Course Description

1st Semester

Course Name: Mathematics-I **Course Code: 18B11MA111**

L-T-P scheme: 3-1-0 **Credits: 4**

Prerequisite: Students should have basic knowledge of Algebra and calculus.

Objective: This course is aimed:

- To introduce the calculus of functions of two variables and applicability of derivatives and integrals of vector functions to Analytical geometry and physical problems.
- To make students aware of the basic mathematical concepts and methods which will help them in learning courses in engineering and Technology.

Learning Outcomes:

Course Outcome	Description
CO1	Understand the rank, eigen values, eigen vectors, diagonalization of matrix; compute inverse of matrix by Caley-Hamilton theorem.
CO2	Recognize consistent and inconsistent systems of linear equations by the row echelon form of the augmented matrix, and solve it by Gauss elimination method.
CO3	Interpret derivatives and integrals of multivariable functions geometrically and physically; implement multivariable calculus tools in engineering, science, optimization, and understand the architecture of surfaces in plane and space etc.
CO4	Know about piecewise continuous functions, Laplace transforms and its properties; use of Laplace transform and inverse transform for solving initial value problems.
CO5	Realize importance of line, surface and volume integrals, Gauss and Stokes theorems and apply the concepts of vector calculus in real life problems.
CO6	Formulate mathematical models in the form of ordinary differential equations and learn various techniques of getting solutions of linear differential equations of second order.

Course Contents:

Unit 1: Algebra of matrices, Determinants, Rank, Gauss elimination method, Eigen values and vectors. Quadratic forms.

Unit 2: Partial differentiation. Taylor's series. Maxima and minima. Jacobians**,** Double integrals,

Unit 3: Differential Equations with constants coefficients.

Unit 4: Gradient, divergence and curl. Line and surface integrals, Normal and tangent to a surface. Gauss and Stokes theorems, Equations to a line, plane, curve and surfaces.

Unit 5: Laplace transforms.

Methodology:

The course will be covered through lectures supported by tutorials. There shall be 3 Lectures per week where the teacher will explain the theory, give some examples supporting the theory and its applications. About 12 Tutorial Sheets covering whole of the syllabus shall be given. Difficulties and doubts shall be cleared in tutorials. Apart from the discussions on the topics covered in the lectures, assignments/ quizzes in the form of questions will also be given.

Evaluation Scheme:

Exams	**Marks**	**Coverage**
Test-1	15 Marks	Syllabus covered upto Test-1
Test-2	25 Marks	Syllabus covered upto Test-2
Test-3	35 Marks	Full Syllabus
Assignment	10 Marks	
Tutorials	5 Marks	
Quiz	5 Marks	
Attendance	5 Marks	
Total	**100 Marks**	

Learning Resources:

Tutorials, lecture slides and books on mathematics-1 will be available on the JUET server.

Books

1. Erwin Kreyszig: Advanced Engineering Mathematics, Wiley Publishers.
2. Lipshuts, S., Lipsom M.: Linear Algebra, 3rd Ed, Schaum series 2001.
3. B. V. Raman: Higher Engineering Mathematics, McGraw-Hill Publishers.
4. R.K. Jain, S.R.K. Iyenger: Advanced Engineering Mathematics, Narosa Publishing House, New Delhi.
5. Thomas, G.B., Finney, R.L.: Calculus and Analytical Geometry, 9th Ed., Addison Wesley,1996.
6. Grewal, B.S. : Higher Engineering Mathematics, Khanna Publishers Delhi.

Title of Course: Physics-I **Course Code: 18B11PH111**

L-T Scheme: 3-1-0 **Course Credits: 4**

Objective: Broadly, the study of Physics improves one's ability to think logically about the problems of science and technology and obtain their solutions. The present course is aimed to offer a broad aspect of those areas of Physics which are specifically required as an essential background to all engineering students for their studies in higher semesters. The course intends to impart sufficient scientific understanding of different phenomena associated with Special relativity, Modern Physics, Statistical physics, atomic physics, and lasers.

Course Outcomes:

Course Outcome	Description
CO1	Describe the limitations of Newton's laws and explain when special relativity become evant, Learn to Apply the principles of Special Relativity to an extended range of problems volving particle kinematics
CO2	Demonstrate the ability to explain the concepts related to the consequences of Special Relativity, the nature of space-time and related dynamic observables
CO3	Acquired a profound understanding of inadequacy of classical mechanics regarding phenomena related to microscopic level, Become well versed with the experimental developments, historical account and importance of probabilistic interpretation
CO4	Understand the basic quantum mechanical ideas and relevant mathematical framework, approach the solution of one dimensional time independent Schrodinger equation
CO5	Appreciate the importance of applying statistical ideas to explore thermodynamic variables, Developed ability to identify and apply appropriate statistical method for describing the assembly of microscopic particles, comprehend basic properties and working of Laser systems

Course Contents:

Unit-I (Theory of Special Relativity): Frames of reference, Galilean transformation, Michelson Morley Experiment, Postulates of special theory of relativity, time dilation and length contraction, twin paradox, Lorentz transformations, addition of velocities, Relativistic Doppler effect, Mass variation with velocity, Mass-energy relation.

Unit-II (Introduction to Modern Physics):

Quantization of Radiation, Black body radiation, Rayleigh-Jeans law, Planck's law of radiation Wien's law, Stefan's law, Photoelectric effect Compton scattering, Atomic spectra, Bohr model of hydrogen atom, Frank hertz experiment, Matter waves, de Broglie hypothesis, Davisson Germer experiment

Unit III Quantum Mechanics

Wave packets, phase and group velocity, Heisenberg's uncertainty principle, Schrödinger wave equation and its applications to the free particle in a box, potential barrier and Harmonic oscillator

Unit-IV (Statistical Mechanics): Maxwell-Boltzmann, Bose-Einstein and Fermi-Dirac distributions and their applications.

Unit- V Laser Physics & Applications

Fundamental ideas of stimulated and spontaneous emission, Einstein's coefficients, Principle and working of laser, Different types of lasers (He-Ne Laser, Ruby Laser, Semiconductor Laser), Applications of Lasers

Text Books and References:

1. A. Beiser, Perspectives of Modern Physics, Tata McGraw Hill.
2. J R Taylor, C D Zafiratos, M A Dubson, Modern Physics for Scientist &
1. Engineers, Pearson Education.
2. K Krane, Modern Physics, Wiley India
3. J Bernstein, P M Fishbane, S. Gasiorowicz, Modern Physics, Pearson
4. Education.
5. B. B. Laud, Laser and Non-Linear Optics, New Age International (P) Ltd.
6. R. Resnick, Relativity, New Age.

Title: English **Code: 18B11HS111**

L-T-P scheme: 2-1-0 **Credit:** 3

Prerequisite: None

Objective:

1. To enable understanding of basics of communication in Business environment.
2. To provide insight into structural aspect of communication in business.
3. To impart knowledge about communication theory and develop skills in oral and non verbal communication.
4. To improve skills as critical readers, thinkers, listener and writer.

Learning Outcomes:

Course Outcome	Description
CO1	Outline the basic concept of verbal/ nonverbal skills to understand the role of effective communication in personal & professional success.
CO2	Describe drawbacks in listening patterns and apply listening techniques for specific needs.
CO3	Develop the understanding to analyze, interpret and effectively summarize a variety of textual content
CO4	Discuss a given technical/non-technical topic in a group setting and arrive at generalizations/consensus.
CO5	Create effective presentations
CO6	Create professional and technical documents that are clear and adhering to all the necessary convention.

Course Content:

Unit-1: Concept and Nature of Communication : Definition of Communication, Process & Stages of Communication, Barriers to Communication, Channels of Communication.

Unit-2: Listening Skills: The listening process**,** Importance of listening, Purpose and types of listening**,** Hearing and listening, Listening with a purpose, Barriers to listening.

Unit-3: Speaking/Oral Skills: Importance of acquiring oral skills, Visual aids, Body Language, Delivery, Pronunciation, Use of connectives Organization of matter: Metadiscourse features, Textual organization, 7 C'S of effective communication , Improving vocabulary by learning Root words in English, Some foreign words, Reading comprehension, Some important synonyms and antonyms, commonly confused words, Etiquettes & grooming.

Unit-4: Reading Skills: Skimming and Scanning, Intensive and extensive reading, SQ3R Technique

Unit-5: Writing Skills: Business letters, Memo, Circulars, Notices, Report writing, resume writing, Agenda & Minutes writing, Tips on clear writing Translation- Hindi to English, Translation -English to Hindi.

Unit-6: Introduction to Modern Communication Media: Technology based communication tools, Committee types, Advantages, Conferences, Audio-video conferencing, Barriers and overcoming negative impact.

Unit-7: Public Speaking and Interviewing Strategies: Speech Preparation, Theory of group discussion, Participation in Group discussion, Oral presentation, Power point presentation ,Tips for successful job interview, Do's and don'ts while appearing for interview, Mock interview, Some interview questions, Telephonic interview tips, Resume writing

Evaluation Scheme:

Exams	Marks	Coverage
Test-1	15 Marks	Based on Unit-1 & Unit-2
Test-2	25 Marks	Based on Unit-3,& Unit-4 and around 30% from coverage of Test-1
Test-3	35 Marks	Based on Unit-5 to Unit-7 and around 30% from coverage of Test-2
Assignment	10 Marks	
Tutorials	5 Marks	
Quiz	5 Marks	
Attendance	5 Marks	
Total	**100 Marks**	

Teaching Methodology:

The course will be taught with the aid of lectures, handouts, case studies, Task-based language learning, and comprehensive language learning through language lab.

Learning Resources:

Lecture slides and e-books on ENGLISH (will be added from time to time): Digital copy will be available on the JUET server.

Text Book:

1. K.K. Sinha- Business Communication (Galgotia Publications)

Reference Books:

1. R.C. Bhatia- Business Communication (Ane Books Pvt. Ltd.)
2. P.D. Chaturvedi – Business Communication (Pearson Education, 1st Edition 2006).
3. Lesikar RV & Pettit Jr. JD – Basic Business Communication: Theory & Application (Tata Mc Graw Hill, 10thEdition)
4. Wren & Martin, High School English Grammar & Composition – S. Chand & Co. Delhi.
5. Raman Meenakshi & Sharma Sangeeta, Technical Communication-Principles & Practice –O.U.P. New Delhi. 2007.
6. Mitra Barum K., Effective Technical Communication – O.U.P. New Delhi. 2006.
7. Better Your English- a Workbook for 1st year Students- Macmillan India, New Delhi.
8. Raymond Murphy,' Essential English Grammar', Cambridge University Press.

Title: Software Development Fundamentals **Code: 18B11CI111**
L-T-P scheme: 3-1-0 **Credit: 4**

Prerequisite: There is no prerequisite in this course; however, students having any prior experience of programming are desirable.

Objective:

1. To provide exposure to problem-solving through programming.
2. To provide students with understanding of code organization and functional hierarchical decomposition with using complex data types.

Learning Outcomes:

Course Outcome	**Description**
CO1	Makes students gain a broad perspective about the uses of computers in engineering industry.
CO2	Develops basic understanding of computers, the concept of algorithm and algorithmic thinking.
CO3	Develops the ability to analyze a problem, develop an algorithm to solve it.
CO4	Develops the use of the C programming language to implement various algorithms, and develops the basic concepts and terminology of programming in general.
CO5	Introduces the more advanced features of the C language

Course Content:

Unit-1: Introduction to Programming: Basic computer organization, operating system, editor, compiler, interpreter, loader, linker, program development. Variable naming, basic function naming, indentation, usage and significance of comments for readability and program maintainability. Types of errors, debugging, tracing/stepwise execution of program, watching variables values in memory. Constants, Variables and data Types Character Set, C tokens, Keywords and Identifiers, Constants, Variables, Data types, Declaration of Variables, assigning values to variables, typedef, and Defining symbolic constants. printf & scanf function.

Unit-2: Operators and Expression: Introduction, Arithmetic Operators, Relational Operators, Logical Operators, Assignment Operators, Increment and Decrement Operators, Conditional Operators, Special Operators, Evaluation of expressions, Precedence of arithmetic operators, Type conversions in expressions, Operator precedence and associativity.

Management Input and Output Operators: Introduction, reading a character, writing a character, formatted input, formatted output.

Unit-3: Decision Making B**ranching:** Introduction, Decision making with IF statement, the IF-ELSE statement, nesting of IF-ELSE statement, ELSE-IF ladder, SWITCH statement, ternary operator, and the GOTO statement.

Looping: Introduction, the WHILE statement, the DO statement, The FOR statement, Break and Continue.

Unit-4: Array: Introduction, One-dimensional arrays, Two-dimensional arrays, arrays, Concept of Multidimensional arrays.

Handling of Character strings: Introduction, Declaring and initializing string variables, reading string from terminal, writing string to screen, String, Operations: String Copy, String Compare, String Concatenation and String Length (using predefined functions & without using them), Table of strings.

Unit-5: User-Defined Functions (UDF): Introduction, need for user-defined functions, the form of C function, elements of UDF, return values and their types, Calling a function, category of functions, Nesting of functions, Recursion, Functions with arrays, The scope and Lifetime of variables in functions, multi file program.

Structures and Unions: Introduction, Structure definition, declaring and initializing Structure variables, accessing Structure members, Copying & Comparison of structures, Arrays of structures, Arrays within structures, Structures within Structures, Structures and functions, Unions.

Unit-6: Pointers: Introduction, understanding pointers, Accessing the address of variable, Declaring and initializing pointers, accessing a variable through its pointer, Pointer expressions, Pointer increments and scale factor, Pointers and arrays, Pointers & character strings, Pointers & Functions, Function returning multiple values, Pointers and structures.

File Management in C and CONSOLE I/O: Introduction, Defining files and its Operations, Error handling during I/O operations, Random access files, Command line arguments. Types of files, File vs. Console, File structure, File attributes, Standard i/o, Formatted i/o, Sample programs.

Teaching Methodology:

This course is introduced to help students understand the discipline of programming. The programming language used to teach this course is C. Starting from the basic computer architecture, the student will slowly be exposed to program designing and later to programming fundamentals. The entire course is broken down into six separate units, from fundamentals of programming to some complex programming structures like pointers. This theory course is well complemented by a laboratory course under the name Software Development Fundamentals Lab in the same semester that helps a student learn with hand-on experience.

Evaluation Scheme:

Exams	Marks	Coverage
Test-1	15 Marks	Based on Unit-1 & Unit-2
Test-2	25 Marks	Based on Unit-3 & Unit-4 and around 20-30% from coverage till Test-1
Test-3	35 Marks	Based on Unit-5 to Unit-6 and around 30% from coverage till Test-2
Assignment	10 Marks	

Tutorials	5 Marks	
Quiz	5 Marks	
Attendance	5 Marks	
Total	**100 Marks**	

Learning Resources:

Tutorials and lecture slides on Software Development Fundamentals (will be added from time to time): Digital copy will be available on the JUET server.

Text Book:

[1] Programming in ANSI C by E. Balguruswamy, Tata Mc-Graw Hill.
[2] Programming With C, Schaum Series.

Reference Books/Material:

[1] The 'C' programming language by Kernighan and Ritchie, Prentice Hall
[2] Computer Programming in 'C' by V. Rajaraman, Prentice Hall
[3] Programming and Problem Solving by M. Sprankle, Pearson Education
[4] How to solve it by Computer by R.G. Dromey, Pearson Education

Web References:

[1] http://www2.its.strath.ac.uk/courses/c/
Notes on C programming by University of Strathclyde Computer Centre. This tutorial was awarded the NetGuide Gold Award during the 1990s.

[2] http://www.princeton.edu/~achaney/tmve/wiki100k/docs/C_%28programming_language%29.html
This site contains notes on C programming from Princeton University, USA.
These are very useful for students who are learning C as their first programming Language.

[3] http://www.stat.cmu.edu/~hseltman/Computer.html
Online reference material on Computers and Programming from Carnegie Mellon University, Pittsburgh, USA

[4] http://projecteuler.net/
Collection of mathematical problems which make you use your programming skills

Title: Physics Lab-I **Code: 18B17PH171**
L-T-P scheme: 0-0-2 **Credit: 1**

Learning Outcomes

Course Outcome	Description
CO1	Demonstrate ability to collect experimental data and understanding the working procedures within the precautionary limits
CO2	Acquired the ability to analyze the experimental data and related errors in a reflective, iterative and responsive way
CO3	Developed understanding of the basic concepts related to Modern Physics, Basic Solid State Physics and Optics
CO4	Acquired a first hand and independent experience of verifying Kirchoff's circuit laws and related concepts e.g. resistivity, measurement of resistance
CO5	Appreciate the importance of the laboratory work culture and ethics that is intended to impart features like regularity, continuity of self evaluation and honesty of reporting the data

List of Experiments

1. To study the variation of magnetic field along the axis of Helmholtz Galvanometer and to determine its reduction factor.
2. To determine the resistance per unit length of a Carey Foster's bridge and to obtain the specific resistance of a given wire.
3. To determine the wavelengths of spectral lines Red, Green and Violet of mercury using plane transmission grating.
4. To determine the specific rotation of cane sugar solution using Bi-quartz polarimeter.
5. To observe Newton's rings and to determine the wavelength of sodium light.
6. To study the CRO and function generator by producing the following waveforms.
   i. 10kHz, 8Vp-p(sine wave, square wave, triangular wave)
   ii. 4kHz, 6Vp-p(sine wave, square wave, triangular wave)
   iii. 10kHz, 8Vpeak(sine wave, square wave, triangular wave)
   iv. 4kHz, $6V_{peak}$(sine wave, square wave, triangular wave)
7. To verify the Kirchhoff's current law.
8. To verify the Kirchhoff's voltage law.

Title: Software Development Lab **Code: 18B17CI171**

L-T-P scheme: 0-0-4 **Credit: 2**

Prerequisite: Experience in programming is desirable.

Objective:

1. To provide exposure to problem-solving through programming.
2. To provide students with understanding of code organization and functional hierarchical decomposition with using complex data types.
3. To give the student hands-on experience with the concepts.

Learning Outcomes:

Course Outcome	**Description**
CO1	Makes students gain a broad perspective about the uses of computers in engineering industry.
CO2	Develops basic understanding of computers, the concept of algorithm and algorithmic thinking.
CO3	Develops the ability to analyze a problem, develop an algorithm to solve it.
CO4	Develops the use of the C programming language to implement various algorithms, and develops the basic concepts and terminology of programming in general.
CO5	Introduces the more advanced features of the C language

Course Content:

The following assignments will be carried out in synchronization with the theory classes.

Unit-1: Introduction to programming Environment (Linux commands, editing tools such as vi editor, sample program entry, compilation and execution). Development of programs using multiple arithmetic and logical operators. Programs for Roots of quadratic equation, conversion of units etc.

Unit-II: Programs using simple control statements such as if else, while, do while etc. Making a program for a calculator for example. Extracting the digits of an integer, reversing digits, finding sum of digits etc.

Unit-III: Programs using For loop, switch statement etc. For example, Finding average of numbers, printing multiplication tables etc. Checking for primes, generation of Armstrong numbers. Generation of the Fibonacci sequence, Finding the square root of a number, calculation of factorials, printing various patterns using for loop. The greatest common divisor of two integers, Raising a number to large power.

Unit-IV: Programs using Arrays: declaring and initializing arrays. Program to do simple operations with arrays. Strings – inputting and outputting strings. Using string functions such as strcat, strlen etc. Writing simple programs for strings without using string functions.Finding the

maximum number in a set, Array order reversal, Finding maximum number from an array of numbers Removal of duplicates from an ordered array,

Unit-V: Selection/ Bubble/ Insertion sort, create a linked list, traverse a linked list, insert a node and delete a node form the list. Recursion and related examples such as Tower of Hanoi, computing factorial etc. Practice sessions and sessions for missed labs

Units to Lab Mapping:

Unit	**Labs**
I	1, 2, 3
II	4, 5
III	6, 7, 8
IV	9, 10, 11
V	12, 13, 14

Teaching Methodology:

This course is introduced to help students understand the discipline of programming. The programming language used to teach this course is C. Starting from the programming environment setup, the student will slowly be exposed to program designing and later to programming fundamentals. The entire course is broken down into six separate units, from fundamentals of programming to some complex programming structures like pointers. This theory course is well complemented by a laboratory course under the name Software Development Fundamentals Lab in the same semester that helps a student learn with hand-on experience.

Evaluation Scheme:

Exams		**Marks**	**Coverage**
P-1		15 Marks	**Based on Lab Exercises: 1-6**
P-2		15 Marks	**Based on Lab Exercises: 7-13**
Day-to-Day Work	Viva	20 Marks	**70 Marks**
	Demonstration	20 Marks	
	Lab Record	15 Marks	
	Attendance & Discipline	15 Marks	
Total		**100 Marks**	

Learning Resources:

Study material of Software Development Fundamentals Lab (will be added time to time): Digital copy will be available on the JUET server.

Text Book:

1. Programming in ANSI C by E. Balguruswamy, Tata Mc-Graw Hill.
2. Programming With C, Schaum Series.

Reference Books/Material:

1. The 'C' programming language by Kernighan and Ritchie, Prentice Hall
2. Computer Programming in 'C' by V. Rajaraman, Prentice Hall
3. Programming and Problem Solving by M. Sprankle, Pearson Education
4. How to solve it by Computer by R.G. Dromey, Pearson Education

Web References:

1. http://www2.its.strath.ac.uk/courses/c/
   a. Notes on C programming by University of Strathclyde Computer Centre. This tutorial was awarded the NetGuide Gold Award during the 1990s.

2. http://www.princeton.edu/~achaney/tmve/wiki100k/docs/C_%28programming_language%29.html
   a. This site contains notes on C programming from Princeton University, USA. These are very useful for students who are learning C as their first programming Language.

3. http://www.stat.cmu.edu/~hseltman/Computer.html
   a. Online reference material on Computers and Programming from Carnegie Mellon University, Pittsburgh, USA

4. http://projecteuler.net/
   a. Collection of mathematical problems which make you use your programming skills

Title: Workshop **Code: 18B17ME171**

L-T-P scheme: 0-0-3 **Credit: 1.5**

Prerequisite: Students must have the knowledge of fundamental principles of Physic and Chemistry upto class 12^{th} which helps them to understand the various process of Workshop Lab.

Objective:

1. To demonstrate students, the basic manufacturing processes of Workshop lab: Carpentry, Fitting, Welding, Machining and Casting Processes.
2. To develop effective skills in students to identify the manufacturing process with its applications
3. To be able to perform basic manufacturing processes safely.

Learning Outcomes:

Course	**Description**
CO1	Identify the various processes of manufacturing.
CO2	Capable to explain the use of various holding, measuring, marking and cutting tools
CO3	Prepare a useful job by performing the various processes in proper sequence safely
CO4	Apply Bernoulli's theorem to analyze the liquid metal velocity in casting process.
CO5	Develop the skills to join two metallic specimen using welding process
CO6	Work as a team on a project

Course Content:

Carpentry Shop

1. To study about various tools/equipments used in carpentry shop
2. To make Cross lap /T joint as per given specification
3. To make Cross lap /T joint as per given specification

Foundry Shop

1. To study about various tools used in foundry shop.
2. To prepare a green sand mould with the help of a given pattern.
3. To perform permeability test on moulding sand

Machine Shop

1. To study various machine tools such as lathe, milling, shaper, drilling, grinding, EDM drill and cutting tools used by them.
2. To perform turning, step turning and taper turning operations on lathe machine
3. To perform threading operation on the lathe machine

Fitting Shop

1. To study about various tools used in fitting shop.
2. To make a fitting job as per given drawing.

Welding Shop

1. To study various types of welding processes available in the workshop such as Electric arc welding, TIG and MIG welding, gas welding and spot resistance welding,
2. To prepare welding joint by using Electric arc welding/gas welding
3. To prepare welding joint by using Spot Resistance welding

Teaching Methodology:

This Lab course has been introduced to help a student to learn with hand-on experience on machines. The entire course is broken down into fourteen experiments. Experiments are performed different shop wise by taking the proper safety precautions. Workshop lab includes five shops namely: Carpentry, Foundry, Machining, Fitting and Welding. Basic principles of manufacturing processes are applied to prepare a job. Students learn here how to handle the real world problems by using technical skills. The way of experimentation here realizes the students that they are now moving on an Engineering path. This Lab course will enable a student to learn with hand-on experience.

Evaluation Scheme:

Exams		**Marks**	**Coverage**
P-1		15 Marks	Based on Lab Experiments: 1-7
P-2		15 Marks	Based on Lab Experiments: 8-14
Day-to-Day Work	Viva	20 Marks	70 Marks
	Demonstration	20 Marks	
	Lab Record	15 Marks	
	Attendance & Discipline	15 Marks	
Total		**100 Marks**	

Learning Resources:

Laboratory Manual available in Lab. Study material of Workshop Lab (will be added time to time): Digital copy will be available on the JUET server.

Text Books:

[1] "Workshop Technology Volume- I & II", B.S. Raghuvanshi, Dhanpat Rai & Co.

[2] "Workshop Technology Volume-I & II", Khanna Publisher.

Reference Books:

[1] "Workshop Technology Vol.- 1, 2, 3 & 4", Butterworth-Heinemann.

[2] "Material Science & Engineering", W. D. Callister, John Wiley

Web References:

[1] https://nptel.ac.in/courses/112/107/112107219/

[2] https://nptel.ac.in/courses/112/107/112107144/

IInd Semester

Course Title: Discrete Mathematics **Course Code: 18B11MA211**

L-T-P scheme: 3-1-0 **Credits: 4**

Objectives:

The aim of the course is to cover the basic principles sets relations functions partially ordered set, lattice, Boolean algebra and its applications. The main objective of the course is to develop in student, an intuitive understanding of graphs by emphasizing on the real world problems.

Course Outcomes:

At the end of the course, the student is able to:

CO1	Employ De Moivre's theorem in a number of applications to solve numerical problems.
CO2	Appreciate the definition and basics of graphs along with types and their examples.
CO3	Visualize the applications of graph theory to network flows. Understand the notion of planarity and coloring of a graph. Relate the graph theory to the real-world problems.
CO4	Understand the definition of a tree and learn its applications to fundamental circuits.
CO5	Solve real-life problems using finite-state and Turing machines
CO6	Learn about partially ordered sets, lattices and their types, Boolean algebra and Boolean functions, logic gates, switching circuits and their applications.

Course Contents:

Unit 1: Basics of set theory, Mathematical induction. Relations, Equivalence relation, partial- ordered relation algorithms and functions.

Unit 2: Big O notation, Proposition, Basic logical operators, Propositional functions and Quantifiers.

Unit 3: Graphs and related definitions, Eulerian and Hamiltonian graphs, Graph colorings. Trees, Algebraic expressions and Polish notation, shortest path.

Unit 4: Algebraic Systems. Lattice and Boolean Algebra.

Unit 5: Language, Finite State Automata and Machines. Grammars.

Methodology:

The course will be covered through lectures supported by tutorials. Apart from the discussions on the topics covered in the lectures assignments/ quizzes in the form of questions will also be given.

Evaluation plan:

Exams	Marks	Coverage
Test-1	15 Marks	Based on Unit-1
Test-2	25 Marks	Based on Unit-2 & Unit-3 and around 30% from coverage of Test-1
Test-3	35 Marks	Based on Unit-4 to Unit-5 and around 30% from coverage of Test-2
Assignment	10 Marks	
Tutorials	5 Marks	
Quiz	5 Marks	
Attendance	5 Marks	
Total	**100 Marks**	

References:

1. B. A, Davey & H. A. Priestley (2002). "Introduction to Lattices and Order" (2nd edition) Cambridge University, Press.
2. Edgar, G. Goodaire & Michael M. Parmenter (2018). "Discrete Mathematics with Graph Theory" (3rd edition). Pearson Education.
3. Rudolf Lidl & Günter Pilz (1998). "Applied Abstract Algebra" (2nd edition).Springer.
4. Kenneth H. Rosen (2012). "Discrete Mathematics and its Applications: With Combinatorics and Graph Theory" (7th edition), McGraw-Hill.
5. C. L. Liu (1985). "Elements of Discrete Mathematics" (2nd edition). McGraw- Hill.

Title of Course: Physics-II **Course Code: 18B11PH211**

L-T Scheme: 3-1-0 **Course Credits: 4**

Objective:

Broadly, the study of Physics improves one's ability to think logically about the problems of science and technology and obtain their solutions. The present course is aimed to offer a broad aspect of those areas of Physics which are specifically required as an essential background to all engineering students for their studies in higher semesters. At the end of the course, the students will have sufficient scientific understanding of basic vector calculus, electrostatics, magnetostatics, electromagnetic fields and waves, basic understanding of physics of semiconducting materials

Course Outcomes:

Course Outcome	**Description**
CO1	Learn to apply the basic concepts of vector calculus and understanding of various co-dlinate systems and related properties, Demonstrate basic understanding of formulation and duction of electric field produced by static charge distributions
CO2	Evaluate the electrostatic field due to symmetric charge distributions, Understand the utility of formulation of electric potential and solve related problems using special techniques and boundary conditions
CO3	Acquired understanding of electrostatic fields inside matter, Explain the magnetic field due to moving charge distribution, evaluate the magnetic field due to current distribution in space,
CO4	appreciate the importance of Maxwell's equations and understand the electromagnetic wave propagation in free space Categorisation of materials on the basis of band structure
CO5	Developed understanding of quantum mechanical origin of band formation in solids, describing the energy state of electrons in crystalline materials, comprehend basic carrier properties

Course Content:

Unit I *(Electrostatics)*

Review of vector calculus, Cartesian, spherical polar and cylindrical co-ordinate systems, concept of gradient, divergence and curl, Coulomb's law, Gauss law and its applications, Boundary condition on electrostatic field, electric potential, Laplace equation, Poisson equation and related boundary value problems, capacitance, electrostatic fields in matter

[**10**]

Unit II (*Magnetostatics*)

Lorentz force, cyclotron formula, line, surface and volume currents, , Biot-Savart law and its applications, Ampere's law and its applications, equation of continuity, Faraday's law of electromagnetic induction, boundary conditions on magnetic field, Magnetic field in matter [**08]**

Unit III (Electromagnetic field)

Maxwell's equations in free space and matter, Maxwell correction to Ampere's law, Electromagnetic waves in free space and matter, Transverse nature of em waves and Polarization, Propagation of electromagnetic field in free space and Poynting vector, Poynting theorem , Normal incidence of em waves **[10]**

Unit IV ***(Elements of Solid State Physics)***

Basic ideas of bonding in solids, Crystal structure, X-ray diffraction, Band theory of solids, Distinction between metals, semiconductors and insulators **[04]**

Unit V (*Physics of Semiconductors)*

Band theory of solids, Kronig Penney model, effective mass, Direct and indirect bandgap semiconductors, optical and thermal properties, Fermi-Dirac Distribution in semi-conductors, Equilibrium carrier concentrations in intrinsic and extrinsic semiconductors, Fermi energy variation with temperature and impurity concentration, Hall Effect in semiconductors, P-N junction characteristics **[10]**

Text/ Reference Books:

1. D.J. Griffiths, *Introduction to electrodynamics*, Prentice Hall of India Ltd.
2. B.G. Streetman, S. Banerjee, Solid State Electronic Devices
3. Semiconductor Physics and Devices, Donald A. Neamen
4. Boylstad and Nashelsky, *Electronic Devices and Circuits*, PHI, 6e, 2001.
5. J. Reitz, F. Milford and R. Christy, *Foundation of Electromagnetic Theory*, Narosa Publishing.
6. J. Millman and C.C. Halkias, Electronic Devices and Circuits, Millman, McGra-Hill

Title: Electrical Science **Code: 18B11EC211**

L-T-P Scheme: 3-1-0 **Credit: 4**

Prerequisite: Students must have studied the core concepts of "*Physics-1*".

Course Objectives:

1. This course is designed for developing the understanding about basics of electrical and electronics concepts.
2. In this course students will have an enough idea about the working of systems and enable them to analyze a circuit.

Learning Outcomes:

1. The students shall acquire the generic skills to study & analyze the electrical and electronic systems.
2. This course will enable them to think and design various applications of the electrical and electronics at basic level.

The student will be able to:

Course Outcome	Description
CO1	Understand the basic electrical and electronics component and their importance determine the current, voltage and power.
CO2	Apply networks laws and theorems to solve electric circuits and may understand circuit reduction techniques with their advantages.
CO3	Understand charging discharging Steady state and transient
CO4	Demonstrate the use of semiconductor diodes in various applications.
CO5	Discuss and explain the working of transistors Amplifiers, their configurations and applications.
CO6	Analysis concept and two port networks simplification technique.

Course Content:

Unit I: Basic Electrical Circuit: Electromotive Force (EMF), Terminal Voltage; Resistance (*R*), Inductance (*L*) and Capacitance (*C*) from (i) Circuit, (ii) Energy, and (iii) Geometrical Points of View; Voltage Divider, Current Divider; Star-Delta Transformation; Voltage Source and Current Source, Source Transformation, Combination of Sources; Controlled (Dependent) Sources.

Unit 2: Methods of Analysis: Kichhoff's Circuit Laws; Loop-Current Analysis, Mesh Analysis; Node-Voltage Analysis; Choices of Method of Analysis.

Unit 3: Network Theorems (DC Circuits): Superposition Theorem; Thevenin's Theorem; Norton's Theorem; Maximum Power Transfer Theorem.

Unit 4: DC Transients: Simple *RL* Circuit, Time Constant, Decay and Growth of Current; Simple *RC* Circuit, Discharging of a Capacitor, Charging of a Capacitor.

Unit 5: Two-Port Networks: Impedance, Admittance, Hybrid, Transmission Parameters; Equivalent Networks.

Unit 6: Diodes and its Applications: Unidirectional property, *PN*-junction with no bias, with forward bias and with reverse bias, *V-I* characteristics, Comparison of Si and Ge diodes,Temperature effects, Diode resistance (static and dynamic), Diode equation, Ideal diode, Circuit model of a diode. Half-wave and full-wave (centre tap and bridge) rectifiers, PIV rating of diode, Performance of half-wave and full-wave rectifiers, Shunt capacitor filter. Clippers: Series and Parallel, Limiters, Clampers. Zener diode, Analysis of Zener voltage regulator. LED, varactor diode .

Unit 7: Transistor: BJT Structure, Working of a transistor, Transistor current equation, Collector reverse saturation current, DC alpha of a transistor. The three configurations, CB and CE input and output characteristics.

Teaching Methodology:

Lectures would be interactive and it would cover the core concepts that are explained in the text and reference materials with adequate examples.

Evaluation Scheme:

Exams	Marks	Coverage
Test-1	15	Based on Unit-1 & Unit-2
Test-2	25	Based on Unit-3, Unit-4 & Unit-5 and around 30% from coverage of Test-1
Test-3	35	Based on Unit-6 to Unit-7 and around 30% from coverage of Test-2
Assignment	10	Based on Unit-1, Unit-2 & Unit-3
Tutorials	5	Based on Unit-4 & Unit-5
Quiz	5	Based on Unit-6 & Unit-7
Attendance	5	Based on attendance in the theory classes
Total	**100**	

Learning Resources:

Tutorials sheets, lecture slides and handwritten notes on Electrical circuit, Electrical Science and Basic Electronics (will be added from time to time): Digital copy will be available on the JUET server.

Text-Books:

1. D.C. Kulshreshtha, "Basic Electrical Engineering", McGraw Hill Education, 2009.
2. W.H. Hayt, J. E. Kemerlay & S.M. Durbin, "Engineering Circuit Analysis (Sixth Edition)", McGraw Hill, 2006.
3. R.C. Dorf & J.A. Svoboda, "Introduction to Electric Circuits", John Wiley, 2004.
4. D.S. Chauhan & D.C. Kulshreshtha, 'Electronics Engineering', New Age, 2e, 2009.
5. D.C. Kulshreshtha, 'Electronic Devices and Circuits', New Age, 2e, 2006.

References:

1. Van Valkenburg, "Network Analysis", Prentice-Hall India Ltd., 2001.
2. Abhijit Chakrabarti, Sudipta Nath, Chandan Kumar Chanda, "Basic Electrical Engineering", Tata McGraw Hill Publishing Co, 2008.
3. Vincent Del Toro, "Principles of Electrical Engineering", Prentice Hall of India.
4. Kumar and Jain, 'Electronic Devices and Circuits', PHI, 2007.
5. Boylstad and Nashelsky, 'Electronic Devices and Circuits', PHI, 6e, 2001.

Web References:

1. https://www.electrical4u.com/electrical-engineering-objective-questions-mcq/
2. https://www.pdfdrive.com/basic-electric-circuit-analysis-books.html
3. https://lecturenotes.in/subject/842

Journals References:

1. Circuits, Systems, and Signal Processing (CSSP), Springer
2. Journal of Electrical & Electronic Systems
3. International Journal of Circuit Theory and Applications, Wiley

Title of Course: Object Oriented Programming **Course Code: 18B11CI211**
L-T-P Scheme: 3-1-0 **Course Credit: 4**

Prerequisites:
Students must have already registered for the course, "Software Development Fundamentals"

Objectives:
To strengthen their problem solving ability by applying the characteristics of an object-oriented approach and to introduce object oriented concepts in C++.

Learning Outcomes

Course Outcome	Description
CO1	List various principles of Object-Oriented Programming (OOP).
CO2	Describe the real world problems using object-oriented programming concepts.
CO3	Develop the programs using the fundamental concepts of OOP.
CO4	Identify and use various techniques used in OOP.
CO5	Apply techniques used in OOP to solve the software design problems on a given software project.
CO6	Demonstrate the learning on the course to solve the real life programming problems.

Course Content

Unit-1: Review of Structured programming in C, Structured versus Object-Oriented Programming, Principles of Object-Oriented Programming, Beginning with C++, Control Structures, Functions in C++, Reference Variables, Default Parameters, Function Overloading, Inline Function, Const Variables.

Unit-2: Classes, Member Functions, Objects, Static Data Members, Static Member Functions, Friend Functions, Pointer to Members, Local classes, Constructors and Destructors of objects in C++,

Unit-3: Operator overloading and Type Conversions, Inheritance and its form, Multiple Inheritance in C++, Function Overriding, Virtual Inheritance, Virtual Base Class .

Unit-4: Pointers, Early binding, late binding, Type of polymorphism, Virtual Functions, Abstract Class, Virtual Destructor

Unit-5: Managing Console I/O Operations, File handling and Exception handling.

Unit-6: Templates, Function templates, Class templates, introduction to Standard Template Library (STL), Sequence, Containers, Iterators

Teaching Methodology

The course will use the mixed technique of interactive lectures, tutorials, guided case studies, literature survey, regular assignments and project work. Teaching in this course is designed to engage the students in active and experiential learning by taking a problem solving and design-oriented approach with special emphasis on real world applications.

In the lectures the fundamental theoretical concepts will be introduced and demonstrated through examples and case studies. Discussion in lecture will be done using design problems which will be implemented in laboratory individually in C++.

Evaluation Scheme

Evaluations	**Marks**	**Remarks**
T1	15 Marks (1 Hour)	
T2	25 Marks (1.5 Hours)	
T3	35 Marks(2 Hours)	
Assignments	10 Marks	2 or 3 Assignments to given
Quiz	5 Marks	2 or 3 quizzes
Tutorials	5 Marks	
Attendance	5 Marks	
Total	**100 Marks**	

Text books

Text book1: Robert Lafore, Object oriented programming in C++, Waite Group.
Text book2: E Balagurusamy, "Object-Oriented Programming with C++"

References

1. Deitel and Deitel,"C++ How to program", - Pearson Education.
2. Stroustrap B., the C++ Programming Language, Addison Wesley.
3. Lippman F. B., C++ Primer, Addison Wesley.
4. Prata S., C++ Primer Plus, Waite Group.
5. Parimala N., Object Orientation through C++, Macmillan India Ltd. 1999.
6. Pohl I., Object oriented Programming Using C++, Addison Wesley.
7. Grady Booch, James Rambaugh, Ivar Jacobson,"Unified Modelling Language user's guide" , Addison Wesley Limited

Title of Course: Physics Lab-II **Course Code: 18B17PH271**
L-T-P Scheme: 0-0-2 **Course Credit: 1**

Learning Outcomes

Course Outcome	Description
CO1	Demonstrate ability to collect experimental data and understanding the working procedures within the precautionary limits
CO2	Acquired the ability to analyze the experimental data and related errors in a reflective, iterative and responsive way
CO3	Developed understanding of the basic concepts related to Modern Physics, Basic Solid State Physics, Optics,
CO4	Acquired a first hand and independent experience of verifying the working principle of solar cell
CO5	Appreciate the importance of the laboratory work culture and ethics that is intended to impart features like regularity, continuity of self evaluation and honesty of reporting the data

Experiments List

1. To determine the magnetic susceptibility of a paramagnetic, $FeCl_3$ solution by Quinck's tube method.
2. To determine dispersive power of a prism using spectrometer.
3. To study the magnetostriction in metallic rod using Michelson-Interferometer.
4. To determine the Planck's constant using Photo electric effect.
5. To study the Hall effect in P type semi conductor and to determine
   (i) Hall voltage and Hall coefficient
   (ii) Number of charge carriers per unit volume
   (iii) Hall angle and mobility
6. To study the variation of resistivity of a semiconductor with temperature and to determine the band gap using Four-Probe method.
7. To study the presence of discrete energy levels in an atom by Franck Hertz experiment.
8. Using solar cell Trainer (a) study voltage and current of a solar cell (b) Voltage and current in series and parallel combinations (c) Draw power curve to find maximum power point (MPP) and to obtain efficiency of a solar cell.

Title: Electrical Science Lab

Code: 18B17EC271

L-T-P Scheme: 0-0-2 **Credit: 1**

Prerequisite: Student must have already registered for the course, "*Physics Lab-I*"

Objective:

1. The main aim of the lab is to familiarize with different types of electrical and electronic circuits
2. Identify their applications to the different electrical and electronic systems.

Learning Outcomes:

1. Completion of lab students will be able to understand the different techniques to simplify circuit
2. Two port networks and basic principles of different electronic devices and their characteristics.

Course Outcome	**Description**
CO1	Simplify complex network using Thevenin theorem and verify it.State Superposition Theorem and verify.Perform and verify Maximum Power Transfer Theorem.
CO2	To determine the Z parameters of the given two port network. Calculate the Y parameters for the given two port network.
CO3	V-I characteristic of p-n junction diode
CO4	Design Clipper and Clamper Circuit.
CO5	Rectifier circuits
CO6	Transistor and their v-I characteristics

Course Content:

1. Simplify complex network using Thevenin theorem and verify it.
2. State Superposition Theorem and verify.
3. Perform and verify Maximum Power Transfer Theorem.
4. To determine the Z parameters of the given two port network.
5. Calculate the Y parameters for the given two port network.
6. Perform Clipper Circuit.
7. Design Clamper Circuit.
8. Half wave rectifier with and without filter circuit.
9. Full wave rectifier with and without filter circuit.
10. Transistor as an Amplifier.
11. Common Emitter *v-i* characteristic of n-p-n transistor.
12. Common base *v-i* characteristic of n-p-n transistor.

Unit I: Basic Electrical Circuit

Voltage Divider, Current Divider; Kichhoff's Circuit Laws; Loop-Current Analysis, Mesh Analysis; Node-Voltage Analysis; Choices of Method of Analysis. Source Transformation, Combination of Sources; series and parallel combination of resistors.

Unit 2: Network Theorems (DC Circuits)

Superposition Theorem; Thevenin's Theorem; Norton's Theorem; Maximum Power Transfer Theorem.

Unit 3: Two-Port Networks

Impedance, Admittance, Hybrid, Transmission Parameters; Equivalent Networks.

UNIT 4: Diodes and its Applications

Unidirectional property, *PN*-junction with no bias, with forward bias and with reverse bias, *V-I* characteristics, Diode resistance (static and dynamic), Diode equation, Ideal diode, Circuit model of a diode. Half-wave and full-wave (centre tap and bridge) rectifiers, PIV rating of diode, Performance of half-wave and full-wave rectifiers, Shunt capacitor filter.

Clippers: Series and Parallel, Limiters, Clampers. Zener diode, Analysis of Zener voltage regulator. LED, varactor diode .

UNIT 5: Transistor

BJT as an amplifier, CB and CE input and output characteristics.

Teaching Methodology:

In each experiment the practical is designed and analyzed on bread board with the help of physical devices by each student and further checked and validated by faculty and lab staff.

Evaluation Scheme:

<table>
<tr><td>Exams</td><td colspan="2">Marks</td><td>Coverage</td></tr>
<tr><td>P-1</td><td colspan="2">15 Marks</td><td>Based on Lab Exercises: 1-6</td></tr>
<tr><td>P-2</td><td colspan="2">15 Marks</td><td>Based on Lab Exercises: 6-12</td></tr>
<tr><td rowspan="4">Day-to-Day Work</td><td>Viva</td><td>20 Marks</td><td rowspan="4">70 Marks</td></tr>
<tr><td>Demonstration</td><td>20 Marks</td></tr>
<tr><td>Lab Record</td><td>15 Marks</td></tr>
<tr><td>Attendance & Discipline</td><td>15 Marks</td></tr>
<tr><td colspan="2">Total</td><td colspan="2">100 Marks</td></tr>
</table>

Learning Resources:

Tutorials sheets, lecture slides and handwritten notes on Electrical circuit, Electrical Science and Basic Electronics (will be added from time to time): Digital copy will be available on the JUET server.

Text-Books:

1. D.C. Kulshreshtha, "Basic Electrical Engineering", McGraw Hill Education, 2009.
2. W.H. Hayt, J. E. Kemerlay & S.M. Durbin, "Engineering Circuit Analysis (Sixth Edition)", McGraw Hill, 2006.
3. R.C. Dorf & J.A. Svoboda, "Introduction to Electric Circuits", John Wiley, 2004.
4. D.S. Chauhan & D.C. Kulshreshtha, 'Electronics Engineering', New Age, 2e, 2009.
5. D.C. Kulshreshtha, 'Electronic Devices and Circuits', New Age, 2e, 2006.

References:

1. Van Valkenburg, "Network Analysis", Prentice-Hall India Ltd., 2001.
2. Abhijit Chakrabarti, Sudipta Nath, Chandan Kumar Chanda, "Basic Electrical Engineering", Tata McGraw Hill Publishing Co, 2008.
3. Vincent Del Toro, "Principles of Electrical Engineering", Prentice Hall of India.
4. Kumar and Jain, 'Electronic Devices and Circuits', PHI, 2007.
5. Boylstad and Nashelsky, 'Electronic Devices and Circuits', PHI, 6e, 2001.

Web References:

1. https://www.electrical4u.com/electrical-engineering-objective-questions-mcq/
2. https://www.pdfdrive.com/basic-electric-circuit-analysis-books.html
3. https://lecturenotes.in/subject/842

Journals References:

1. Circuits, Systems, and Signal Processing (CSSP), Springer
2. Journal of Electrical & Electronic Systems
3. International Journal of Circuit Theory and Applications, Wiley

Title of Course: Object Oriented Programming Lab **Course Code: 18B17CI271**
L-T-P Scheme: 0-0-2 **Course Credit: 1**

Pre-requisites
Students must have already registered for the course, "Software Development Fundamentals Lab".

Objectives

To strengthen their problem solving ability by applying the characteristics of an object-oriented approach and to introduce object oriented concepts in C++.

Learning Outcomes

CO1	Define basic concepts of Object-Oriented Programming (OOP).
CO2	Illustrate the key features available in OOP using C++.
CO3	Apply the concepts of OOP to solve different common problems.
CO4	Utilize the knowledge of OOP in solving programming problems.
CO5	Analyze the various concepts of OOP for their suitability on a given problem.
CO6	Design the systems, from concept to executable artefact, using object oriented techniques.

Course Content

Unit-1: Structured versus Object-Oriented Programming, Principles of Object-Oriented Programming, Beginning with C++, Control Structures, Functions in C++, Reference Variables, Default Parameters, Function Overloading, Inline Function, Const Variables.

Unit-2: Classes, Member Functions, Objects, Static Data Members, Static Member Functions, Friend Functions, Pointer to Members, Local classes, Constructors and Destructors of objects in C++,

Unit-3: Operator overloading and Type Conversions, Inheritance and its form, Multiple Inheritance in C++, Function Overriding, Virtual Inheritance, Virtual Base Class .

Unit-4: Pointers, Early binding, late binding, Type of polymorphism, Virtual Functions, Abstract Class, Virtual Destructor

Unit-5: Managing Console I/O Operations, File handling and Exception handling.

Unit-6: Templates, Function templates, Class templates, introduction to Standard Template Library (STL), Sequence, Containers, Iterators

Laboratory work and project

The students shall be given regular lab assignments, which will allow them to practically apply the concepts studied in the lecture Session. The lab assignments will be designed with focus on applying the concepts learnt in object-oriented programming, Data structures in an integrated manner.

Evaluation Scheme

Evaluations		**Marks**	**Remarks**
P-1		15 Marks	
P-2		15 Marks	
Continuous Evaluations	Viva	20 Marks	
	Demonstration	20 Marks	
	Lab Record	15 Marks	
	Discipline and Punctuality and Attendance	15 Marks	
Total		**100 Marks**	

Text book

Text Book1: Robert Lafore, Object oriented programming in C++, Waite Group
Text Book2: E Balagurusamy, "Object-Oriented Programming with C++"

References

1. Stroustrap B., the C++ Programming Language, Addison Wesley.
2. Lippman F. B., C++ Primer, Addison Wesley.
3. Prata S., C++ Primer Plus, Waite Group.
4. Parimala N., Object Orientation through C++, Macmillan India Ltd. 1999.
5. Pohl I., Object oriented Programming Using C++, Addison Wesley.
6. Grady Booch, James Rambaugh, Ivar Jacobson,"Unified Modelling Language user's guide" , Addison Wesley Limited

Title: Engineering Drawing & Design Lab **Code: 18B17ME271**
L-T-P scheme: 0-0-3 **Credits: 1.5**

OBJECTIVE

[1] Enables students to learn the concepts of graphic communication, their role in sanitary construction.
[2] Make familiar with different drawing equipment, technical standards and procedures for construction of geometric figures.
[3] Equipped with the skill that enables them to convert pictorial to orthogonal representations.

Learning Outcomes:

Course Outcome	**Description**
CO1	**Outline** the objectives of scale and develop the imagination and mental visualization capabilities for correlating the geometrical details of objects.
CO2	To **develop** the constructional ability for a different curve.
CO3	To **Describe** BIS rules for orthogonal projection and understand the fundamental concept of orthogonal projection for point, line, plane and solids.
CO4	Understand and **apply** orthogonal projection for solids, section and intersection of solid objects/structures
CO5	To **apply** the skill of development of surfaces of three dimensional objects for evaluation of black size of the components.
CO6	**Demonstrate** computer aided drafting tools and techniques using CAD software's

Course Content:

Unit-1: Study and construction of lines, lettering, dimensioning, plane scales, diagonal scales, construction of different methods used for the construction of conic curves.
Unit-2: Study and construction of geometrical construction, cycloidal curves, involutes and helix etc.
Unit-3: Orthogonal projection of point in all possible positions, Study and construction of projection of line and its applications (inclined to both planes), and projection of planes (inclined to both planes).
Unit-4: Study and construction of projection of solids (right circular cone, prism, pyramid and cylinders), and true shape of sections,
Unit-5: Study and construction of oblique projection and development of surface, isometric view using orthogonal projection on isometric scales.
Unit-6: Introduction to basic and editing command of CAD software, 2-D drafting, surface modeling, and 3-D geometrical model.

Teaching Methodology:

This course is introduced to build the imagination and established the correlation between the real object and engineering drawing and CAD developed by the design engineers and the requirement of the production engineers of the different units.

Evaluation Scheme:

Exams		Marks	Coverage
P-1		15 Marks	**Based on Lab Exercises: 1-7**
P-2		15 Marks	**Based on Lab Exercises: 8-14**
Day-to-Day Work	Viva	20 Marks	**70 Marks**
	Demonstration	20 Marks	
	Lab Record	15 Marks	
	Attendance & Discipline	15 Marks	
Total		**100 Marks**	

Learning Resources:

The study material of engineering drawing & design lab (will be added time to time): Digital copy will be available on the JUET server.

Text Book:

1. Bhatt, N.D., Engineering Drawing,

Reference Books:

1. Gill, PS, A Text Book of Engineering Drawing (Geometrical Drawing)
2. Dhananjay A J, Engineering Drawing with an introduction to Auto CAD, Mc Graw Hill

3rd Semester:

Title: Techniques for Decision Making **Code**: **18B11HS311**
L-T-P scheme: 2-1-0 **Credit: 3**

Prerequisite: None

Objectives:

1. To use basic techniques of inferential data analysis, quality control, and regression modeling;
2. To analyze a set of data, to reach a conclusion based on these analyses, and to make and defend a recommended course of action;
3. To be well-equipped to take courses in Marketing, Investments, Accounting, Finance, and Operations Management that require proficiency in statistical methods.

Learning Outcomes:

Course Outcome	Description
CO1	Outline various concepts of techniques for decision making with respect to the needs of modern business management.
CO2	Describe the real world problems using basic techniques of descriptive and inferential data analysis and business forecasting.
CO3	Identify and use various index numbers used in business decision making.
CO4	Apply decision making techniques to reach a conclusion based on the data analysis, and to make and defend a recommended course of action.
CO5	Deployment and proficiency in statistical methods.
CO6	Develop the understanding to analyze a set of data using correlation analysis and regression analysis.

Course Content:

Unit-1: Collection of data and Presentation of data: Classification of data, Secondary data, Primary data, Designing of questionnaire, Unstructured and structured questionnaire, Tabulation of data, Charting of data.

Unit-2: Business Forecasting: Introduction, steps in forecasting, good forecasting, Time series forecasting, secular trend, seasonal variations, cyclical variations.

Unit-3: Index numbers: Uses, classification, problems, Methods of constructing index numbers, unweighted index numbers, Consumer Price index numbers.

Unit-4: Statistical Decision making : Decision making under certainity, Risk , uncertainty and conflict, Zero sum game, Prisoner's dilemma , Payoff Table, Maximin and minimax strategy.

Unit-5: Correlation Analysis and Regression analysis: Significance of the study of correlation, Correlation and causation, Karl Pearson's coefficient of correlation, Rank correlation, Method of least squares, Difference between correlation and regression,

Regression lines and regression equation, Regression equation of Y on X and regression equation of X on Y.

Teaching Methodology:

The course "Techniques for Decision Making" is introduced to explain the basic concepts in statistics that have wide applicability in business decision making. As such, the focus will be more practical than theoretical. Because statistical analysis informs the judgment of the ultimate decision-maker—rather than replaces it—we will cover some key conceptual underpinnings of statistical analysis to insure that the students understand its proper usage. Statistics is about improved decision-making, which can be achieved through a thorough understanding of the data. We want to leave our pre-conceived notions at the door, and let the data tell us what is going on in a situation. The analytical techniques should provide valuable information to decision-makers. As such, it plays an important role in management decision processes. The course will be taught with the aid of lectures, tutorials, handouts, case studies, and problem-based learning.

Evaluation Scheme:

Exams	Marks	Coverage
Test-1	15 Marks	Based on Unit-1 & Unit-2
Test-2	25 Marks	Based on Unit-3 & Unit-4 and around 30% from coverage of Test-1
Test-3	35 Marks	Based on Unit-5 and around 30% from coverage of Test-2
Assignment	10 Marks	
Tutorials	5 Marks	
Quiz	5 Marks	
Attendance	5 Marks	
Total	100 Marks	

Learning Resources:

Lectures, tutorials and e-books on Techniques for Decision Making (will be added from time to time): Digital copy will be available on the JUET server.

Text Book:

1. "Business Statistics"; S.P. Gupta & M.P. Gupta, S. Chand Publishing, New Delhi, 2013.

Reference Books/Material:

1. "Statistics for Business & Economics"; Anderson, Thomson Learning, Bombay.
2. "Quantitative Methods in Business"; Anderson, Thomson Learning, Bombay.
3. "Business Statistics"; R.S. Bhardwaj, Excel Books.
4. "Statistics for Management"; Levin & Rubin, Prentice Hall of India, New Delhi.
5. "Two Person Game Theory"; A. Rapport & Anne Arbric, The University of Michigan Press, 1966.

Title of Course: Data Structures **Course Code: 18B11CI311**
L-T-P Scheme: 3-1-0 **Credits: 4**

Scope and Objectives:

This course develop problem solving ability using programming, develop ability to express solutions to problems clearly and precisely, develop ability to design and analyze algorithms, introduce with fundamental data structures, develop ability to design and evaluate abstract data types and data structures.

Learning Outcome:

The students shall acquire the generic skills to design and implement data structures and related algorithms for a broad-based set of computing problems**.**

18B11CI311: Data Structures	
Course Outcome	**Description**
CO1	List various types of data structures with respect to their requirements in different fields.
CO2	Describe the various methods to evaluate the algorithms.
CO3	Develop algorithms based on linear data structures
CO4	Identify the suitability of the data structures as per the requirements.
CO5	Apply data structures to solve the software design problems.
CO6	Demonstrate the learning on the course to solve the real life programming problems.

Course Description:

This course is intended to provide a thorough introduction to the use of data structures in programming. This course will cover the necessary mathematical background, but will assume the required programming experience.

Course Contents:

UNIT 1: Introduction to Data Structures, Algorithm and Complexity

Data structure overview, need of data structure and how to select relevant data structure for given problem, basic C data types and ADT.

Algorithm overview and its properties, problem analysis and construction of algorithm, difference between algorithm, program and software, algorithm analysis and complexity, asymptotic notations to represent the time complexity, Software Development Life Cycle (SDLC) phase

UNIT 2: Array

Overview, memory representation of 1D and 2D array, sparse matrix, operation supported by an array

Part 1: Searching

Linear search with illustration, analysis of linear search, binary search (iterative) and its analysis, binary search (recursive) and its analysis using recurrence relation, recurrence relation

Part 2: Sorting

Types of sorting algorithms, bubble sort, selection sort, insertion sort, quick sort, merge sort

UNIT 3: Linked List

Overview, types of linked list, linear linked list – overview, traversing, insertion, deletion, searching and reverse, doubly linked list – overview, traversing, insertion, deletion, circular linked list – overview, header linked list, applications of linked list

UNIT 4: Stack

Overview, stack implementation using stack and linked list, basic operations on stack using array and linked list – push, pop, dispose applications of stack – evaluation of mathematical expression, conversion of expression from one form to another (Polish Notation), Tower of Hanoi problem

UNIT 5: Queue

Overview, basic operations on queue – enqueue, dequeue, implementation of queue using array and linked list, types of queue - linear queue, circular queue, deque, priority queue, applications

UNIT 6: Tree

Tree definition and its terminology, representation of graph using array and linked list, tree traversals – preorder, inorder and postorder, binary search tree (BST) with insertion, deletion and searching operations, extended binary tree and its application in Huffman tree, threaded binary tree

UNIT 7: Graph

Introduction to graph, types of graph, traversal algorithms in graph – breadth first search, depth first search, spanning tree, minimum cost spanning tree - Kruskal’s, Prim’s.

Evaluation Scheme:

Component & Nature	Duration	Marks / Weightage
T1	1 hr	15
T2	1&1/2 hrs	25
T3	2hrs	35
Tutorials		05
Attendance		05
Quiz		05
Assignments		10
Total		**100**

Text Book::

T1: Sartaj Sahni, "Fundamentals of Data Structures", Tata Mc Graw Hill, New York
T2: Seymour Lipschutz., "Data Structures with C", Schaum's Outline Series
T3: Narasimha Karumanchi, "Data Structures and Algorithms" Made Easy

Reference Books:

R1: Corman et al: Introduction to Computer Algorithms
R2: Langsam, Augestein, Tenenbaum: Data Structures using C and C++
R3: Weiss: Data Structures and Algorithm Analysis in C/C++
R4: Samir K. Bandyopadhyay," Data Structures using C"
R5: Hopcraft, Ullman: Data Structures and Algorithms

Title of Course: Digital Systems and Microprocessor **Course Code: 18B11EC311**

L-T-P Scheme: 3-1-0 **Credits:4**

Course Objective:

Digital Systems and Microprocessor Course is the Second year's course which is totally based on study and designing Digital components, digital circuits using basic components, types of signals on which these devices works and at last the study of the Microprocessor basics in a single course. This course aims to introduce students with a fundamental understanding of digital electronics and its application, Produce digital circuit, how signals are formed and further applications of microprocessor with all conditions.These undergraduate students will be equipped to play valuable roles in the Information Technology,Electronics and Communication industries.

Learning Outcomes:

Digital Systems and Microprocessor	
Course Outcome	**Description**
CO1	**Outline various number systems of Digital Electronics with respect to the requirements of the computer systems used in technical industries fulfilling the user requirement.**
CO2	**Solving various problems based on the number systems, complements techniques, compute simple arithmetic operations addition, subtraction, multiplication & division including ability to prove implication problems using truth table method, Boolean method etc. considering the real world examples.**
CO3	**Design Karnaugh map and Quine McCluskey method to get simplified form of a Boolean function.**
CO4	**Design combinational and sequential digital functions.**
CO5	**Understanding the various types of signals used for the various explained devices and getting knowledge of trans-receiving the signals using explained devices.**
CO6	**Introduction of Microprocessor with its interfaces and basic coding understanding utilized in it. Understand the features and architecture of 16 bit Microprocessor.**
CO7	**Understand the data types and addressing modes of 8086 Microprocessor. Demonstrate deployment and basic maintenance skills.**

Teaching Methodology:

Lectures would be interactive and it would cover the core concepts that are explained in the text and reference materials with adequate examples. Tutorials will have conceptual and numerical questions that would aid in strengthening the Digital electronics, signals and Microprocessors principles.Keeping in view the student's background, starting from number system to Basic pulse circuits design, the student will cover the study of basic signal types and application of microprocessors. In this course a student will learn about various digital components and designing digital circuits and moreover he will study about the various

sequential and combinational circuits using basic gates and K-Map designing using the same gates. After this he will be taught combinational and sequential circuits which will make him proficient in designing any digital circuit. After this the basic knowledge of types of signals will be taught which will make them to learn how to implement these digital circuits over different types of signals and at last they will be taught about the Microprocessor basics which will guide them how Microprocessor world is more emphasizing on basics of Digital Electronics. And at the end of the course, successful students should have knowledge of and ability to apply the Mathematics and scientific concepts required by Digital Electronic engineers, basic level of knowledge of and ability to apply the concepts, principles and theories of Computing and IT, as likely to be required by a Digital Electronic engineer, detailed knowledge of and ability to apply the essential facts, concepts, principles and theories needed by Digital Electronic engineers.

Course Outline:

Unit I:
Conversion of bases, Representation of negative numbers, 9's and 1's complement, 10's and 2's complement, Binary arithmetic, BCD code, Excess-3 code, Gray Code and Alphanumeric code. Logic gates and Boolean algebra, Standard and canonical representation and minimization of Boolean expressions using Karnaugh Map and Quine – McClausky methods.

Unit II:
Half & full adder and subtractor, Parallel adder, BCD adders, Lookahead carry generator. Decoders, Encoders, Multiplexers and De-multiplexers, Code convertor, Comparator, Parity generator and Checker. Binary multiplier.

Unit III:
Flip Flops: SR, JK, Master slave JK, T and D. Shift Registers and their Applications. Synchronous and Asynchronous counters, ROM, PROM, EPROM, EEPROM.

Unit IV:
Basics of Signals and Systems, Elements of a communication system, Continuous-time and discrete-time signals, signal energy and power, Periodic signals, even-odd signals, Exponential and Sinusoidal Signals.

Unit V:
Evolution of Microprocessor, Cache Memory, 8085 Architecture and its pin descriptions.

Evaluation Scheme:

Exams	**Marks**	**Coverage**
Test-1	15 Marks	Based on Unit-1, Unit-2 (30%)
Test-2	25 Marks	Based on Unit-2 (70%), Unit-3 and around 30% from coverage of Test-1
Test-3	35 Marks	Based on Unit-4 & Unit-5 and around 30% from coverage of Test-2
Assignment	10 Marks	
Tutorials	5 Marks	
Quiz	5 Marks	
Attendance	5 Marks	
Total	**100 Marks**	

Learning Resources:

Tutorials and lecture slides on Web Development (will be added from time to time): Digital copy will be available on the JUET server.

Text Books :

1. Morris Mano, Digital Logic and Computer Design, PHI
2. Taub and Schilling, Digital Integrated Electronics, McGraw Hill, Int. Ed.
3. Signal and Systems, 2nd Edition, PHI Publications, India 1997 by Alan V. Oppenheim, Alan S. Willsky, S. Hamid Nawab.
4. Fundamentals of Microprocessors and Microcontrollers, 7th edition, Dhanpat Rai Publication, India, 2010 by B. Ram.
5. Introduction to Microprocessors, Wiley Eastern (Latest Edition) R.S. Gaonkar.

Web References:

1. https://nptel.ac.in/courses/117106086/
2. http://web.iitd.ac.in/~shouri/eel201/lectures.php
3. https://www.geeksforgeeks.org/digital-electronics-logic-design-tutorials
4. https://www.electrical4u.com/digital-electronics

Journals References:

1. IEEE Transactions on Circuits and Systems
2. International Journal of Electronics by Taylor and Francis
3. AEÜ - International Journal of Electronics and Communications by Elsevier

Title of Course: Database Systems **Course Code: 18B11CI312**
L-T-P Scheme: 3-0-0 **Course Credits: 3**

Objectives: To develop the ability to design, implement and manipulate databases as well as to build Database management systems

Learning Outcome:

1. Ability to build normalized data bases.
2. Ability to design systems by using ER Modeling.
3. Ability to develop skills of writing applications by using SQL.
4. Ability to understand query optimization techniques.
5. Understanding of transaction processing.
6. Ability to handle recovery and concurrency issues

18B11CI312: Database Systems	
Course Outcome	**Description**
CO1	Introduction various types of database systems with respect to their features and charterstics and requirements in different fields.
CO2	Describe the various data definition, manipulation and various modifiers queries for database design.
CO3	Develop algorithms based on linear data structures
CO4	Develop the database using relational database query, Identify the suitable of the data structures as per the requirements.
CO5	Develop the normalized database with features of transaction, concurrency and recovery control
CO6	Demonstrate the learning on the course to deployed the database systems basis of the real life database problems.

Course Contents:

Introduction to Databases, Database Environment, Relational Model, Relational Algebra, SQL: Data Manipulation, Data Definition, And Commercial RDMS: MS-Access/MySQL, PL/SQL, ER Modeling: Entity type, Attributes, Relation types, Notations, Extended ER Features, Normalisation and building normalized databases & Data Dependencies, Case Study, Database Connectivity: Python MySQL Connectivity, Transactions, Concurrency, Recovery & Security, Query Processing & Optimization.

Text Book

1. "Database system concepts", Henry F Korth, Abraham Silberschatz, S. Sudurshan, McGraw-Hill, 4th Edition.

References

1. “An Introduction to Database Systems” Bipin. C. Desai. Revised Edition 2006.
2. "Fundamentals of Database Systems", Elmasri, Navathe, Pearson Education, IVth Edition.
3. “An Introduction to Database Systems”, C. J. Date, Pearson Education.
4. “Introduction to Data Base Management”, Naveen Prakash, Tata McGraw Hill.
5. “Database Management Systems” , Ramakrishna, Gehrke; McGraw-Hill.
6. “Database Systems: A Practical Approach to design, Implementation and Management”, Thomas Connolly, Carolyn Begg; Third Edition, Pearson Education.
7. “A first course in Database Systems”, Jeffrey D. Ullman, Jennifer Windon, Pearson Education
8. “Data Management: databases and organization”, Richard T. Watson, Wiley Publication.
9. “Data Modeling Essentials”, Graeme C. Simxion, Dreamtech Publications.
10. MS-ACCESS Projects “Oracle 8i manuals”.

Title: Environmental Science **Code: 18B11GE311**

L-T-P Scheme: 2-0-0 **Credit: 2**

Prerequisite: The students must be aware of basic Environmental Science upto class 12^{th}. Basic knowledge of Environmental Science helps them to correlate in various division of Engineering during this course.

Objective:

The purpose behind this course is to make the students familiar with Environment (surrounding) and to understand the significance/importance of natural resource, biodiversity, environment pollution and impact of intervention of human being in the Ecosystem. This course is mandatory for all branches of the Engineering and Sciences.

Course Learning Outcomes:

Course Outcome	Description
CO1	The outline, outcomes and attributes provide students with learning experiences that help in learning the significance and importance of environment in their life.
CO2	Describe the real world problems, challenges with the suitable case study based on conservation (natural resource and biodiversity), ecosystem, socio-economic development and remedial measure of the various pollutions (air, water, soil, noise and radiation).
CO3	Develop in students the ability to apply the knowledge and skills they have acquired to the solution of specific theoretical and applied problems in their surrounding (the Environment).
CO4	Identify and use of various techniques for solving the Environmental Problems.
CO5	Apply filed visit and justification by using various analytical techniques.
CO6	Demonstrate students with the knowledge and skill base that would enable them to undertake further studies in the Environmental Science and related multidisciplinary areas that involve Environmental Science and help to develop a range of generic skills that are relevant to wage employment, self-employment and entrepreneurship.

Modules	Description	No. of lectures
Unit 1:	Introduction to Environmental Science: Multidisciplinary nature of environmental science; components of environment –atmosphere, hydrosphere, lithosphere and biosphere. Scope and importance; Concept of sustainability and sustainable development.	2
Unit 2:	Ecosystems: What is an ecosystem? Structure and function of ecosystem; Energy flow in an ecosystem: food chain, food web and ecological succession. Case studies of the following ecosystems: a) Forest ecosystem b) Grassland ecosystem c) Desert ecosystem d) Aquatic ecosystems (ponds, streams, lakes, rivers, oceans, estuaries)	4
Unit 3:	Natural Resources: Renewable and Non-renewable Resources • Land Resources and land use change; Land degradation, soil erosion and desertification. • Deforestation: Causes and impacts due to mining, dam building on environment, forests, biodiversity and tribal populations. • Water: Use and over-exploitation of surface and ground water, floods, droughts, conflicts over water (international & inter-state). • Heating of earth and circulation of air; air mass formation and precipitation. • Energy resources: Renewable and non-renewable energy sources, use of alternate energy sources, growing energy needs, case studies.	5
Unit 4:	Biodiversity and its conservation: Levels of biological diversity: genetic, species and ecosystem diversity; Biogeography zones of India; Biodiversity patterns and global biodiversity hot spots. • India as a mega-biodiversity nation; Endangered and endemic species of India. • Threats to biodiversity: habitat loss, poaching of wildlife, man-wildlife conflicts, biological invasions; Conservation of biodiversity: In-situ and Ex-situ Conservation of biodiversity. • Ecosystem and biodiversity services: Ecological, economic, social, ethical, aesthetic and Informational value.	4
Unit 5:	Environmental Pollution: Environmental pollution: types, causes, effects and controls; Air, water, soil, chemical and noise pollution. • Nuclear hazards and human health risks. • Solid waste management: Control measures of urban and industrial waste. • Pollution case studies.	5
Unit 6:	Environmental Policies & Practices: Climate change, global warming, ozone layer depletion, acid rain and impacts on human communities and agriculture.• Environment Laws : Environment Protection Act; Air (Prevention & Control of Pollution) Act; Water (Prevention and control of Pollution) Act; Wildlife Protection Act; Forest Conservation Act; International agreements; Montreal and Kyoto protocols and conservation on Biological Diversity (CBD). The Chemical Weapons Convention (CWC). • Nature reserves, tribal population and rights, and human, wildlife conflicts in Indian context.	4
Unit 7:	Human Communities and the Environment Human population and growth: Impacts on environment, human health and welfares.	4

	• Carbon foot-print. • Resettlement and rehabilitation of project affected persons; case studies. • Disaster management: floods, earthquakes, cyclones and landslides. • Environmental movements: Chipko, Silent valley, Bishnios of Rajasthan. • Environmental ethics: Role of Indian and other religions and cultures in environmental conservation. • Environmental communication and public awareness, case studies (e.g., CNG vehicles in Delhi).	
Unit 8:	Field Work: Visit to a local area to document assets-river / forest / grassland /hill / mountain. polluted sites(Urban, rural ,industrial, agriculture), plants, insects, bird, Ecosystem (pond, river, hill slopes etc)	4
	Total	32

Teaching Methodology:

The core module Syllabus for Environment Science includes class room teaching and Field Work. The syllabus is divided into eight units covering lectures. The first seven units will cover 28 lectures, which are class room based to enhance knowledge skills and attitude to environment. Unit eight is based on field activities which will be covered in 4 lecture hours and would provide student firsthand knowledge on various local environmental aspects. Field experience is one of the most effective learning tools for environmental concerns. This moves out of the scope of the text book mode of teaching into the realm of real learning in the field, where the teacher merely acts as a catalyst to interpret what the student observes or discovers in his/her own environment. Field studies are as essential as class work and form an irreplaceable synergistic tool in the entire learning process. Course material provided by UGC for class room teaching and field activities is utilized.

Evaluation Scheme:

Exams	Marks	Coverage
Test-1	15 Marks	Based on Unit-1 Unit 2 and Unit-3
Test-2	25 Marks	Based on Unit-4 & Unit-5 (70 %) and around 30% from coverage of Test-1
Test-3	35 Marks	Based on Unit-6 to Unit-7 and around 30% from coverage of Test-1 and Text-2
Assignment	10 Marks	
Tutorials	5 Marks	
Quiz	5 Marks	
Attendance	5 Marks	
Total	**100 Marks**	

Learning Resources:

Tutorials and lecture slides on Web Development (will be added from time to time): Digital copy will be available on the JUET server.

Text Book

1. Bharucha Erach, 2003. The Biodiversity of India, Mapin Publishing Pvt. Ltd, Ahmadabad – 380013, India.
2. De Anil Kumar, Environmental Chemistry, Wiley Eastern Ltd, 2007.
3. Agarwal KC, 2001. Environmental Biology, Nidhi Publishers Ltd. Bikaner.

Reference Book

1. Brunner RC, 1989, Hazardous Waste Incineration, McGraw Hill Inc. 480pgs.
2. Clark R B, Marine Pollution, Clanderson Press, Oxford (TB).2001.
3. Cunningham WP, Cooper TH, Gorhani E & Hepworth MT, 2001. Environmental Encyclopedia, Jaico Publishing House, Mumbai, 1196 pgs.
4. Gleick HP, 1993. Water in Crisis, Pacific Institute for Studies in Development, Environment and Security. Stockholm Environmental Institute, Oxford University Press, 473pgs.
5. Heywood VH, and Watson RT, 1995. Global Biodiversity Assessment. Cambridge University Press 1140pgs.
6. Jadhav H and Bhosale VM, 1995. Environmental Protection and Laws. Himalaya Publishing House, Delhi 284pgs.
7. Mckinney ML and Schoch RM, 1996. Environmental Science Systems and Solutions. Web enhanced edition, 639pgs.

Title of Course: Data Structures Lab **Course Code: 18B17CI371**
L-T-P Scheme: 0-0-2 **Credits: 1**

Scope and Objectives:

This course develop problem solving ability using programming, develop ability to express solutions to problems clearly and precisely, develop ability to design and analyze algorithms, introduce with fundamental data structures, develop ability to design and evaluate abstract data types and data structures.

Learning Outcome:

The students shall acquire the generic skills to design and implement data structures and related algorithms for a broad-based set of computing problems

18B11CI371: Data Structures Lab	
CO1	Define basic operations on linear data structures
CO2	Illustrate the efficiency of a data structures in terms of time and space complexity.
CO3	Apply the data structures solve the searching and sorting problems.
CO4	Utilize the knowledge of non-linear data structures in solving programming problems.
CO5	Analyze the data structures for their suitability on a given problem.
CO6	Design the systems, from concept to executable artefact using data structures techniques.

Course Description:

This course is intended to provide a thorough introduction to the use of data structures in programming. This course will cover the necessary mathematical background, but will assume the required programming experience.

Course Contents:

UNIT 1: Introduction to Data Structures, Algorithm and Complexity

Data structure overview, need of data structure and how to select relevant data structure for given problem, basic C data types and ADT.

Algorithm overview and its properties, problem analysis and construction of algorithm, difference between algorithm, program and software, algorithm analysis and complexity, asymptotic notations to represent the time complexity, Software Development Life Cycle (SDLC) phase

UNIT 2: Array

Overview, memory representation of 1D and 2D array, sparse matrix, operation supported by an array

Part 1: Searching

Linear search with illustration, analysis of linear search, binary search (iterative) and its analysis, binary search (recursive) and its analysis using recurrence relation, recurrence relation

Part 2: Sorting

Types of sorting algorithms, bubble sort, selection sort, insertion sort, quick sort, merge sort

UNIT 3: Linked List

Overview, types of linked list, linear linked list – overview, traversing, insertion, deletion, searching and reverse, doubly linked list – overview, traversing, insertion, deletion, circular linked list – overview, header linked list, applications of linked list

UNIT 4: Stack

Overview, stack implementation using stack and linked list, basic operations on stack using array and linked list – push, pop, dispose applications of stack – evaluation of mathematical expression, conversion of expression from one form to another (Polish Notation), Tower of Hanoi problem

UNIT 5: Queue

Overview, basic operations on queue – enqueue, dequeue, implementation of queue using array and linked list, types of queue - linear queue, circular queue, deque, priority queue, applications

UNIT 6: Tree

Tree definition and its terminology, representation of graph using array and linked list, tree traversals – preorder, inorder and postorder, binary search tree (BST) with insertion, deletion and searching operations, extended binary tree and its application in Huffman tree, threaded binary tree

UNIT 7: Graph

Introduction to graph, types of graph, traversal algorithms in graph – breadth first search, depth first search, spanning tree, minimum cost spanning tree - Kruskal’s, Prim’s.

Text Book:

T1: Sartaj Sahni, “Fundamentals of Data Structures”, Tata Mc Graw Hill, New York
T2: Seymour Lipschutz., “Data Structures with C”, Schaum's Outline Series
T3: Narasimha Karumanchi, “Data Structures and Algorithms” Made Easy

Reference Books:

R1: Corman et al: Introduction to Computer Algorithms
R2: Langsam, Augestein, Tenenbaum: Data Structures using C and C++
R3: Weiss: Data Structures and Algorithm Analysis in C/C++
R4: Samir K. Bandyopadhyay,” Data Structures using C”
R5: Hopcraft, Ullman: Data Structures and Algorithms

Evaluation Scheme:

Component & Nature	**Marks**
Lab work	40
Lab record	15
Mid sem lab –Viva/Test	15
End sem lab – Viva/Test	15
Attendance & discipline in lab	15
Total	**100**

Title of Course: Database Systems Lab **Course Code: 18B11CI373**
L-T-P Scheme: 0-0-2 **Course Credit: 1**

Objectives: To develop the ability to design, implement and manipulate databases as well as to build Database management systems.

Learning Outcome

1. Ability to design systems by using ER Modeling.
2. Ability to develop skills of writing applications by using SQL.
3. Ability to understand query optimization techniques and transaction processing.

18B11CI373: Database Systems Lab	
CO1	Define basic requirement and operations of file based and database systems.
CO2	Illustrate the relational database design using data definition, data manipulation queries.
CO3	Develop the database using relational database query, Identify the suitable of the data structures as per the requirements.
CO4	Utilize the knowledge of structured query language to develop and deploy the database for real life based problems.
CO5	Develop the normalize database for their suitability on a given problem.
CO6	Design the database systems, from concept to executable transaction, concurrency and recovery control using the real time based problems in group project based task .

Course Contents:

- SQL queries for the creation of tables and insertion of values into tables.
- SQL queries for viewing all data and specific data corresponding to a particular row or column in a table.
- SQL queries for the updation, deletion and dropping of tables.
- SQL queries for aggregation, range finding etc on the tables.
- SQL queries for renaming, truncating and destroying the tables.
- SQL queries for the use of not null, group by, having clause.
- SQL queries for the computation done on the table data.
- Exercise on nested SQL queries and sub queries.
- Use of cursors, triggers, functions and writing pl/sql block.
- A brief idea about oracle report builder.

Evaluation scheme:

Exams		Marks	Coverage
P-1		15 Marks	**Based on Lab Exercises: 1-7**
P-2		15 Marks	**Based on Lab Exercises: 8-14**
Day-to-Day Work	Viva	20 Marks	**70 Marks**
	Demonstration	20 Marks	
	Lab Record	15 Marks	
	Attendance & Discipline	15 Marks	
Total		**100 Marks**	

Text Book

1. SQL, PL/SQL the Programming Language of Oracle, Ivan Bayross, 3rd edition.

4th Semester

Title of Course: Computer Networks **Course Code: 18B11CI411**

L-T-P Scheme: 3-0-0 **Credit: 3**

Objective:

The objective of this course is to build basic concepts of Computer network established for the data communication. This course also aims to provide the fundamental concepts in the design and implementation of networks, their protocols and its applications.

Learning Outcomes:

Computer Networks +Lab	
Course Outcome	**Description**
CO1	Outline basics to advanced concepts and techniques of Computer networks.
CO2	Describe problem solving approaches as applied in Data communication networking areas.
CO3	Analyse performance of basic communication networks using both analytical and simulation techniques.
CO4	Develop the Computer network design techniques and practical implementation issues.
CO5	Understand the basic properties of internet and data traffic properties.
CO6	Apply verification and validation techniques on a given software project.
CO7	Demonstrate deployment and basic maintenance skills.

Course Content:

Unit I: Introduction: Introduction to computer network, classification of networks WAN, MAN, LAN), distributed systems, digital signals and data rates, bit stream, symbols and band rate, transmission media, modems, structure of computer network, circuit, packet, message switching, Network topological, Network model, ISO-OSI model, TCP/IP model, primitives and services.

Unit II: Physical Layer: Physical Layer Design Issues (Service provided to data link Layer) Introduction Transmission media, RS-232-C and RS-449, Line coding, Switching Techniques.

Unit III: Data Link Layer: Data Link Layer Design Issues (Service Provided to N/w Layer), Framing, error control, flow control, Link Management, Error Detection and Error Correction Coding, Data Link Protocols (Elementary and sliding Window), local and metropolitan area networks. The Medium Access sub layer, Static and Dynamic Channel Allocation in LANs and MANs, ALOHA Protocols (Pure and Slotted), Different Protocols of LAN, IEEE Standard 802 for LAN (802.2, 802.4, 802.5).

Unit IV: Network Layer: Network Layer Design Issues (Service Provided to Transport Layer). Routing, Congestion, Internetworking. Routing Algorithms, Congestion Control Algorithm Internetworking, congestion control. Design issues, buffer management, synchronization. Session and presentation layer synchronization issues, formatting, data compression, data security.

Unit V: Transport Layer: Transport Layer Design Issue .Connection Management, Buffer Management, Quality of Service. Session Layer Design Issues Synchronization issues. Introduction to Presentation Layer. Encryption and decryption. RSA algorithm.

Teaching Methodology

This course will help the students to facilitate interaction and information transfer over large distances. With internet, computer and telephone networks, buisenesses can allocate their resources efficiently. The Students will be able to learn basic concepts of computer network, its working principle & operation of Internet and Intranet. They will also learn the working principle of operation of LAN, WAN, MAN, congestion in the network and network management.

Evaluation Scheme:

Exams	**Marks**	**Coverage**
Test-1	15 Marks	Based on Unit-1, Unit-2
Test-2	25 Marks	Based on Unit-2, 3 & Unit-4 and around 30% from coverage of Test-1
Test-3	35 Marks	Based on Unit-4 and Unit-5 and around 30% from coverage of Test-2
Assignment	10 Marks	

Tutorials	5 Marks	
Quiz	5 Marks	
Attendance	5 Marks	
Total	**100 Marks**	

Learning Resources:

Tutorials and lecture slides on Telecommunication networks (will be added from time to time): **Text Books:**

1. A.S. Tennenbaum, Computer Networks, PHI
2. W. Stallings, Data & Computer Communication, PHI
3. Forouzen, Behrouz A.Fegan, Sophia Chung Data Communications and Networking, TMH

Reference Books:

1. Carne, E. Bryan Professional's Guide to Data Communication in a TCP/IP World Artech House, London, 2004
2. Young, Margret Levine Internet: The Complete Reference, Tata McGraw Hill, New Delhi, 2002

Web References:

1. **www.britannica.com**
2. **www.vssut.ac.in**

Journals References:

1. International Journal on Advances in Telecommunications
2. Journal of Network and Computer applications- Elsevier
3. IEEE transactions on networking
4. ACM Journals on networking

Title of Course: **Algorithms and Problem Solving** **Course Code: 18B11CI412**
L-T Scheme: **3-1** **Course Credit**: **4**

Prerequisites:

Student must have already registered for "Introduction to Computer and programming" (07B11CI101), Data Structures (07B21CI102).

Objectives:

- Strengthen higher level cognitive Skills of analysis, creation and evaluation.
- Strengthen Ability of data abstraction and problem solving using computer
- Strengthen ability to express solution to problem clearly and precisely.
- Strengthen ability to design and evaluate ADTs, nonlinear temporary and persistent data structures and also related algorithms.
- Introduce students to some domain specific data structures and related algorithms in various domains.

Learning Outcomes:

Upon completion of the subject, students will be able to:

1 Get **familiar** with different basic concepts of algorithms and analyze the performance of algorithms.
2 Have a **good grounding** of advance data structures like R-B Tree, M – way tree, models and IDEs.
3 Get to **learn** about various algorithm design techniques for developing algorithms.

4 Possess **demonstrative skills** in solving optimization problems.
5 Be able to **design, develop algorithms,** and employ appropriate data structures for solving real world computing problems efficiently.

Course Content:

Analysis of algorithm: Asymptotic Notation, Sorting and merging Algorithm
Tree and related data Structure: Heap, Priority Queues, B+ Tree, AVL, Splay Tree, Red-Black Tree, Threaded Tree
Files: Classification, Record Organization, Retrieval System, External Sorting
Set, Dictionary: Design, Analysis, integration and applications
Fundamental techniques: Divide and Conquer method, Dynamic Programming, Introduction to Greedy Method
Hashing: technique, collision resolution and analysis
Text Processing: String operation, pattern matching algorithm, tries, text compression, text similarity testing.

Teaching Methodology:

The Course will use the mixed technique of interactive lectures, guided case studies, literature survey, regular assignments and project work. In addition to the material covered in the class, student will be required to explore study, evaluate present and implement domain specific data structure in different domain. Teaching in this course is designed to engage the student in active and experimental learning by taking a problem solving and design oriented approach with special emphasis on real world applications. Lectures will be highly interactive and work oriented. Student will have to work individually as well as in groups inside as well as outside the class. Students are expected to carry out a lot of design and programming oriented project work. Each student is expected to write minimum 3000 lines of documented program code as part of this course. Students are encouraged to learned use toolkits like STL for project implementation. Each student is also expected to do literature survey making use of the library and web resources (including digital library) to identify ,understand ,summarize and present at least one research paper on science and application of non-linear data structure and algorithms.

Evaluation Scheme (Theory):

Evaluation Scheme is designed to promote and test higher level thinking skills and de-emphasis rote learning through holistic and continuous evaluation. Written exam will be designed and conducted as open Book(s), open notes tests. One of the minor tests may me designed and conducted as a take home test. Evaluation scheme will have following components

Test-1	15 Marks
Test-2	25 Marks
Test-3	35 Marks
Home assignment /Quizzes	10 Marks
Tutorial/Problem solving session	10 Marks
Attendance	05 Marks
Total	**100** Marks

Text book

T1: Thomas H., Coremen: Introduction to algorithm, the Massachusetts institute of Technology, Cambridge, Massachusetts.

Reference Books:

1. Aho, Hopcraft, Ullman: Data Structure and Algorithms
2. Kruse, Tonso, Leung: Data Structure and program Design in C
3. Sahni: Data structure and algorithm and application in C++
4. Weiss: Data Structure and Algorithm analysis in C/C++

Title of Course: **Operating Systems** **Course Code: 18B11CI413**

L-T-P scheme: 3-0-0 **Credit: 3**

Prerequisite: Students must have knowledge of C programming and working of the computer systems.

Objective:

1. To familiarize with the basic functionality and the evolution of different types of operating systems.
2. To Learn and understand various algorithms related to CPU scheduling, deadlocks, memory management, and storage management.
3. To learn basic aspects of real time operating systems.

Learning Outcomes:

Course Outcome	Description
CO1	Gain knowledge of OS fundamentals along with process management concepts
CO2	Apply various process management concepts including scheduling, synchronization, dead-lock to solve given problem.
CO3	Explain various memory management techniques including virtual memory.
CO4	Analyse issues related to memory management.
CO5	Understand file system including disk structure by applying disk scheduling algorithm.
CO6	Work as a team on a project.

Course Content:

Unit-1: Introduction: Operating system structure, Operating system operations, Distributed systems, Special purpose systems, Computing environments, Open source operating systems.

Unit-2: CPU Scheduling: Process concepts: Process states, Process control block, Scheduling queues, Schedulers, Context switch, Multi-threaded programming: Overview, Multithreading models, Threading issues, Process scheduling: Basic concepts, Scheduling criteria, scheduling algorithms.

Unit-3: Synchronization: The Critical section problem, Synchronization hardware, Semaphores, Classic problems of synchronization, monitors.

Unit-4: Deadlocks: Deadlock characterization, Methods for handling deadlocks, Deadlock prevention, Deadlock avoidance, Deadlock detection, Recovery from deadlock.

Unit-5: Memory management: Memory management strategies, Swapping, Contiguous memory allocation, Paging, Structure of the page table, Segmentation.

Unit-6: Virtual Memory: Demand paging, copy on write, page replacement, allocation of frames, thrashing.

Unit-7: Storage Management: File concept, Access methods, directory structure, file system structure, directory implementation, allocation methods, free space management, disk structure, and disk-scheduling.

Unit-8: Case study on UNIX based Operating system: Design principles, Kernel modules, Process management, Memory management.

Unit-9: Real time systems: Characteristics of Real time operating systems, classification of real time systems, Micro kernels and RTOS, scheduling in RTOS, Rate monotonic scheduling, EDF, Priority inversion

Evaluation Scheme:

Exams	Marks	Coverage
Test-1	15 Marks	Based on Unit-1, Unit-2 & Unit-3
Test-2	25 Marks	Based on Unit-4 & Unit-5 and around 30% from coverage of Test-1
Test-3	35 Marks	Based on Unit-6 to Unit-9 and around 30% from coverage of Test-2
Assignment	10 Marks	
Tutorials	5 Marks	
Quiz	5 Marks	
Attendance	5 Marks	
Total	**100 Marks**	

Learning Resources:

Tutorials and lecture slides on Operating Systems (will be added from time to time): Digital copy will be available on the JUET server.

Text Book:

1. "Operating System Concepts"; A. Silberschatz , P. B. Galvin & G. Gagne , Wiley 10e 2018.
2. "Operating Systems: Internals and Design Principles"; W. Stallings, Pearson 9e, 2017.

Reference Books/Material:

1. "Real time systems design and analysis"; P. A. Laplante & S. J. Ovaska, Wiley, 2013.
2. "Real time systems: Theory and Practice"; Mall R., Pearson, 2e, 2009.

Title: Artificial Intelligence & Cognitive Science **Code: 18B11CI611**

L-T-P scheme: 3-0-0 **Credit: 3**

Prerequisite: Students must have already studied courses, "*Data Structure and Algorithm*" and "Object Oriented Programming Language".

Objective:

1. To learn basic components of an intelligence system their functions.
2. To develop the abilities to map the human behavior into machine device mechanisms, policies and techniques
3. To implement the artificial intelligence in software applications using examples of real time scenario.

Learning Outcomes:

Course Outcome	**Description**
CO1	Demonstrate knowledge of the building blocks of AI as presented in terms of intelligent agents.
CO2	Analyze and formalize the problem as a state space, graph, design heuristics and select amongst different search or game-based techniques to solve them.
CO3	Develop intelligent algorithms for constraint satisfaction problems and design intelligent systems for Game Playing.
CO4	Attain the capability to represent various real-life problem domains using logic-based techniques and use this to perform inference or planning. playing.
CO5	Formulate and solve problems with uncertain information using Bayesian approaches.
CO6	Apply concept Natural Language processing to problems leading to understanding of cognitive computing.

Course Content:

Unit-1: (a) Introduction to AI:

Definitions, Goals of AI, AI Approaches, AI Techniques, Branches of AI, Applications of AI.

(b) Introduction of Intelligent Systems

Agents and Environments, Good behaviors: the concept of Rationality, The Nature of Environments, The structure of Agents, How the components of agent programs work.

(c) Introduction of cognitive science

What is Cognitive Science, Philosophy: Foundations of Cognitive Science, Cognitive Psychology: The Architecture of the Mind, Artificial Intelligence: Search, Control and Learning.

Unit-2: Problems Solving, Search and Control Strategies:

Solving Problems by Searching**,** Study and analysis of various searching algorithms. Implementation of Depth-first search, Problem-Solving Agents, Searching for Solutions, Uninformed Search Strategies: Breadth-first search, Uniform-cost search, Depth-first search, Depth-limited search, Iterative deepening depth-first search, Bi-directional search Informed (Heuristic) Search Strategies: Greedy best-first search A* search: Minimizing the total estimated solution cost, Conditions for optimality: Admissibility and consistency, Optimality of A*, Memory-bounded heuristic search, Heuristic Functions, Generating admissible heuristics from sub problems: Pattern databases, Learning heuristics from experience.

Beyond Classical Search

Local Search Algorithms and Optimization Problems: Hill-climbing search Simulated annealing, Local beam search, Genetic algorithms, Local Search in Continuous Spaces, Searching with Non-deterministic

Actions: AND-OR search trees, Searching with Partial Observations

Unit-3: Adversarial Search and Constraint Satisfaction Problems:

Adversarial Search: Games, Optimal Decisions in Games, The mini-max algorithm, Optimal decisions in multiplayer games, Alpha--Beta Pruning, Move ordering, Imperfect Real-Time Decisions, Evaluation functions, Cutting off search, Forward pruning, Search versus lookup, Stochastic Games, Evaluation functions for games of chance, Partially Observable Games

Constraint Satisfaction Problems: Defining Constraint Satisfaction Problems, Variations on the CSP formalism, Constraint Propagation: Inference in CSPs, Backtracking Search for CSPs, Local Search for CSPs, Alpha-beta pruning and CSP, Implementation aspects of mini-max algorithm and CSP.

Unit-4: Knowledge Representations Issues, Predicate Logic, Rules:

Knowledge representation, KR using predicate logic, KR using rules

Unit-5: **Reasoning System - Symbolic, Statistical**

Reasoning, Symbolic reasoning, Statistical reasoning

Unit-6: Quantifying Uncertainty, Learning Systems

Acting under Uncertainty, Basic Probability Notation, Inference Using Full Joint Distributions, Bayes' Rule and Its Use, Representing Knowledge in an Uncertain Domain, Other Approaches to Uncertain Reasoning, Rule-based methods for

uncertain reasoning, Representing vagueness: Fuzzy sets and fuzzy logic, Study of fuzzy logic and Decision trees, Implementation aspects of Decision trees.

Learning from Examples:
Forms of Learning, Supervised Learning, Learning Decision Trees, The decision tree representation, Expressiveness of decision trees, inducing decision trees from examples.

Unit-7: Expert Systems:

Introduction, Knowledge acquisition, Knowledge base, Working memory, Inference engine, Expert system shells, Explanation, Application of expert systems.

Unit-8: Fundamentals of Neural Networks:

Introduction and research history, Model of artificial neuron, Characteristics of neural networks, Learning methods in neural networks, Single-layer neural network system, Applications of neural networks

Unit-9: Natural Language Processing:
Introduction, Syntactic processing, Semantic and pragmatic analysis.

Teaching Methodology:

Lectures would be focused towards the practical approach to map the intelligence behavior into software or hardware applications and it would cover the core concepts that are explained in the text and reference materials with adequate real time case study.

Evaluation Scheme:

Exams	**Marks**	**Coverage**
Test-1	15 Marks	Based on Unit-1 & Unit-2
Test-2	25 Marks	Based on Unit-3, Unit-4 & Unit-5 and around 30% from coverage of Test-1
Test-3	35 Marks	Based on Unit-6 to Unit-9 and around 30% from coverage of Test-2
Assignment	10 Marks	
Tutorials	5 Marks	
Quiz	5 Marks	
Attendance	5 Marks	
Total	**100 Marks**	

Learning Resources:

Tutorials and lecture slides on Artificial Intelligence and Cognitive Science (will be added from time to time): Digital copy will be available on the JUET server.

Text Books:

1. Rich, Elaine Knight, Kevin, Artificial Intelligence, Tata McGraw Hill.
2. Luger, George F, Artificial Intelligence: Structures and Strategies for Complex Problem Solving, Pearson Education.
3. Friedenberg & Silverman; Cognitive Science: An Introduction to the study of mind. SAGE Publications,3rd edition.

References:

1. Nilsson, Nils J, Artificial Intelligence, Morgan Kaufmann
2. Russell, Stuart J. Norvig, Peter, Artificial Intelligence: A Modern Approach, Pearson Education
3. Negnevitsky, Michael , Artificial Intelligence: A Guide to Intelligent Systems, Addison -Wesley.
4. Introduction to Cognitive Science by Paul Thagard.
5. The MIT Encyclopedia of the Cognitive Science by Robert A. Wilson and Frank C. Keil.

Title: Algorithms Lab **Code: 18B17CI472**
L-T-P scheme: 0-0-2 **Credit: 1**

Prerequisite: Experience in programming is desirable. Student must have already registered for "Software Development Lab" (18B17CI171) and "Data Structures lab" (18B17CI371).

Objective:

3. To provide exposure to problem-solving through programming.
4. Strengthen higher level cognitive Skills of analysis of problem, creation of solution and evaluation of performance.
5. Strengthen Ability of data abstraction and problem solving using computer
6. Strengthen ability to express solution to problem clearly and precisely.
7. Strengthen ability to design and evaluate ADTs, nonlinear temporary and persistent data structures and also related algorithms.
8. Introduce students to some domain specific data structures and related algorithms in various domains.

Learning Outcomes:

Course Outcome	Description
CO1	Design new algorithms, prove them correct, and analyze their asymptotic and absolute runtime and memory demands.
CO2	Find an algorithm to solve the problem (create) and prove that the algorithm solves the problem correctly (validate).
CO3	Understand the mathematical criterion for deciding whether an algorithm is efficient, and know many practically important problems that do not admit any efficient algorithms.
CO4	Apply classical sorting, searching, optimization and graph algorithms.
CO5	Understand basic techniques for designing algorithms, including the techniques of Recursion, Divide-and-Conquer, Greedy Algorithms and Dynamic Programming

Course Content:

The following assignments will be carried out in synchronization with the theory classes.

Unit-1: Development of programs including analysis of algorithm Asymptotic Notation, Sorting and merging Algorithm.

Unit-II: Programs using Heap, Priority Queues, B-Tree, AVL, Splay Tree, Red-Black Tree, Threaded Tree.

Unit-III: Programs using Classification, Record Organization, and Retrieval System of files External Sorting. Design, Analysis, integration of set & dictionary, collision resolution and analysis

Unit-IV: Programs using Divide and Conquer method, Dynamic programming, Introduction to Greedy Method.

Unit-V: Program using String operation, pattern matching algorithm, tries, text compression, text similarity testing application.

Units to Lab Mapping:

Unit	Labs
I	1, 2, 3
II	4, 5
III	6, 7, 8
IV	9, 10, 11
V	12, 13, 14

Teaching Methodology:

This course is introduced to help students understand the designing and analysis of algorithm. Any (C, C++, JAV etc) programming language used to implement algorithms. Starting from the programming environment setup, the student will slowly be exposed to program designing and later to complexity analysis fundamentals. The entire course is broken down into five separate units, from fundamentals of algorithms to some complex algorithms designing methodology like Dynamic Programming Greedy Techniques etc.

Evaluation Scheme:

Exams		**Marks**	**Coverage**
P-1		15 Marks	**Based on Lab Exercises: 1-6**
P-2		15 Marks	**Based on Lab Exercises: 7-13**
Day-to-Day Work	Viva	20 Marks	**70 Marks**
	Demonstration	20 Marks	
	Lab Record	15 Marks	
	Attendance & Discipline	15 Marks	
Total		**100 Marks**	

Learning Resources:

Study material of Algorithms Lab (will be added time to time): Digital copy will be available on the JUET server.

Text Book:

1. Thomas H., Coremen: Introduction to algorithm, the Massachusetts institute of Technology, Cambridge, Massachusetts.

Reference Books/Material:

1. Aho, Hopcraft, Ullman: Data Structure and Algorithms
2. Kruse, Tonso, Leung: Data Structure and program Design in C
3. Sahani: Data structure and algorithm and application in C++
4. Weiss: Data Structure and Algorithm analysis in C/C++

Online Courses:
NPTEL-Algorithms and Problem Solving: https://nptel.ac.in/courses/106/105/106105164/

Videos Available on YouTube:
https://www.youtube.com/watch?v=OQ5jsbhAv_M

https://www.youtube.com/watch?v=huQojf2tevI
https://www.youtube.com/watch?v=sSno9rV8Rhg

Website

- https://www.geeksforgeeks.org
- https://www.indiabix.com
- https://www.includehelp.com
- https://www.tutorialspoint.com
- https://www.sanfoundry.com
- https://www.programiz.com

Coding Platforms

- https://www.codechef.com
- https://www.hackerrank.com
- https://www.interviewbit.com
- https://www.spoj.com
- https://www.hackerearth.com
- https://leetcode.com

Integrated Development Environment

- Turbo C++
- Dev-c++
- Code::Block

Title of Course: **Operating Systems Lab** **Course Code: 18B17CI473**

L-T-P scheme: 0-0-2 **Credit: 2**

Prerequisite: Students must have knowledge of C programming and working of the computer systems.

Objective:

1. To execute shell scripts in UNIX based operating system.
2. To implement inter process communication using system calls.
3. To implement algorithms for CPU scheduling as well as process synchronization learn and be able to implement the front-end and back-end web-technologies.

Learning Outcomes:

Course Outcome	**Description**
CO1	Understand and execute basic commands of shell script.
CO2	Apply basic operations in shell scripts which are required for different applications.
CO3	Identify and understand concept of file systems in shell script.
CO4	Apply concept of creating new process from parent process.
CO5	Apply concept of virtual file and execute basic commands on it.
CO6	Design communication mechanisms ipc and pipe on linux.

Course Content:

Unit-1; Comparative Study of different operating systems
Unit-2: Demonstration of multitasking concept.
Unit-3: Implementing various process creation algorithms(FCFS,SJF and Round-Robin Scheduling)
Unit-4: Implementation of memory allocation policies.
Unit-5: Implementing Page replacement algorithms (FIFO,LIFO)
Unit-6: Implementing segmentation algorithms
Unit-7: Implementing file-handling algorithms
Unit-8: Demonstration of working of distributed OS environment.

Evaluation Scheme:

Exams		**Marks**	**Coverage**
P-1		15 Marks	**Based on Lab Exercises: 1-4**
P-2		15 Marks	**Based on Lab Exercises: 5-8**
Day-to-Day Work	Viva	20 Marks	**70 Marks**
	Demonstration	20 Marks	
	Lab Record	15 Marks	
	Attendance & Discipline	15 Marks	
Total		**100 Marks**	

Learning Resources:

Study material of Web Technology Lab (will be added time to time): Digital copy will be available on the JUET server.

Text Book:

1. "Operating System Concepts"; A. Silberschatz , P. B. Galvin & G. Gagne , Wiley 10e 2018.
2. "Operating Systems: Internals and Design Principles"; W. Stallings, Pearson 9e, 2017.

Reference Books/Material:

1. "Real time systems design and analysis"; P. A. Laplante & S. J. Ovaska, Wiley, 2013.
2. "Real time systems: Theory and Practice"; Mall R., Pearson, 2e, 2009.

Title of Course: Artificial Intelligence Lab **Course Code: 18B17CI671**

L-T-P Scheme: **0-0-2** **Credits:** 1

Objectives: In this course we will practical knowledge of the basic components of an intelligence system, their functions, mechanisms, policies and techniques used in their implementation and examples.

Learning Outcomes:

Course Outcome	**Description**
CO1	Apply various AI search algorithms (uninformed, informed, heuristic, constraint satisfaction,).
CO2	Understand the fundamentals of knowledge representation, inference and theorem proving using AI tools.
CO3	Demonstrate working knowledge of reasoning in the presence of incomplete and/or uncertain information.
CO4	Ability to apply knowledge representation, reasoning, and machine learning techniques to real-world problems

Course Content:

- Introduction to Python. [Quick Introduction of List, Tuple, Dictionary etc.]

1. Problem solving Agents using Python
    - Simple Reflex Agent
    - Table Driven Agent
    - Random Agent.
    - Goal-Based Agent
    - Utility Based Agent

2. Problem Spaces and blind search techniques by using Python.
    - Breadth First Search
    - Depth First Search
    - Uniform Cost Search
    - Depth Limit Search
    - Recursive Depth Limit Search
    - Iterative Deepening Search

3. Informed search techniques by using Python.
    - Greedy Best First Search
    - A* Search

4. Beyond Classical Search

    - Hill-Climbing Search Algorithm

5. Game playing by using Python.
6. Constraint satisfaction problems by using strawberry prolog.
7. Logic programming by using Python.

Teaching Methodology:

Project Application based lectures would be interactive, and it would cover the core concepts that are explained in the text and reference materials with adequate examples. Demo of tree graph search tutorials will have conceptual and numerical questions that would aid in strengthening the Artificial system principles.

Evaluation Scheme:

Component & Nature	**Duration**	**Marks / Weightage**
Practical Test – 1	2 hrs	15
Practical Test – 2	2 hrs	15
Lab Performance	---	10
Day to Day work	---	45
Attendance & Punctuality	----	15
Total		100

Text Books:

1. Allen B. Downey, Think Python, O'Reilly Media
2. Rich, Elaine Knight, Kevin, Artificial Intelligence, Tata McGraw Hill.
3. Luger, George F, Artificial Intelligence: Structures and Strategies for Complex Problem Solving, Pearson Education.

Reference Books:

1. Russell, Stuart J. Norvig, Peter, Artificial Intelligence: A Modern Approach, Pearson Education

Title of Course: COMPUTER NETWORKS LAB **Course Code: 18B17CI471**

L-T-P Scheme: 0-0-2 **Credits:** 1

Course Objectives

- To understand the working principle of various communication protocols.
- To analyze the various routing algorithms.
- To know the concept of data transfer between nodes.

Learning Outcomes:

Course Outcome	Description
CO1	Understand fundamental underlying principles of computer networking
CO2	Understand details and functionality of layered network architecture.
CO3	Apply mathematical foundations to solve computational problems in computer networking
CO4	Analyze performance of various communication protocols.
CO5	Compare routing algorithms
CO6	Practice packet /file transmission between nodes.

Course Content:

1. Identification of network hardware.
2. Fabrication of network cables and trouble shooting.
3. To study *stop &wait* and *sliding window* protocol.
4. To study MAC ALOHA protocol.
5. To study MAC CSMA and MAC CSMA/CD protocol.
6. To study TOKEN BUS and TOKEN RING.
7. To study ETHERNET.
8. To study TOKEN RING.
9. To study SWITCHED LAN.
10. To study Static routing.
11. To study dynamic routing.

Text Books:

1. Sybex CCNA Cisco Certified Network Associate Study Guide.5th Edition
2. Forouzen, Behrouz A.Fegan, Sophia Chung Data Communications and Networking TMH

Reference Books:

1. Carne, E. Bryan Professional's Guide to Data Communication in a CP/IP World Artech House, London, 2004

Title of Course: Mobile and Application Development Lab **Course Code:** 18B28CI408

L-T-P scheme: 1-0-2 **Credit: 3**

Prerequisite: Students must have already registered for the course, "***Introduction to Computers and Programming***" and "**Object Oriented Programming**".

Objective:

1. To learn and be able to implement different mobile-technologies.
2. To develop the abilities to call oneself mobile application developer.

Learning Outcomes:

At the end of the course, a student will:

1. Get familiar with different approaches to mobile application development.
2. Get to learn about application marketing.
3. Have a good grounding of mobile application development requirements, models and IDEs.
4. Possess demonstrative skills in building native applications.
5. Be able to design and develop cross-platform applications.
6. Learn to work in a team on a project.

Course Content:

Part-1: Orientation and Fundamentals of Development

Unit-1 Mobile applications and different approaches to mobile application development. Java features and review of Object Oriented Programming fundamentals.

Part-2: Android Studio and Basic Development Skills

Unit-2 Installing and getting accustomed to the android studio environment. Using activities and views. Working on different views like TextViews, ImageViews etc. Creating simple applications using basic view types.

Unit-3 Using animations, audio and video. Advanced android features like list views, Exception handling, Timers in androids, Advanced String manipulations.

Part-3: Serious Development

Unit-4 Maps and GeoLocation, Storing data permanently, Alert dialogs, SQLite databases, Advanced SQLite, Webviews.

Unit-5 Submitting app to distribution channels, marketing mobile app, Mobile App development models.

Part-4: Working in a team and Cross Platform Development

Unit-6 Using Git, Common Git commands, Project Development, Cross Platform Development using Flutter, Coding using Dart, MVC design pattern, Networking, Data storage, Authentication, State Management.

Teaching Methodology:

This course is introduced to help students transition from a regular developer to a mobile app developer. Starting from the basics, the student will slowly progress to become to other aspects of development including database, version control and other essential technologies that are helpful for a developer. The entire course is broken down into four separate parts: Orientation and Fundamentals of Development, Android Studio and Basic Development Skills, Serious Development, and Working in a team and Cross Platform Development. Each section includes multiple technologies to help a student gain more experience as a developer. This lab course is well complemented by a lecture in the same semester that helps a student learn and discuss the technical details of the underlying technologies.

Evaluation Scheme:

Exams		**Marks**	**Coverage**
P-1		15 Marks	**Based on Lab Exercises: 1-7**
P-2		15 Marks	**Based on Lab Exercises: 8-14**
Day-to-Day Work	Viva	20 Marks	**70 Marks**
	Demonstration	20 Marks	
	Lab Record	15 Marks	
	Attendance & Discipline	15 Marks	
Total		**100 Marks**	

Learning Resources:
Tutorials and lecture slides on Mobile Development (will be added from time to time): Digital copy will be available on the JUET server.

Books:
Text Book

i. Hello, Android (3rd edition): Introducing Google's Mobile Development Platform by Ed Burnette ISBN: 978-1-93435-656-2

ii. Android Programming for Beginners: Build in-depth, full-featured Android 9 Pie apps starting from zero programming experience, 2nd Edition by John Horton ISBN: 978-1789538502

iii. Head First Android Development: A Brain-Friendly Guide 1st Edition by Dawn Griffiths, David Griffiths. ISBN: 978-1449362188

Reference Books

1. Android Programming: The Big Nerd Ranch Guide (3rd Edition) (Big Nerd Ranch Guides) 3rd Edition by Bill Phillips, Chris Stewart, Kristin Marsicano ISBN: 978-0134706054
2. The Busy Coder's Guide to Android Development Version 8.0 by Mark M Murphy (O nline Book)

Web References:

1. https://developer.android.com
2. https://www.androidauthority.com
3. https://www.vogella.com

Journals:

1. International Journal of Interactive Mobile Technologies (iJIM)
2. ACM Transactions on the Information Systems (TOIC).
3. International Journal of Modern Computer Science (IJMCS)
4. ACM Transactions on Internet Technology (TOIT).

Title: Life Skills Code: **18B11HS411**

L-T-P scheme: 2**-0-0** **Credit: 2**

Prerequisites: None

Objective:

1. To employ positive behavior management techniques and to develop skills to manage their own behavior effectively
2. To develop one's personality by being aware of the self, connecting with others, reflecting on the abstract and the concrete.
3. To enhance the employability and maximize the potential of the students by introducing them to the principles that underlying personal and professional success, and help them acquire the skills needed to apply these principles in their lives and careers.

Learning Outcomes:

CO 1	Outline different life skills required in personal and professional life.
CO2	Describe the application of different theoretical perspectives within the field of motivation and applying these motivation theories to everyday settings (e.g., business, social interactions, education)
CO3	Develop the understanding of personality and shaping behavior through personality
CO4	Identify the basic mechanics of perception by demonstrating these through presentations.
CO5	Apply well-defined techniques to cope with emotions and stress and develop an awareness of the self.
CO6	Understand the basics of leadership and Learning

Course Content:

Unit-1: Overview of Life Skills: Meaning and significance of life skills, Life skills identified by WHO: Self-awareness, Empathy, Critical thinking, Creative thinking, Decision making, problem solving, Effective communication, interpersonal relationship, coping with stress, coping with emotion.

Unit-2: Motivation: Morale and Morale Building, Need and Importance of motivation, Process and types of motivation, Theories of motivation, Essentials of Good Motivation system

Unit-3: Overview of Personality concept and types, Personality traits, Factors that help in shaping personality, Theories of personality, Measurement of personality

Unit-4: Perception: - Factors affecting perception, Perceptual mechanisms Perceptual errors and distortions, Behavioral applications of perceptions
Unit-5: Self Awareness, Coping with emotions: Identifying and managing emotions, harmful ways of dealing with emotions, Stress Management: Stress, reasons and effects, identifying stress, Managing Stress
Unit-6: Conflict Management –sources, process and resolution of conflict
Unit-7: Leadership: Need for Leadership, Models of leadership development, and Characteristics of a good leader.
Unit-8: Learning: Concepts and Theories, classical conditioning, operant conditioning, Biological influences, Cognitive influences, Social learning theory, Behavioral modification theory

Teaching Methodology:

Life skills are those competencies that provide the means for an individual to be resourceful and positive while taking on life's vicissitudes .This course will equip students with the social and interpersonal skills that enable them to cope with the demands of everyday life. There will be a particular focus on social-cognitive processes and how situational factors trigger various emotions and corresponding motives that can then drive behavior. The main objectives of this course is to build self-confidence, encourage critical thinking, foster independence and help students to communicate more effectively

Evaluation Scheme:

Exams	**Marks**	**Coverage**
Test-1	15 Marks	Based on Unit-1, Unit-2 & Unit-3
Test-2	25 Marks	Based on Unit-4 & Unit-5 and around 30% from coverage of Test-1
Test-3	35 Marks	Based on Unit-6, Unit-7 & Unit-8 and around 30% from coverage of Test-2
Assignment	10 Marks	
Tutorials	5 Marks	
Quiz	5 Marks	
Attendance	5 Marks	
Total	**100 Marks**	

Learning Resources:

Case studies, video lectures and lecture slides on Life Skills (will be added from time to time): Digital copy will be available on the JUET server.

Text Book:

1. "Effective Communication and Soft Skills"; Nitin Bhatnagar, Pearson Education India,1e, 2011
2. "Personality Development and Soft Skills"; Barun Mitra, Oxford Higher Education,2016
3. "Sizzling Soft Skills for Spectacular Success"; P. Ameer Ali, Notion Press, 2017
4. "Organizational Behavior"; Stephen P. Robbins, Timothy A. Judge, Neharika Vohra, Pearson Education India, 16e, 2016
5. "Managing Organisations"; Rachna Chaturvedi, Vikas Publications, 2013

Reference Books/Material:

1. "The Power of Your Subconscious Mind"; Joseph Murphy, General press, 2015
2. "The Life-Changing Magic of Tidying Up: The Japanese Art of De cluttering and Organizing"; Marie Kondō, 1e,Ten speed Press, 2011
3. "The Power of Habit: Why We Do What We Do in Life and Business"; Charles Duhigg, Random House, 2012

5th Semester

Title: Probability Theory and Random Processes **Code: 18B11MA511**

L-T-P scheme: 3-1-0 **Credit: 4**

Prerequisite: Students must have already studied course, "Mathematics-I" and should have the Knowledge of Differential & Integral Calculus.

Objective: Objective of this course is to provide a foundation in the theory and applications of probability and stochastic processes and an understanding of the mathematical techniques relating to random processes in the areas of signal processing, detection, estimation, and communication. Topics include the axioms of probability, random variables, and distribution functions; functions and sequences of random variables; stochastic processes; and representations of random processes.

Learning Outcomes:

Course Outcome	**Description**
CO1	Construct sample spaces of random experiments; identify and specify events, and perform set operations on events; compute probabilities by counting; evaluate conditional probability, and apply Bayes' theorem to simple situations.
CO2	Express random variables by using CDFs, PMFs; calculate moments related to random variables; understand the concept of inequalities and probabilistic limits. Understand the axiomatic approach of probability theory and intrinsic need of (functions of) random variables for the analysis of random phenomena.
CO3	Compute probability distributions and correlation measures of bivariate random variables; obtain marginal and conditional distributions of random variables; find probabilities for outcomes of various events related to an uncertain phenomenon using appropriate probability distributions as models.
CO4	Conduct hypotheses tests concerning population parameters based on sample data; perform and interpret chi-square test of goodness-of-fit and test of independence; find the equation of regression line and second degree curve, and to predict the value of one variable based on the value of the other variable.
CO5	Identify and classify random processes and determine covariance and spectral density of stationary and ergodic random processes; demonstrate specific applications to Gaussian process.
CO6	Students are able to provide the theories associated with the random variable and random process. The course particularly provides the student with an ability to apply to real-world problems in the communication and physical systems.

Course Contents:

Unit-1: Random experiments, sample space and events. Three basic approaches to probability, conditional probability, total probability theorem, Bayes' theorem of Probability of causes, Bayes' theorem of future events, total independence, mutual independence and pair wise independence.

Unit-2: One dimensional random variables(discrete and continuous) and their distributions, bivariate distributions, joint, marginal and conditional distributions, characteristic function.

Unit-3: Covariance and correlation of random variables. Some special probability distributions: Binomial, Poisson, probability distributions. Negative Binomial,Geometric and Normal probability distributions. Fitting of probability distributions.

Unit-4: Concept of reliability: Reliability function, Hazard rate function, Mean time to failure, cumulative and average failure rate, Conditional reliability and failure rates, residual MTTF, some special failure rate distributions- exponential distribution and the Weibull distribution, reliability of systems- series configuration and some deductions, parallel-series configuration, series -parallel configuration.

Unit-5: Introduction and description of random processes, average values of random processes, stationary processes and computation of their averages, autocorrelation function and its properties, Cross correlation and its properties. Power spectral density function and its properties. Ergodicity of a random process, Poisson processes.

Teaching Methodology:

The course will be covered through lectures supported by tutorials. Apart from the discussions on the topics covered in the lectures assignments/ quizzes in the form of questions will also be given.

Evaluation Scheme:

Exams	**Marks**	**Coverage**
Test-1	15 Marks	Based on Unit-1
Test-2	25 Marks	Based on Unit-2 & Unit-3 and around 30% from coverage of Test-1
Test-3	35 Marks	Based on Unit-4 to Unit-5 and around 30% from coverage of Test-2
Assignment	10 Marks	
Tutorials	5 Marks	

Quiz	5 Marks	
Attendance	5 Marks	
Total	**100 Marks**	

Learning Resources:

Tutorials and lecture slides on Probability Theory and Random Processes (will be added from time to time): Digital copy will be available on the JUET server.

Text books:

1. T. Veerarajan ,Probability, Statistics and Random Processes, Tata McGraw Hill.
2. J.J. Aunon & V. Chandrasekhar, Introduction to Probability and Random Processes, Mc- Graw Hill International Ed.
3. A. Papoulis & S.U. Pillai, Probability, Random Varibles and Stochastic Processes, Mc-Graw Hill.
4. H. Stark, and J.M. Woods, Probability and Random Processes with Applications to Signal Processing, Pearson Education.

Title: Theory of Computation **Code:** 18B11C511

L-T-P scheme: 3-1-0 **Credit: 4**

Prerequisite:

Students must have already studied for the course Set algebra, elementary formal logic, constructing proofs, recurrence relations.

Objective:

1. To give an overview of the theoretical foundations of computer science from the perspective of formal languages
2. To illustrate finite state machines to solve problems in computing
3. To explain the hierarchy of problems arising in the computer sciences.
4. To familiarize Regular grammars, context frees grammar.

Learning Outcomes:

Course Outcome	**Description**
CO1	Students will demonstrate knowledge of basic mathematical models of computation and describe how they relate to formal languages.
CO2	To Design Finite Automata's for different Regular Expressions and Languages
CO3	To Construct grammar for various languages and applying normal forms and push down automata
CO4	To solve various problems of Turing Machines and types of TM

Course Content:

UNIT – I

Mathematical Concepts: Review definitions and notations for sets, relations and functions. Basic concepts and definitions Set operations; partition of a set, Equivalence relations;

Properties on relation on set; Proving Equivalences about Sets. Central concepts of Automata Theory.

UNIT – II

FINITE AUTOMATA (FA): Introduction, Deterministic Finite Automata (DFA) - Formal definition, simpler notations (state transition diagram, transition table), language of a DFA. Nondeterministic Finite Automata (NFA)- Definition of NFA, language of an NFA, Equivalence of Deterministic and Nondeterministic Finite Automata, Applications of Finite Automata, Finite Automata with Epsilon Transitions, Eliminating Epsilon transitions, Minimization of Deterministic Finite Automata, Finite automata with output (Moore and Mealy machines) and Inter conversion.

UNIT - III

REGULAR EXPRESSIONS (RE): Introduction, Identities of Regular Expressions, Finite Automata and Regular Expressions- Converting from DFA's to Regular Expressions, Converting Regular Expressions to Automata, applications of Regular Expressions.

REGULAR GRAMMARS: Definition, regular grammars and FA, FA for regular grammar, Regular grammar for FA. Proving languages to be non-regular -Pumping lemma, applications, Closure properties of regular languages.

UNIT - IV

CONTEXT FREE GRAMMER (CFG): Derivation Trees, Sentential Forms, Rightmost and Leftmost derivations of Strings. Ambiguity in CFG's, Minimization of CFG's, CNF, GNF, Pumping Lemma for CFL's, Enumeration of Properties of CFL (Proof's omitted).

UNIT – V

PUSHDOWN AUTOMATA: Definition, Model, Acceptance of CFL, Acceptance by Final State and Acceptance by Empty stack and its Equivalence, Equivalence of CFG and PDA.

UNIT VI

TURING MACHINES (TM): Formal definition and behaviour, Languages of a TM, TM as accepters,

and TM as a computer of integer functions, Types of TMs.

RECURSIVE AND RECURSIVELY ENUMERABLE LANGUAGES (REL): Properties of recursive and recursively enumerable languages, Universal Turing machine, The Halting problem, Undecidable problems about TMs. Context sensitive language and linear bounded automata (LBA), Chomsky hierarchy, Decidability.

Teaching Methodology:

Teaching in this course is designed to engage the students in active and experimental learning by taking a problem solving and design oriented approach with special emphasis on real world applications. Students are expected to carry out lot of design and programming.

Evaluation Scheme:

Exams	**Marks**	**Coverage**
Test-1	15 Marks	Based on Unit-1, Unit-2
Test-2	25 Marks	Based on Unit-3 & Unit-4 and around 30% from coverage of Test-1
Test-3	35 Marks	Based on Unit-5 to Unit-6 and around 30% from coverage of Test-2
Assignment	10 Marks	
Tutorials	5 Marks	
Quiz	5 Marks	
Attendance	5 Marks	
Total	**100 Marks**	

Learning Resources:

Tutorials and lecture slides on Theory of Computation (will be added from time to time): Digital copy will be available on the JUET server.

Text Books:

1. K. L. P Mishra, N. Chandrashekaran (2003), Theory of Computer Science-Automata Languages and Computation, 2nd edition, Prentice Hall of India, India.
2. John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman (2007), Introduction to Automata Theory Languages and Computation, 3rdedition, Pearson Education, India.

Reference Books:

1. Papadimitriou, Elements of the Theory of Computation, Prentice-Hall, 1998
2. Peter Dehning, Jack B. Dennis, “Machines, Languages and Computation”, Second Edition, Prentice-Hall, 1978
3. Harry R. Lewis, Christos H. Papadimitriou, "Elements of the theory of computation", Second Edition, Prentice-Hall, 1998

Title: Open Source Software Lab **Code: 18B17CI507**

L-T-P scheme: 0**-0-2** **Credit: 1**

This Lab will be based on the subject run in CSE-Elective -1

6th Semester

Title of Course: Software Engineering **Course Code 18B11CI612**
L-T-P Scheme: 3-0-0 **Credits: 3**

Pre-requisite: Good Knowledge of Computer Programming

Post Course:
Object Oriented Software Engineering, Software Quality Management Objective: To engineer good quality software from its specification.

Learning Outcomes

Software Engineering	
Course Outcome	**Description**
CO1	Outline various software models with respect to their needs of the customer requirement and concepts of some modeling language.
CO2	Describe the real world problems using software engineering concepts and tools.
CO3	Develop the software design to meet customer expectations using modeling language.
CO4	Identify and use various cost estimation techniques used in software engineering project management.
CO5	Apply verification and validation techniques on a given software project.
CO6	Demonstrate deployment and basic maintenance skills.

Course Outline:
Interactive Systems, Usability, Introduction to software engineering, Software process models, PSP, TSP Requirement Engineering: Requirement Elicitation, Analysis, Specification, SRS, Formal system development techniques, Analysis and Modeling: Data modeling, Functional modeling, Software Architecture and Design: Data design, Architectural Design Process, SADT, OOAD, function-oriented design

UML: Use case diagram, State diagram, Activity Diagram, Class Diagram, Sequence diagram, Collaboration diagram, Deployment Diagram, Event trace diagram, Design Patterns: Structural Patterns, Behavioral Patterns, Creational Patterns

Software Estimation- Estimating Size, Effort and Cost: Metric for Analysis, Metric for Design, COCOMO model, Putnam Model etc., Implementation and Integration: Coding standard and practices, Top-Down and Bottom–up Approach, Verification and Validation,

Software Testing: Structural testing, functional Testing, Testing Strategies, Test Case design.

Software Maintenance: Types, Cost of Software, maintenance, Software Maintenance Models

CASE Tool Taxonomy: Business Process Engineering tool, Process modeling and management tool, project planning tool, requirement tracking tool, Metric and management tool, documentation tool, system software tool etc.Introduction to software engineering for web and mobile applications.

Teaching Methodology:
This course should be conducted in a highly interactive environment. Students will work on different software projects in small groups. Exercises shall almost exclusively consist of design work and the laboratory shall be a place to develop these designs using CASE tools. As part of lab work there shall be a project to build a specification and convert it into working software using Rational Unified Process. Also, there shall be a testing project. There is a self learning component that shall be announced.

Evaluation Scheme:

Exams	**Marks**	**Coverage**
Test-1	15 Marks	Based on Unit-1 & Unit-2
Test-2	25 Marks	Based on Unit-3 to Unit-4 and around 20% from coverage of Test-1
Test-3	35 Marks	Based on Unit-5 to Unit-6 and around 40% from coverage of Test-2
Assignment	10 Marks	
Tutorials	5 Marks	
Quiz	5 Marks	
Attendance	5 Marks	
Total	**100 Marks**	

Text Book

1. The Unified Modeling Language Users Guide: Grady Booch, James Rambaugh, Ivar Jacobson, Addision Wesley.
2. Douglas Bell, “Software Engineering for students: a programming approach”, 4th Ed Pearson Education, 2005.
3. Dines, Bjorna “Software Engineering: abstraction and Modelling”Vol.1, 2006, Springer Verlag Berlin Heidelberg (206).
4. Cooling Jin, “Software Engineering for real time systems, Addison Wesley.
5. Khoshgoftaar, Taghi M. “Software Engineering with Computational Intelligence”.
6. Sommerville, Ian, “Software Engineering”, 8th Edition, Pearson Education Ltd.
7. Pressman S. Roger, “Software Engineering: A practitioner's Approach”, 7th Edition, McGraw Hill.

Title of Course: Software Engineering Lab **Course Code: 18B17CI672**
L-T-P scheme: 0-0-2 **Course Credit: 1**

Prerequisite: Students must have already registered for the course, "Software Engineering".

Objectives: Students will be capable to acquire the generic software development skill through various stages of software life cycle. He will also be able to ensure the quality of software through software development with various protocol based environment.

Software Engineering Lab	
Course Outcome	Description
CO1	Outline various software models with respect to their needs of the customer requirement and concepts of some modelling language.
CO2	Describe the real world problems using software engineering concepts and tools.
CO3	Develop the software design to meet customer expectations using modelling language.
CO4	Identify and use various cost estimation techniques used in software engineering project management.
CO5	Apply verification and validation techniques on a given software project.
CO6	Demonstrate deployment and basic maintenance skills.

Course Contents:

Unit I- Introduction to software engineering: Code comprehension.

Unit II- Requirement engineering: Requirement Elicitation, specification, IEEE standard template for SRS, Requirement Engineering tools.

Unit III- UML Modeling: Use case diagram , State diagram, Activity Diagram, Class Diagram, Sequence diagram, Collaboration diagram, Deployment Diagram, Component Diagram, Event trace diagram , c++ code generation, Introduction to Sec UML.

Unit IV- Software Metrics: Product, process and project metrics.

Unit V- Software Testing: Structural testing, functional Testing, Testing Strategies and Tactics, Test Case design.

Text Books

1. Software Engineering: A practitioner's approach: Roger S. Pressman, McGraw-Hill Publications (Sixth Edition)
2. The Unified Modeling Language Users Guide: Grady Booch, James Rambaugh, Ivar Jacobson, Addision Wesley

References

1. Modern Structured Analysis: Edward Yourdon , PHI Publications

Title of Course: Advanced Programming Lab-3 **Course Code: 18B17CI673**

L-T-P scheme: 0-0-2 **Credit: 3**

Prerequisite: Students must have already registered for the course, "***Introduction to Computers and Programming***" and "**Object Oriented Programming**".

Objective:

1. To learn and be able to implement the front-end and back-end web-technologies.
2. To develop the abilities to call oneself full-stack web developer.

Learning Outcomes:

Course Outcome	**Description**
CO1	Get familiar with processes of Full Stack Web Development.
CO2	Have a good grounding of Web Application Terminologies, Internet tools and languages like HTML5 and CSS.
CO3	Possess demonstrative skills in using and applying JavaScript.
CO4	Build modern, fast and scalable server-side web applications with NodeJS and databases like SQL or MongoDB.
CO5	Apply web engineering approaches required to create web applications
CO6	Work as a team on a project.

Course Content:

Part-1: Fundamentals of Full Stack Web Development

Unit-1 Creating first web-application, hosting a web application, creating websites, authoring tools,
domain names.

Part-2: Front End Tools & Technologies

Unit-2 Markup and Styling: HTML, Cascading Style Sheets, using Bootstrap.

Unit-3 JavaScript Fundamentals: Language Features, JSON, Ajax, jQuery, Popular Frameworks like React, Angular JS.

Part-3: Back End Tools & Technologies

Unit-4 Web Programming through Node.js and/or Java. Node.js Modules, NPM, Events, Upload File, Email, Get/Post methods, Java Servlets vs. JSP, Request vs. Response objects, other Java objects and features.

Unit-5 Databases and Web Storage: Designing and creating databases, database connection through back end programming languages, Web storage to store sessions, cookies, and cached data in the browser.

Part-4: Project Development

Unit-6 Using Git, Common Git commands, Project Development.

Teaching Methodology:

This course is introduced to help students transition from a simple developer to a full stack developer. Starting from frontend development, the student will slowly progress to become to other aspects of development including backend, database, version control and other essential technologies that are helpful for a developer. The entire course is broken down into four separate parts: Fundamentals of Web Development, Front End tools & Technologies, Back End Tools & Technologies, and Project Development. Each section includes multiple technologies to help a student gain more experience as a developer. This lab course is well complemented by a theory course under the name Web Development in the same semester that helps a student learn and discuss the technical details of the underlying technologies.

Evaluation Scheme:

Exams		**Marks**	**Coverage**
P-1		15 Marks	**Based on Lab Exercises: 1-7**
P-2		15 Marks	**Based on Lab Exercises: 8-14**
Day-to-Day Work	Viva	20 Marks	**70 Marks**
	Demonstration	20 Marks	
	Lab Record	15 Marks	
	Attendance & Discipline	15 Marks	
Total		**100 Marks**	

Learning Resources:

Tutorials and lecture slides on Web Technology Lab (will be added from time to time): Digital copy will be available on the JUET server.

Books:

Text Book

1. Web Technologies: Achyut Godbole, Atul Kahate, McGraw-Hill Education (Third Edition).
2. Web Engineering: A Practitioner's Approach by Roger Pressman and David Lowe, McGraw-Hill, 2009.
3. HTML and CSS: Comprehensive 7th edition, by by Denise M. Woods and William J. Dorin. Publisher: Cengage Learning; (2012) ISBN-10:1133526144
4. Internet & World Wide Web How to Program, 5/e Paul J. Deitel, Harvey M. Deitel, Abbey Deitel, Pearson Education 2012.

Reference Books

1. Internet & World Wide Web How to Program / Deitel, H.M.
2. Web Design with HTML/Flash/Java Script and E-Commerce Bible / Crowder, David
3. Database Driven Web Sites / Feiler, Jesse
4. Web design: the complete reference / Powell Thomas A
5. Internet 101: a beginner's guide to the Internet and the WorldWideWeb/Lehnert Wendy G
6. E-Commerce: Fundamentals and Applications / Chan, Henry
7. E-commerce: strategy, technology & applications / Whiteley, David
8. E-Commerce Logistics & Fulfillment: delivering the goods / Bayles, Deborah L.

Web References:

1. www.w3schools.com
2. http://www.techtutorials.info/ecommerce.html

Journals:

1. ACM Transactions on the Web (TWEB).
2. ACM Transactions on the Information Systems (TOIC).
3. ACM Transactions on Graphics (TOG).
4. ACM Transactions on Internet Technology (TOIT).

Science Elective

Course Title: Data Science **Course Code: 18B14MA541**
L-T-P Scheme: 3-0-0 **Credit: 3**

Prerequisite: Students must have already studied the course "Business Analysis Techniques"

Course Objectives:

This course will introduce students to this rapidly growing field and equip them with some of its basic principles and tools as well as its general mindset.Students will learn concepts, techniques and tools they need to deal with various facets of data
Science practice, including data collection and integration, exploratory data analysis, predictivemodelling, descriptive modelling, data product creation, evaluation, and effective communication.

Learning Outcomes:

At the end of the course students should be able to:
Students will develop relevant programming abilities.
Students will demonstrate proficiency with statistical analysis of data.
Students will develop the ability to build and assess data-based models.
Students will execute statistical analyses with professional statistical software.
Students will demonstrate skill in data management.
Students will apply data science concepts and methods to solve problems in real-world contexts and will communicate these solutions effectively

Course Content:

Unit I:Introductionand Data Pre-processing

Data Science Introduction

Big Data and Data Science

Current landscape of perspectives

Unit II: Data Analysis and Correlations: Basic Concepts and Methods

Populations and samples

Statistical modelling, probability distributions, Regression,fitting a model

Dimensionality Reduction: PCA & DWT, Correlation and regression analysis.

Chisquare t and F distributions (definitions only) Confidence interval Single mean and difference known and unknown variances.

UnitIII: Introduction to machine learning and Cluster Analysis : Basic Concept and Methods

Supervised and unsupervised learning, Training and testing data, Over fitting and under fitting. Distance measures :- Manhattan, Chebbychev, Mahalanobis Distance
Cluster Analysis, Partitioning Methods, Hierarchical Methods, Density-Based Methods,

Grid-Based Methods, Evaluation of Clustering, Clustering High-Dimensional Data, Clustering Graph and Network Data

UnitIV:ClassificationAlgorithms

Basic Concepts, Decision Tree Induction, Bayes Classification Methods, Rule-Based Classification, Model Evaluation and Selection, Techniques to Improve Classification Accuracy, Support Vector Machines, Lazy Learners (or Learning from Your Neighbors)

UnitV: Introduction to Web Search and Social Media Analytics

Data Wrangling: APIs and other tools for scrapping the Web

Mining Complex Data Types, Other Methodologies of Data, Mining, Data Mining Applications, Data Mining and Society, Data Mining Trends

Social Media Analytics is the science of analyzing data to convert information to useful knowledge. This knowledge could help us understand our world better and, in many contexts, enable us to make better decisions.

Evaluation Scheme:

Exams	**Marks**	**Coverage**
Test-1	15 Marks	Based on Unit-1, Unit-2
Test-2	25 Marks	Based on Unit-3 & Unit-4 and around 30% from coverage of Test-1
Test-3	35 Marks	Based on Unit-5 around 30% from coverage of Test-2
Assignment	10 Marks	
Tutorials	5 Marks	
Quiz	5 Marks	
Attendance	5 Marks	
Total	**100 Marks**	

Text Books:

1. Cathy O'Neil and Rachel Schutt. Doing Data Science, Straight Talk fromthe Frontline. O'Reilly. 2014.
2. Avrim Blum, John Hopcroft and Ravindran Kannan. Foundations of Data Science.

Refernce Books:

1. Jure Leskovek, Anand Rajaraman and Je_rey Ullman. Mining of Massive Datasets. v2.1,
2. Cambridge University Press. 2014.
3. Kevin P. Murphy. Machine Learning: A Probabilistic Perspective. ISBN 0262018020. 2013.
4. Foster Provost and Tom Fawcett. Data Science for Business: What You Need to Know aboutData Mining and Data-analytic Thinking. ISBN 1449361323. 2013.
5. Trevor Hastie, Robert Tibshirani and Jerome Friedman. Elements of Statistical Learning,
6. Second Edition. ISBN 0387952845. 2009.

CSE Elective-I

Title: Compiler Design **Code: 18B14CI541**

L-T-P scheme: 3-0-0 **Credit: 3**

Prerequisite:

Students must have already registered for the course, "Data Structures" and "Theory of Computation ".

Objective:

- Deepen the understanding of compiler design
- Develop problem solving ability using programming
- Develop ability to design and analyze a compiler

Learning Outcomes:

Course Outcome	**Description**
CO1	Specify and analyse the lexical, syntactic and semantic structures of advanced language features.
CO2	Separate the lexical, syntactic and semantic analysis into meaningful phases for a compiler to undertake language translation
CO3	Write a scanner, parser, and semantic analyser without the aid of automatic generators
CO4	Turn fully processed source code for a novel language into machine code for a novel computer
CO5	Describe techniques for intermediate code and machine code optimisation
CO6	Design the structures and support required for compiling advanced language features.

Course Content:

UNIT I: INTRODUCTION TO COMPILERS

Translators-Compilation and Interpretation, Language processors, The Phases of Compiler, Errors Encountered in Different Phases,The Grouping of Phases of Compiler , Programming Language basics.

UNIT II: LEXICAL ANALYSIS

Need and Role of Lexical Analyzer,Lexical Errors,Expressing Tokens by Regular Expressions, Converting Regular Expression to DFA ,Minimization of DFA, Language for Specifying Lexical Analyzers, LEX (Design of Lexical Analyzer for a sample Language).

UNIT III: SYNTAX ANALYSIS

Need and Role of the Parser, Context Free Grammars , Top Down Parsing ,General Strategies, Recursive Descent Parser ,Predictive Parser ,LL(1) Parser,Shift Reduce Parser-LR Parser, LR (0)Item, Construction of SLR Parsing Table , Introduction to LALR Parser ,Error Handling and Recovery in Syntax Analyzer, YACC (Design of a syntax Analyzer for a Sample Language) .

UNIT IV: SYNTAX DIRECTED TRANSLATION & RUN TIME ENVIRONMENT

Syntax directed Definitions, Construction of Syntax Tree, Bottom-up Evaluation of S-Attribute Definitions, Design of predictive translator, Type Systems, Specification of a simple type checker, Equivalence of Type Expressions, Type Conversions. RUN-TIME ENVIRONMENT: Source Language Issues, Storage Organization-Storage Allocation, Parameter Passing, Symbol Tables.

UNIT V: CODE OPTIMIZATION

Principal Sources of Optimization, DAG, Optimization of Basic Blocks, Global Data Flow Analysis, Efficient Data Flow Algorithms.

UNIT VI: CODE GENERATION

Issues in Design of a Code Generator, a Simple Code Generator Algorithm.

Teaching Methodology:

Teaching in this course is designed to engage the students in active and experimental learning by taking a problem solving and design oriented approach with special emphasis on real world applications. Students are expected to carry out lot of design and programming.

Evaluation Scheme:

Exams	Marks	Coverage
Test-1	15 Marks	Based on Unit-1, Unit-2
Test-2	25 Marks	Based on Unit-3 & Unit-4 and around 30% from coverage of Test-1
Test-3	35 Marks	Based on Unit-5 to Unit-6 and around 30% from coverage of Test-2
Assignment	10 Marks	
Tutorials	5 Marks	
Quiz	5 Marks	
Attendance	5 Marks	
Total	**100 Marks**	

Learning Resources:

Tutorials and lecture slides on Compiler Design (will be added from time to time): Digital copy will be available on the JUET server.

Text Books:

1. Compilers : Principles, Techniques and Tools, Aho, Sethi and Ullman, Pearson Education
2. Principles Of Compiler Design by Alfred V Aho and Ullman, Narosa Publication

Reference Books:

1) Compiler Design in C, Holub, Prentice Hall of India

2) Advanced Compiler Design and Implementation, Muchnick Steven, Morgan Kauffman Publishers

3) Compiler Design, Santanu Chattopadhyay, PHI

4) Compiler Construction Principles and Practice, Kenneth C. Louden, Thomson

5) Compiler Construction and Design, Rajni Jindal , Umesh Publications

CSE Elective-4

Title of Course: Software Engineering Management **Course code: 18B14CI750**

L-T Scheme: 3 **Course Credits: 3**

Pre-requisites
Students must have knowledge of "Software Engineering"

Objectives

1. To strengthen their ability to apply Software Engineering Principles and practices to manage individuals and teams in software projects.
2. To strengthen their skills in Requirements engineering, Configuration management, quality management, applying design patterns and software testing techniques.
3. To provide experience in the use of project management planning tools.

Learning Outcomes
Student will be able to:

Course Outcome	Description
CO1	Manage and Plan Team based Projects.
CO2	Elicit, document and validate requirements for projects.
CO3	Learn and apply Design pattern concepts in developing applications of varying complexities.
CO4	To achieve good quality software.
CO5	To ensure the delivery of the system is on time and within budget.
CO6	Develop test cases and automate software testing.

Course content:

1. **Introduction to Software Engineering & Project Management**
   Modeling Processes and Life-Cycle , Software process models , Process iteration , Process activities , Cost estimation, Project scheduling, Staffing,
2. **Software Configuration management**
   Base line, Software Configuration Items, The SCM Process, Version Control , Change Control , Configuration Audit , Status Reporting , SCM Standards
3. **Software Quality Management**
   Quality concepts, Quality Assurance, Quality Planning, Quality control, Software measurement And metrics.
4. **Software Reengineering and Maintenance:**

   Reverse engineering, Forward engineering, Restructuring, Reengineering Process Model.
5. **Risk Management**

Risk strategies, Reactive & Proactive Risk strategies, Software Risk, Risk Identification, Risk projection, Risk Assessment, Risk Refinement, Risk Mitigation, monitoring and Management.

6. **AGILE**

   Agile development, Classification of methods, Agile principals, Agile project management, SCRUM, XP, EVO and UP Method overview, Life cycle, Work product, role and services, Common mistakes and misunderstandings, Process mixture, Adoption strategy.

7. Software Reuse

   Introduction of software reuse, Basic Issues in any Reuse Program,Reuse Approach,Reuse at Organization Level, Introduction of Reusable Component, COTS,Component Adaptation Technoques.

Teaching Methodology

Course will be delivered through lecture sessions and assignments. Course will emphasize more on Mini-Projects. Students will apply advance concepts of Requirements Engineering, Agile methods, Design patterns, RMMM in their mini-projects. They will design test cases for their problem domain and also work on automated testing tools. Students will have to maintain a report on each mini-project. Research literature on topics mentioned in course outline will be studied and presented by the students.

Evaluation Scheme

Evaluations	**Marks**	**Remarks**
T-1	15 Marks(1-Hour)	
T-2	25 Marks (1 Hour 30 Min.)	
T-3	35 Marks (2- Hours)	
Tutorials and Case Study	5 Marks	
Assignments	10 Marks	
Quiz	5 Marks	
Attendance	5 Marks	
Total	**100 Marks**	

Reference Material

1. Agile and Iterative Development: A Manager's Guide, Craig Larman
2. Introduction to the Personal Software Process (SM), Watts Humphrey
3. Introduction to the Team Software Process(SM),Watts Humphrey

4. Software Engineering, R.S. Pressman, McGraw Hill
5. Software Engineering Project Management, by Richard Thayor, Forwarded By Edward Yourden
6. Software Testing Techniques, B. Beizer
7. Software Testing ,Louis Tamres
8. Aspect-Oriented Analysis and Design: The Theme Approach (The Addison-Wesley Object Technology Series)
9. Engineering and Managing Software Requirements, by Claes. Wohlin
10. Requirements Engineering, by Elizabeth. Hull, Ken. Jackson, Jeremy. Dick
11. User-Centered Requirements Engineering, by Alistair Sutcliffe

CSE Elective-6

Title of Course: Introduction to Machine Learning **Course Code: 18B11CI843**
L-T-P Scheme: 3-0-0 **Course Credit: 3**

Prerequisite: Students must have knowledge of statistical techniques

Objectives:
In this course we will study the basic component of an intelligence system i.e. machine learning, their functions, mechanisms, policies and techniques used in their implementation and examples.

Learning Outcomes:
The students will have

1. Detailed knowledge of the concepts of machine learning.
2. Hypotheses Generation
3. Classifications and Regression
4. Ensemble Learning Techniques
5. Unsupervised Learning Techniques
6. Various application of machine learning in AI and different fields.

Course Outcome	**Description**
CO1	List various approaches of Machine Learning.
CO2	Describe machine learning algorithms to solve the real world problems
CO3	Develop Hypothesis and machine learning models
CO4	Identify appropriate models for solving machine learning problems.
CO5	Apply learning techniques to solve real world machine learning problems
CO6	Evaluate and interpret the results of the algorithms.

Course Details:

1. Introduction:
What Is Machine Learning?, Why Use Machine Learning? ,Types of Machine Learning Systems, Supervised/Unsupervised Learning, Batch and Online Learning, Instance-Based Versus Model-Based Learning, Hypothesis generation, Main Challenges of Machine Learning, Data sets and Testing and Validating.

2. Concept Learning:
Introduction to Concept Learning, Concept Learning Task, Notation, Inductive Learning Hypotheses, Concept Learning as Search: Generic-to-Specific Ordering of Hypotheses, Finding a Maximally Specific Hypotheses, Version Spaces, Candidate-Elimination Algorithms.

3. Classification:
MNIST Training a Binary Classifier, Performance Measures, Measuring Accuracy Using Cross-Validation, Confusion Matrix, Precision and Recall Precision/Recall Tradeoff, The ROC Curve, Multiclass Classification, Error Analysis, Multi label and Multi output classification

4. Training Models:
Linear Regression, The Normal Equation, Computational Complexity, Gradient Descent, Polynomial Regression, Learning Curves, Regularized Linear Models, Logistic Regression, Estimating Probabilities, Training and Cost Function, and Decision Boundaries

5. Support Vector Machines
Linear SVM Classification, Soft Margin Classification, Nonlinear SVM Classification, Polynomial Kernel, Adding Similarity Features, Gaussian RBF Kernel, Computational Complexity, SVM Regression, Decision Function and Predictions, and The Dual Problem

6. Decision Trees
Training and Visualizing a Decision Tree, Making Predictions, Estimating Class Probabilities, The CART Training Algorithm, Computational Complexity, Gini Impurity or Entropy, Regularization of hyper parameters, and Random Forests

7. Dimensionality Reduction:
The Curse of Dimensionality, Main Approaches for Dimensionality Reduction, Projection, Manifold Learning, PCA, Preserving the Variance, Principal Components, Choosing the Right Number of Dimensions

8. Unsupervised Learning Techniques:
Clustering, K-Means, Limits of K-Means, Using clustering for image segmentation, Using Clustering for Pre-processing and for Semi-Supervised Learning

9. Introduction to Neural Networks:
From Biological to Artificial Neurons, Biological Neurons, Logical Computations with Neurons, The Perceptron, Multi-Layer Perceptron and Backpropagation

Reference Books:

1: Machine Learning, TOM M MITCHELL, TMH
2: Introduction to Machine Learning, 2nd Ed, Ethem Alpaydin, The MIT Press Cambridge, Massachusetts,London, England.
3. Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow, 2nd Ed, Aurelien Geron, O'RIELLY

CSE Elective-7

Title of Course: Software Quality Management **Course Code: 18B14CI850**
L-T-P Scheme: 3-0-0 **Course Credit: 3**

Pre-requisites: Students must have knowledge of "Software Engineering and Software Management"

Objectives: The course has the basic scope to provide the students with theoretical knowledge about concepts of software quality, about the quality- models, - standards and – methodologies used in the software industry. The theory is supported and supplemented by the lecturer's 10 years experience in software quality management. Understanding and usage of the theory are consolidated by the case studies and exercises.

Course Outcome	Description
CO1	List various principles Software Quality Management.
CO2	Describe the real world problems that may arise during software development and affects the quality.
CO3	Develop a appropriate plan for software quality management .
CO4	Explore key contributors / metrics for effective quality control
CO5	Identify appropriate international standard for real life software project for controlling and managing the quality of product.
CO6	Demonstrate and present the learning of course on real life problems.

Course Contents:

Introduction to Software Quality Engineering: what is software quality, who cares for software quality, benefits of software quality, phases in software development, views of quality, hierarchical models of quality**,** types of defects, cost of fixing defects, cost of poor quality, definitions used in software quality engineering, software quality assurance, quality control, software configuration management.

Software Testing: guiding principles of testing, composition of a testing team, skills of a tester, types of testing, evaluating the quality of test cases, techniques for reducing number of test cases, requirements for effective testing, test oracle, economics of software testing, handling defects, risk in software testing, requirement traceability matrix.

Metrics for Software Quality: categories of software metrics, metrics program, goal question metric method, types of metrics, commonly used software metrics, process metrics, product metrics, metrics for resources.

Tools for Quality Improvement: basic quality control tools, check sheet, cause and effect diagram, pareto diagram, histogram, scatter plot, run chart, control chart, orthogonal defect classification.

Software Quality Measurement: Measuring quality, software metrics, problems with metrics, an overall measure of software quality. Developments in Measuring Quality**:** The work of Gilb, the COQUAMO project.

The ISO9000 series of quality management standards: The purpose of standards, the ISO9000 series: a generic quality management standard, ISO9000-3: notes for guidance on the application of ISO9001 in software development, the impact of ISO9000 and TickIT. Models and standards for process improvement: The Capability Maturity Model, individual levels of CMM, the role of the CMM, SPICE modeling.

Text Book:

1. "Software Quality : Theory and Management" by Alan C Gillies, CENGAGE Learning, Second edition.
2. " Software Quality Assurance, Testing and Metrics" by Anirban Basu, PHI Publication.

References:

1. Agile and Iterative Development: A Manager's Guide, Craig Larman.
2. Practical Guide to Software Quality Management, John W. Horch.
3. Introduction to the Team Software Process(SM),Watts Humphrey.
4. Software Engineering, R.S. Pressman, McGraw Hill.