability to identify the scope and characteristics of people friendly and eco-friendly production systems,
    c. Ability to identify and develop appropriate technologies and management patterns for above production systems.
5. Case studies of typical holistic technologies, management models and production systems.
6. Strategy for transition from the present state to Universal Human Order:
    a. At the level of individual: as socially and ecologically responsible engineers, technologists and managers
    b. At the level of society: as mutually enriching institutions and organizations.
7. Sum up. Include practice Exercises and Case Studies will be taken up in Practice (tutorial) Sessions eg. To discuss the conduct as an engineer or scientist etc.

**2. READINGS:**

**2.1 Text Book**

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

**2.2 Reference Books**

1. Jeevan Vidya: Ek Parichaya, A. Nagaraj, Jeevan Vidya Prakashan, Amarkantak, 1999.
2. Human Values, A.N.Tripathi, New Age International Publishers, New Delhi, 2004.
3. The Story of Stuff (Book).
4. The Story of My Experiments with Truth-by Mohan das Karam chand Gandhi
5. Small is Beautiful-E.F Schumacher.
6. Slow is Beautiful-Cecile Andrews
7. Economy of Permanence- JC Kumarappa
8. Bharat Mein Angreji Raj-Pandit Sunder lal
9. Re discovering India –by Dharampal

10. Hind Swarajor Indian Home Rule-by Mohan das 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: SE405-19 | Course Title: Database management System lab | 0:0T:4P | 4Credits |
|---|---|---|---|

**List of Experiments:**

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 Statements.
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:SE406-19 | Course Title: Computer Organization & Architecture Lab | 0L:0T:2P | 1Credits |
|---|---|---|---|

**List of Experiment:**

**Task1:** Computer Anatomy- Memory, Ports, Mother board and add-on cards.
**Task2:** Dismantling and assembling PC.
**Task3:** Introduction to 8085 kit.
**Task4:** Addition of two 8 bit numbers, sum 8 bit.
**Task5:** Subtraction of two 8 bit numbers.
**Task6:** Find 1's complement of 8-bit number.
**Task7:** Find 2's complement of 8-bit number.
**Task8:** Shift an 8-bit no. by one bit.
**Task9:** Find Largest of two 8 bit numbers.
**Task10:** Find Largest among an array often numbers (8 bit).
**Task11:** Sum of series of 8 bit numbers.
**Task12:** Introduction to 8086 kit.
**Task13:** Addition and subtraction of two 16 bit numbers, sum 16 bit.
**Task 14:** Implement of Booth's algorithm for arithmetic operations.
**Task15:** Find 1's and 2'scomplement of 16-bit number.
**Task16:** Implement simple programs using I/O based interface.

## Lab Outcomes:

The student will be able to:

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

## Reference Books:

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

| Course Code: SE 407-19 | Course Title: Operating Systems Lab | 0L:0T:4P | 2Credits |
|---|---|---|---|

**List of Experiment:**

**Task1:** Installation Process of various operating systems.

**Task 2:** Implementation of CPU scheduling algorithms to find turnaround time and waiting time.

a) FCFS
b) SJF
c) Round Robin (pre-emptive)
d) Priority.

**Task3:** Virtualization, Installation of Virtual Machine Software and installation of Operating System on Virtual Machine.

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

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

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

**Lab Outcomes:**

The student will be able to:

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

**Reference Books:**

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

| Course Code: SE 408-19 | Course Title: Design and Analysis of Algorithms Lab | 0L:0T:4P | 2Credit |
|---|---|---|---|

**List of Experiment:**

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

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

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

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

**Task4:** Implementing an application of DFS such as:

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

**Task5:** Implement an application of BFS such as:

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

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

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

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

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

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

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

**Lab Outcomes:**

The student will be able to:

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

## Reference Books

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