# B. Tech. (CSE) – 2024 Curriculum

| Course Code | Semester - I | Credit |
|---|---|---|
| HM101 | English for Communication | 2 |
| MA101 | Mathematics - I | 3 |
| PH101 | Physics | 3 |
| CS101 | Introduction to Computer Programming | 3 |
| ME101 | Basics of Mechanical Engineering | 2 |
| ME102 | Engineering Practice | 2 |
| PH102 | Physics Laboratory | 2 |
| CS102 | Introduction to Computer Programming Laboratory | 2 |
| | **Total Credits** | 19 |

| Course Code | Semester - II | Credit |
|---|---|---|
| MA151 | Mathematics - II | 3 |
| ME151 | Engineering Graphics | 3 |
| EC151 | Basics of Electronics Engineering | 3(2T+2L) |
| CS151 | Introduction to Python Programming | 3(2T+2L) |
| CS152 | Programme Core -I / Digital System Design | 3 |
| CS153 | Programme Core -II / Data Structures | 3 |
| SH151 | Energy and Environmental Science | 0 |
| CS154 | Digital Laboratory | 2 |
| CS155 | Data Structures Laboratory | 2 |
| | **Total Credits** | 22 |

| Course Code | Semester - III | Credit |
|---|---|---|
| CS201 | Programme Core -III / Discrete Structures | 4 |
| CS202 | Programme Core -IV / Design and Analysis of Algorithms | 3 |
| CS203 | Programme Core -V / Programming Paradigms | 3 |
| CS204 | Programme Core -VI / Computer Organisation and Architecture | 3 |
| CS205 | Programme Core -VII / Object Oriented Programming (C++) | 3 |
| CS206 | DAA laboratory | 2 |
| CS207 | Programming Paradigms laboratory | 2 |
| CS208 | Object Oriented Programming laboratory | 2 |
| | **Total Credits** | 22 |

| Course Code | Semester - IV | Credit |
|---|---|---|
| MA251 | Probability, Statistics and Queuing Theory | 4 |
| HM251 | Economics for Engineers | 3 |
| CS251 | Programme Core -VIII / Data Communication and Computer Networks | 3 |
| CS252 | Programme Core - IX / Operating Systems | 3 |
| CS253 | Programme Core - X / Automata and Formal Languages | 3 |
| CS254 | Programme Core - XI / Embedded Systems | 3 |
| CS255 | Operating Systems laboratory | 2 |
| CS256 | Embedded Systems laboratory | 2 |
| | **Total Credits** | 23 |

| Course Code | Semester - V | Credit |
|---|---|---|
| MA301 | Operations Research | 3 |
| CS301 | Programme Core - XII / Compiler Design | 3 |
| CS302 | Programme Core - XIII /Database Management Systems | 3 |
| CS303 | Programme Core - XIV / Artificial Intelligence | 3 |
| E1 | Elective -I | 3 |
| GE1 | Global Elective -I | 3 |
| CS304 | Compiler Design laboratory | 2 |
| CS305 | DBMS laboratory | 2 |
| | **Total Credits** | 22 |

| Course Code | Semester - VI | Credit |
|---|---|---|
| CS351 | Programme Core - XV / Software Engineering | 3 |
| CS352 | Programme Core - XVIII / Machine Learning | 3 |
| CS353 | Programme Core - XVII / Web Technology | 3 |
| E2 | Elective -II | 3 |
| GE2 | Global Elective -II | 3 |
| HM351 | Technical English | 2 |
| CS354 | Machine Learning laboratory | 2 |
| CS355 | Web Technology laboratory | 2 |
| | **Total Credits** | 21 |

| Course Code | Semester - VII | Credit |
|---|---|---|
| CS401 | Summer Internship | 2 |
| CS402 | Programme Core - XVI / Internetworking Protocols | 3 |

| | | |
|---|---|---|
| E3 | Elective -III | 3 |
| E4 | Elective -IV | 3 |
| E5 | Elective -V | 3 |
| E6 | Elective -VI | 3 |
| CS403 | Networks laboratory | 2 |
| CS404 | Comprehensive Viva | 1 |
| | **Total Credits** | 20 |

| **Course Code** | **Semester - VIII** | **Credit** |
|---|---|---|
| CS451 | Project work | 6 |
| E7 | Elective - VII / MOOC online course | 3 |
| E8 | Elective -VIII / MOOC online course | 3 |
| E9 | Elective -IX / MOOC online course | 3 |
| | **Total Credits** | 15 |

## Summary:

| **Branch** | **I Sem** | **II Sem** | **III Sem** | **IV Sem** | **V Sem** | **VI Sem** | **VII Sem** | **VIII Sem** | **Total** |
|---|---|---|---|---|---|---|---|---|---|
| CSE | 19 | 22 | 22 | 23 | 22 | 21 | 20 | 15 | 164 |

# FIRST SEMESTER

| | | |
|---|---|---|
| Course Code | : | **HM101** |
| Course Title | : | **English for Communication** |
| Number of Credits | : | **2** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **GIR** |

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

**CO1** Use the features of communication to express themselves orally in English in an intelligible way.
**CO2** Develop an awareness of problems related to listening in different contexts.
**CO3** Apply reading strategies to comprehend different difficulty levels in English at a speed suited to their needs.
**CO4** Employ strategies to write acceptable sentences and coherent paragraphs in English.

**Course Content:**

**Communication:** An introduction – Its role and importance in the corporate world – Tools of communication – Barriers – Levels of communication – English for Specific purposes.

**Listening:** Listening process & practice – Exposure to recorded & structured talks, class room lectures – Problems in comprehension & retention – Note-taking practice – Listening tests – Importance of listening in the corporate world.

**Reading:** Introduction of different kinds of reading materials: technical & non-technical – Different reading strategies: skimming, scanning, inferring, predicting and responding to content – Guessing from context – Note making – Vocabulary extension.

**Speaking**: Barriers to speaking – Building self-confidence & fluency – Conversation practice- Improving responding capacity – Extempore speech practice – Speech assessment.

**Writing**: Effective writing practice – Effective sentences: role of acceptability, appropriateness, brevity & clarity in writing – Cohesion & coherence in writing – Writing of definitions, descriptions & instructions – Paragraph writing – Perspective Writing – Letter Writing – Introduction to report writing

**Text Books:**


**1** William Strunk Jr. and E.B.White "The Elements of Style", Allyn & Bacon, Pearson Education, 1999.
**2** Dhanavel, S. P., "English And Communication Skills For Students Of Science And Engineering", Orient Black Swan, Chennai, 2009.

**3** Geoffrey Leech, Fan Svartvik, "A Communicative Grammar of English", Pearson Education Asia, 1994.

**Reference Books:**

**1** Krishna Mohan and Meenakshi Raman , "Effective English Communication", Tata McGraw Hill, New Delhi, 2000.
**2** Golding S.R., "Common Errors in English Language", Macmillan, 1978.
**3** Christopher Turk, "Effective Speaking", E & FN Spon, London, 1985**.**

**Web link(s):**

**1** Communication - https://nptel.ac.in/courses/109/104/109104031/
**2** Listening - https://learnenglish.britishcouncil.org/skills/listening
http://www.elllo.org/archive/
**3** Speaking - https://nptel.ac.in/courses/109/106/109106067/
**4** Reading & Vocabulary - https://nptel.ac.in/courses/109/106/109106129/ (Week 1 & 2)
**5** Writing - https://www.time4writing.com/free-writing-resources/
https://www.edx.org/course/academic-and-business-writing
https://www.coursera.org/learn/advanced-writing

| | | |
|---|---|---|
| Course Code | : | **MA101** |
| Course Title | : | **Mathematics - I** |
| Number of Credits | : | **3** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **GIR** |

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

**CO1** Examine the system of linear equations with matrices.
**CO2** Convert linear first order differential equations into separable form.
**CO3** Solve the ordinary linear differential equations with constant coefficients
**CO4** Identify the maxima and minima of multivariable functions
**CO5** Analyze the physical problems that arise in the field of engineering and apply the concepts to solve them.

**Course Content:**

**Matrices:** Rank of a matrix - Consistency of the system of linear equations - linear dependence and independence of vectors. Eigen values and Eigen vectors of a matrix - Caley-Hamilton theorem and its applications - Reduction to diagonal form - Reduction of a quadratic form to canonical form - orthogonal transformation and congruent transformation. Properties of complex matrices - Hermitian, skew-Hermitian and Unitary matrices.

**Ordinary differential equations of first order:** Separable equations - equations reducible to separable form - exact equations - integrating factors. Linear first order equations - Bernoulli's equation - Orthogonal trajectories - Newton's law of cooling - Law of natural growth and decay.

**Ordinary higher order differential equations:** Higher order linear equations with constant coefficients. Euler and Cauchy's equations - Method of variation of parameters - System of linear differential equations with constant coefficients – Applications to electrical circuits.

**Differential Calculus**: Rolle's theorem - Mean value theorem - Taylor's and Maclaurin's theorems (without proof) with remainders – simple illustrations; Functions of several variables - Partial differentiation - Total Differentiation - Euler's theorem and generalization. Maxima and minima of functions of several variables (two and three variables) – Lagrange's method of Multipliers - Change of variables –Jacobians – simple illustrations.

**Multiple Integrals:** Double and triple integrals - computation of surface areas and volumes; change of variables in double and triple integrals.

**Text Books:**


**1** R. K. Jain and S. R. K. Iyengar, Advanced Engineering Mathematics", 5th ed, Narosa Publishing House, 2016.

**2** B.S. Grewal, "Higher Engineering Mathematics", Khanna Publications, 44nd ed, 2015.

**3** Erwin Kreyszig, "Advanced Engineering Mathematics", 8th ed, John Wiley and Sons, 2015.


**Reference Books:**


1 N. Piskunov, "Differential and Integral calculus, Vol. 1&2", MIR Publishers, Moscow - CBS Publishers and Distributors (India).

2 Michael D. Greenberg, "Advanced Engineering Mathematics", Pearson Education Pvt. Ltd.


**Web link(s):**

1 https://nptel.ac.in/courses/111/101/111101115/
https://nptel.ac.in/courses/111/102/111102133/

2 https://nptel.ac.in/courses/111/104/111104092/

| | | |
|---|---|---|
| Course Code | : | **PH101** |
| Course Title | : | **Physics** |
| Number of Credits | : | **3** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **GIR** |

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

**CO1** Define the basic principles of thermodynamics and its significance.
**CO2** Describe electromagnetic theory in the field of signal propagation.
**CO3** Interpret various concepts and theories of waves and quantum optics.
**CO4** Explain the principle of light transmission in a fiber for modern communication.
**CO5** Apply the concepts of semiconductor physics in solid state electronic devices and technologies.

**Course Content:**

**Thermodynamics:** Introduction to thermodynamic system, surrounding, equilibrium, heat and work, Zeroth Law of Thermodynamics, Equation of state of ideal and real gases, Internal energy, first law and its applications enthalpy, second law, reversible and irreversible processes, Carnot cycle, entropy, Maxwell's relations, Clausius- Clayperon equation, Joule-Thomson process, Clausius inequality, entropy as a property, principle of increase of entropy. Calculation of entropy change, Third law.

**Electromagnetics:** Gauss's Theorem of Electrostatics, Ampere's law of Magnetostatics, EMF, Ohm's Law and laws of Electromagnetic Induction, Self and Mutual induction, Concept of Displacement Current, Difference between Conduction Current and Displacement Current, Maxwell's Equations in free space and dielectric media, Propagation of Electromagnetic Waves in Free Space.

**Waves and Quantum Optics:** Wave motion, Wave equation, Superposition of waves along same direction (equal frequency) and in perpendicular directions, Lissajous figures. Transverse waves, solution of wave equation, Theory of interference of light- Newton's rings, Diffraction, applications of Interference (colours of thin films). Diffraction, Fraunhoffer diffraction due to single slit, double slit and, Diffraction grating (N-slit), applications of Diffraction (List only).

**Lasers and Fiber Optics:** Introduction, Coherence, Spontaneous and stimulated emissions, Einstein‟s coefficients, population inversion and lasing action, laser systems: Ruby laser, He-Ne Laser, semiconductor laser, Applications. Fiber Optics Introduction, numerical aperture, different types of fibres, attenuation & dispersion mechanism in optical fibers (Qualitative only), application of optical fibres, Fiber optic communication (block diagram only).

**Semiconductor Physics:** Energy bands; semiconductors different types, charge carriers: electrons and holes, effective mass, doping. Carrier concentration: Fermi level, temperature dependence of carrier concentration. Drift and diffusion of carriers: excess carriers;

recombination and lifetime, Hall effect, p-n Junction: depletion region, forward and reverse-bias, depletion and diffusion capacitances, switching characteristics; breakdown mechanisms.

**Text Books:**


**1** M. N. Avadhanulu and P.G. Kshirsagar, "A textBooks of Engineering Physics", S. Chand and Company, New Delhi 2009.

**2** R.K. Gaur and S.L. Gupta, "Engineering Physics",Dhanpat Rai Publications (P) Ltd., 8th ed., New Delhi 2001.

**3** R. K. Rajput, "A TextBooks of Engineering Thermodynamics" 4th Edition, L.B. Enterprizes, New Delhi 2010.


**Reference Books:**


**1** Halliday, Resnic and Walker, "Fundamentals of Physics", John Wiley, 9 th Edition, 2011.

**2** David J. Griffiths, "Introduction to Electrodynamics", 3rd Edition, Printice Hall of India, New Delhi 2012.

**3** Donald A. Neamen, "Semiconductor Physics and Devices: Basic principle", 4th Edition,, McGraw- Hill, New York 2012

| | | |
|---|---|---|
| Course Code | : | **CS101** |
| Course Title | : | **Introduction to Computer Programming** |
| Number of Credits | : | **3** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **GIR** |

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

**CO1** Explain the basics of computers and software systems.
**CO2** Discuss the various conditional control statements in C programming.
**CO3** Apply the concept of arrays to solve sorting and searching problems.
**CO4** Define pointers and its association with arrays and functions in C.
**CO5** Develop C program with structures and perform Read-Write operations with files.

**Course Content:**

**Introduction to Computers, Number Systems, C language:** Introduction to Computers: Computer Systems, Computing Environments, Computer Languages, Creating and Running Programs, Software Development, Flow charts. Number Systems: Binary, Octal, Decimal, Hexadecimal Introduction to C Language - Background, C Programs, Identifiers, Data Types, Variables, Constants, Input / Output Statements Arithmetic Operators and Expressions: Evaluating Expressions, Precedence and Associativity of Operators, Type Conversions.

**Control Statements:** Conditional Control Statements: Bitwise Operators, Relational and Logical Operators, If, If-Else, Switch-Statement and Examples. Loop Control Statements: For, While, Do While and Examples. Continue, Break and Goto statements Functions: Function Basics, User-defined Functions, Inter Function Communication, Standard Functions, Methods of Parameter Passing. Recursion- Recursive Functions. Storage Classes: Auto, Register, Static, Extern, Scope Rules, and Type Qualifiers.

**Preprocessors, Arrays:** Preprocessors: Preprocessor Commands Arrays - Concepts, Using Arrays in C, Inter-Function Communication, Array Applications, Two- Dimensional Arrays, Multidimensional Arrays, Linear and Binary Search, Selection and Bubble Sort.

**Pointers, Strings:** Pointers - Introduction, Pointers for Inter-Function Communication, Pointers to Pointers, Compatibility, Lvalue and Rvalue, Arrays and Pointers, Pointer Arithmetic and Arrays, Passing an Array to a Function, Memory Allocation Functions, Array of Pointers, Programming Applications, Pointers to void, Pointers to Functions, Command line Arguments. Strings - Concepts, C Strings, String Input/Output Functions, Arrays of Strings, String Manipulation Functions.

**Structures, Input and Output:** Structures: Definition and Initialization of Structures, Accessing Structures, Nested Structures, Arrays of Structures, Structures and Functions, Pointers to Structures, Self-Referential Structures, Unions, Type Definition (typedef), Enumerated Types.

Input and Output: Introduction to Files, Modes of Files, Streams, Standard Library Input/Output Functions, Character Input/Output Functions.

**Text Books:**


**1** R G Dromey, "How to Solve It by Computer", Prentice-Hall International Series in Computer Science, 2006.
**2** G. Michael Schneider, "Invitation to Computer Science", Eighth Edition, 2018.
**3** Byron S Gotrified, "Programming with C", Third Edition, McGraw Hill Companies, 2017.


**Reference Books:**


**1** Michael Vine, "C Programming for the Absolute Beginner", Third Edition, 2014.
**2** Brian W Kernighan, Dennis M. Ritchie, "C Programming Language", Second Edition, Pearson Education India, 2015.
**3** Herbert Schildt, "C++ Complete Reference", McGraw Hill, Fourth Edition, 2017.


**Web link(s):**

**1** http://uru.ac.in/uruonlinelibrary/Cloud_Computing/Basics%20of%20Computer.pdf
**2** https://www.tutorialspoint.com/basics_of_computers/index.htm
**3** https://en.wikiBookss.org/wiki/Computers_for_Beginners/The_Basics
**4** http://ecoursesonline.iasri.res.in/course/view.php?id=94
**5** https://www.tutorialspoint.com/cprogramming/index.htm

| | | |
|---|---|---|
| Course Code | : | **ME101** |
| Course Title | : | **Basics of Mechanical Engineering** |
| Number of Credits | : | **2** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **GIR** |

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

**CO1** Discuss the basic concepts of thermodynamics, systems, and energy resources

**CO2** Explain the basic functions of internal combustion engines, refrigeration, and heat transfer in engineering applications

**CO3** Select a type of power transmission system as per the application

**CO4** Identify the suitable Materials for Engineering Applications

**CO5** Describe the functions and operations of various conventional and advanced manufacturing processes

**Course Content:**

**Thermodynamics:** Thermodynamic system, State, Properties, Thermodynamic Equilibrium, Process and Cycle, Fundamental Units and conversions, Zeroth law of Thermodynamics, Work and Heat, First law- Cyclic process, Change of State, Limitations of First law, Thermal Reservoirs, Heat Engine, Heat Pump/Refrigerator, Efficiency/COP, Second law, PMM2, Carnot Cycle, Entropy - Example problems. Energy Sources - Conventional/Renewable.

**I.C. Engines**: 2-Stroke & 4-Stroke Engines, P-v Diagram; S.I. Engine, C.I. Engine, Differences, **Refrigeration**: Vapour Compression Refrigeration Cycle - Refrigerants, Desirable Properties of Refrigerants **Heat Transfer**: Modes of Heat Transfer, Thermal Resistance Concept, Composite Walls and Cylinders, and Overall Heat Transfer Coefficient - Example problems

**Power Transmission:** Classification of different power transmission systems, Transmission of Power, Belt Drives, Chain Drives, Gears and Gear Trains – Example problems

**Engineering Materials** Properties of materials, Classification of Materials, Selection of Engineering Materials, Introduction to materials structure, Applications, Testing of materials.

**Manufacturing Processes:** Casting - Patterns and Moulding, Hot Working and Cold Working, **Metal Forming processes:** Extrusion, Drawing, Rolling, Forging, Welding - Arc Welding & Gas Welding, Soldering, Brazing. **Advanced manufacturing**: introduction to CNC machines, laser based manufacturing processes, 3D printing.

**Text Books:**

**1** Basant Agarwal and C.M. Agarwal, "Basic Mechanical Engineering", Wiley India Pvt. Ltd., 2008.

**2** Sadhu Singh, “Basic Mechanical Engineering”, S. Chand & Company Limited, 2009.
**3** Praveen Kumar, “Basic Mechanical Engineering”, Pearson Education, India, 2013.

**Reference Books:**

**1** M.L. Mathur, F.S. Mehta and R.P. Tiwari, R.S. Vaishwnar, “Elements of Mechanical Engineering”, Jain Brothers, New Delhi, 2008.
**2** P.N. Gupta, M.P. Poonia, “Elements of Mechanical Engineering”, Standard Publishers, 2004
**3** C.P. Gupta, Rajendra Prakash, “Engineering Heat Transfer”, Nem Chand Brothers, New Delhi, 1994.

**Web link(s):**

**1** https://nptel.ac.in/courses/112/105/112105123/
**2** https://nptel.ac.in/courses/112/103/112103262/
**3** https://nptel.ac.in/courses/112/105/112105234/

| | | |
|---|---|---|
| Course Code | : | **ME102** |
| Course Title | : | **Engineering Practice** |
| Number of Credits | : | **2** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **GIR** |

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

**CO1** Explain the basic manufacturing processes of Casting, Joining, Fitting and Forming.
**CO2** Use hand tools and basic machineries in Foundry, Welding shop, Carpentry, Fitting shop and Sheet Metal work
**CO3** Design simple prototypes and produce engineering products.

**Course Content:**

**Foundry**: Preparation of sand mould for the following
01. Flange
02. Hand Wheel
**Welding**: Fabrication of metals joint of the following
01. Butt Joint
02. Lap Joint
**Carpentry**: Wood sizing exercise in planning, marking, sawing, chiseling and grooving to make
1. Tee through Halving Joint
2. Dovetail Scarf Joint
**Fitting**: Preparation of joints, markings, cutting and filling for making
1. Semi-circle part
2. Dovetail part
**Sheet metal**: Fabrication of simple products of the following
1. Dust Pan
2. Corner Tray

**Text Books:**


**1** R.K. Rajput, "Workshop Practice", Laxmi Publications (P) Limited.
**2** Shashi Kant Yadav, "Workshop Practice", Discovery Publishing House, New Delhi.
**3** K.C.John, "Mechanical workshop practice" PHI Learning Pvt. Ltd., (2010).


**Reference Books:**


**1** H.S. Bawa, "Workshop Practice", Tata McGraw – Hill Publishing Company Limited, (2009).
**2** T.Jeyapoovan, M.Saravanapandian & S.Pranitha, "Engineering Practices Lab Manual", Vikas Puplishing House Pvt.Ltd, (2006).
**3** S K Hajra Choudhury, A K Hajra Choudhury, N. Roy, "Workshop Technology Vol I & II", Media Promoters & Publishers Pvt. Ltd.

**Web link(s):**

| | |
|---|---|
| **1** | https://nptel.ac.in/courses/112/107/112107145/ |
| **2** | https://nptel.ac.in/courses/112/107/112107144/ |
| **3** | https://nptel.ac.in/courses/112/107/112107219 |

| | | |
|---|---|---|
| Course Code | : | **PH102** |
| Course Title | : | **Physics laboratory** |
| Number of Credits | : | **2** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **GIR** |

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

**CO1** Describe the basic scientific principles of the designed experiments
**CO2** Practice the theoretical concepts of physics through experiments
**CO3** Demonstrate experiments through various experimental setups.
**CO4** Evaluate, analyze and interpret the experimental data.
**CO5** Design new devices based on scientific understanding

**Course Content:**

Wavelength of laser using diffraction grating

Wavelength of mercury spectrum – Spectrometer

Radius of curvature of lens – Newton's rings

Numerical aperture of an optical fiber

Field along the axis of a circular coil

Measurement of temperature using thermistor

Thermo e.m.f by Potentiometer

| | | |
|---|---|---|
| Course Code | : | **CS102** |
| Course Title | : | **Introduction to Computer Programming Laboratory** |
| Number of Credits | : | **2** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **GIR** |

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

**CO1** Develop C program for solving basic mathematical problems.
**CO2** Write C program for solving problems that require more iteration.
**CO3** Construct C program for various sorting and searching algorithms.
**CO4** Perform operations related to strings using C functions.
**CO5** Compose file handling programs in C language.

**Course Content:**

Finding the maximum and minimum of given set of numbers

Finding Roots of a Quadratic Equation

Sin x and Cos x values using series expansion

Conversion of Binary to Decimal, Octal, Hexa and Vice versa

Generating a Pascal triangle and Pyramid of numbers

Recursion: Factorial, Fibonacci, GCD

Matrix addition and multiplication using arrays

Bubble Sort, Selection Sort

Programs on Linear Search and Binary Search using recursive and non-recursive procedures.

Functions for string manipulations

Finding the No. of characters, words and lines of given text file

File Handling programs

# SECOND SEMESTER

| | | |
|---|---|---|
| Course Code | : | **MA151** |
| Course Title | : | **Mathematics - II** |
| Number of Credits | : | **3** |
| Prerequisites (Course code) | : | **Mathematics – I** |
| Course Type | : | **GIR** |

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

| | |
|---|---|
| **CO1** | Apply the concepts of gradient, divergence and curl to solve engineering problems |
| **CO2** | Convert line integrals into area integrals and surface integrals into volume integrals |
| **CO3** | Determine the Fourier series for a given function |
| **CO4** | Change the given function into transform coefficients using Fourier transformation. |
| **CO5** | Apply Laplace transforms to solve physical problems arising in engineering |

**Course Content:**

**Vector Calculus:** Scalar and Vector fields - Vector Differentiation - Level surfaces - Directional derivative - Gradient of a scalar field - Divergence and Curl of a vector field – Laplacian.

**Vector Integrals:** Line, surface and volume integrals; Green's theorem in a plane - Gauss Divergence theorem and Stokes' theorem.

**Fourier Series**: Expansion of a function in Fourier series for a given range - Half range sine and cosine expansions

**Fourier Transforms**: Complex form of Fourier series -Fourier transformation and inverse transforms - sine, cosine transformations and inverse transforms - simple illustrations.

**Laplace Transformation:** Laplace transform - Inverse Laplace transform - properties of Laplace transforms - Laplace transforms of unit step function, impulse function and periodic function - Convolution theorem - Solution of ordinary differential equations with constant coefficients and system of linear differential equations with constant coefficients using Laplace transform – Applications to electrical circuits.

**Text Books:**


1 Erwin Kreyszig, "Advanced Engineering Mathematics", 8th edition, John Wiley and Sons, 2015.

2 R. K. Jain and S. R. K. Iyengar, "Advanced Engineering Mathematics", 5th edition, Narosa Publishing House, 2016.

**Reference Books:**


1. B.S. Grewal, “Higher Engineering Mathematics”, Khanna Publications, 44nd ed, 2015.
2. Michael D. Greenberg, “Advanced Engineering Mathematics”, Pearson Education Pvt. Ltd.


**Web link(s):**

1. https://nptel.ac.in/courses/111/105/111105122/
2. https://nptel.ac.in/courses/111/102/111102129/

| | | |
|---|---|---|
| Course Code | : | **ME151** |
| Course Title | : | **Engineering Graphics** |
| Number of Credits | : | **3** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **GIR** |

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

**CO1** Discuss the fundamentals and standards of Engineering drawings/ graphics
**CO2** Visualize the structure of engineering components
**CO3** Create geometric construction, multi-view, dimensioning and detail drawings of typical 3-D engineering objects
**CO4** Develop projections, solid objects and surfaces of engineering components
**CO5** Devise 3D Isometric View in relation with 2D orthographic views

**Course Content:**

**Fundamentals** Drawing standard - BIS, dimensioning, lettering, type of lines, scaling, conventions

**Geometrical constructions** Dividing a given straight line into any number of equal parts, bisecting a given angle, drawing a regular polygon given one side, special methods of constructing a pentagon and hexagon – conic sections – ellipse – parabola – hyperbola - cycloid.

**Orthographic projection:** Introduction to orthographic projection, drawing orthographic views of objects from their isometric views - Orthographic projections of points lying in four quadrants, Orthographic projection of lines parallel and inclined to one or both planes Orthographic projection of planes inclined to one or both planes. Projections of simple solids – axis perpendicular to HP, axis perpendicular to VP and axis inclined to one or both planes.

**Sectioning of solids:** Section planes perpendicular to one plane and parallel or inclined to other plane. **Intersection of surfaces:** Intersection of cylinder & cylinder, intersection of cylinder and cone, and intersection of prisms.

**Development of surfaces:** Development of prisms, pyramids, cylindrical and conical surfaces. **Isometric and perspective projection:** Isometric projection and isometric views of different planes and simple solids, introduction to perspective projection

**Text Books:**


**1** Natrajan K.V., "A text Books of Engineering Graphics", Dhanalakshmi Publishers, Chennai, 2009.
**2** Venugopal K. and Prabhu Raja V., "Engineering Graphics" New Age International (P) Limited, 2008.

**3** Giesecke, F. E., Mitchell, A., Spencer, H., Hill, I., Dygdon, J., and Novak, J., "Technical drawing with engineering graphics", 2016.

**Reference Books:**

**1** Bhatt N.D. and Panchal V.M., "Engineering Drawing", Charotar Publishing House, 50th Edition, 2010.
**2** Basant Agarwal and Agarwal C.M., "Engineering Drawing", Tata McGraw Hill Publishing Company Limited, New Delhi, 2008.
**3** Shah M.B., and Rana B.C., "Engineering Drawing", Pearson, 2nd Edition, 2009.

**Web link(s):**

**1** https://nptel.ac.in/courses/112/103/112103019/
**2** http://www.iitg.ac.in/rkbc/me111.htm

| | | |
|---|---|---|
| Course Code | : | **EC155** |
| Course Title | : | **Basic of Electronics Engineering** |
| Number of Credits | : | **3 (2T+2L)** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **PC** |

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

**CO1** Explain the basic properties of electrical elements, laws and parameters.
**CO2** Illustrate the basic properties of semiconductor devices.
**CO3** Identify different electronic circuits with semiconductor devices and electrical elements.
**CO4** Explain the function of various digital logic gates and blocks.
**CO5** To analyse and design of various modulation and demodulation techniques.

**Course Content:**

Ohms Law -Kirchhoff's Laws - steady state solution of DC Circuits - Introduction to AC circuits - Waveforms and RMS value - power and power factor, single phase and three phase balanced circuits. Principles of operation and characteristics of DC machines, Transformers - Synchronous Machines - three Phase and single phase induction motors

Classification of solids based on energy hand theory - Intrinsic semiconductors - Extrinsic semiconductors - P type and N type - P-N junction – I-V characteristics of PN junction diode - Zener diode - Zener diode characteristics - Half wave and full wave rectifiers - Voltage regulation, SCR, Diac, Triac, Characteristics and simple applications.

Bipolar junction transistor - CB, CE, CC - Configurations and characteristics - Biasing circuits - Field Effect Transistor - Configurations and characteristics - FET amplifier - UJT - characteristics and simple applications - switching transistors - concept of feedback - negative feedback - application in temperature and motor speed control.

Binary number system - AND, OR, NOT, NAND, NOR circuits - Boolean algebra - Exclusive OR gate - Half and Full adders - flip flops - registers and counters - A/D, D/A conversion - Digital computer principle.

Amplitude Modulation: Need for modulation, Amplitude Modulation - Time and frequency domain description, single tone modulation, power relations in AM waves, Generation of AM waves - Basic concepts of Phase Modulation, Frequency Modulation: Single tone Frequency modulation, Narrow band FM, Wide band FM, Types of Pulse modulation- PAM, PWM and PPM. Comparison of FDM with TDM.

**Text Books:**

1 Salivahanan S, "Basic Electrical and Electronics Engineering", Tata McGraw Hill Education (India) Private Limited, New Delhi, 2013

2 Simon Haykin, V. K. , “Analog and Digital Communications”, John Wiley, 2005.
3 Thomas Floyd, “Digital Fundamentals”, Prentice Hall, 10th Edition, 2011.

**Reference Books:**

1 Robert L. Boylestad and Louis Nashelsky, "Electronic Devices & Circuit Theory", 11/e Pearson, 2013.
2 K. Sam Shanmugam, “Analog and Digital Communication”, Wiley, 2005.

**Web link(s):**

1 https://nptel.ac.in/courses/108/101/108101091/ (NPTEL Video by Dr.Mahesh B. Patil from IIT Bombay)
2 https://nptel.ac.in/courses/117/106/117106108/ (NPTEL Video by Prof. Nagendra Krishnapura from IIT Madras)

| | | |
|---|---|---|
| Course Code | : | **CS151** |
| Course Title | : | **Introduction of Python Programming** |
| Number of Credits | : | **3 (2T+2L)** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **PC** |

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

**CO1** Describe the basic data types and operations in Python programming language.
**CO2** Explain the various conditional control statements and string manipulations in Python.
**CO3** Discuss the advanced data types and built-in functions in Python.
**CO4** Develop python programs for simple graphical applications.
**CO5** Construct simple web applications using Django

**Course Content:**

**Introduction**: Installing Python; basic syntax, interactive shell, editing, saving, and running a script. The concept of data types; variables, assignments; immutable variables; numerical types; arithmetic operators and expressions; comments in the program; understanding error messages; Conditions, boolean logic, logical operators; ranges

**Loops, Strings, Files** : Control statements: if-else, loops (for, while); short-circuit (lazy) evaluation, String manipulations: subscript operator, indexing, slicing a string, Files; manipulating files and directories, os and sys modules; text files: reading/writing text and numbers from/to a file; creating and reading a formatted file (csv or tab-separated).

**Datatypes and Functions:** Lists, tuples, and dictionaries; basic list operators, replacing, inserting, removing an element; searching and sorting lists; dictionary literals, adding and removing keys, accessing and replacing values; traversing dictionaries. Design with functions: hiding redundancy, complexity; arguments and return values; formal vs actual arguments, named arguments. Recursive functions.

**Graphics, Images, Classes**: Simple Graphics and Image Processing: turtle module; simple 2d drawing - colors, shapes; digital images, image file formats, image processing. Simple image manipulations with 'image' module (convert to bw, greyscale, blur, etc). Classes: classes, objects, attributes and methods; defining classes; design with classes, data modeling.

**Multithreading, Web development**: Multithreading in Python. Concurrent threads, applications, examples. Web development; introduction to HTML, introduction to Django, models, templates, forms etc.

**Text Books:**


**1** Kenneth Lambert, "Fundamentals of Python: First Programs", Course Technology, Cengage Learning, 2012, ISBN-13: 978-1-111-82270-5
**2** Swaroop, H. "A Byte of Python". Independent, 2013. ISBN: 9781365042911

**3** Pilgrim, Mark, and Simon Willison. “Dive Into Python 3”. Vol. 2. Apress, 2009. ISBN: 9786612825347

**Reference Books:**

**1** Beazley, David M. “Python essential reference”. Addison-Wesley Professional, 2009. ISBN: 0672329786
**2** Beazley, David, and Brian K. Jones. “Python Cook Books: Recipes for Mastering Python 3”. O'Reilly Media, Inc., 2013.
**3** George, Nigel. “Beginning django CMS”. A press, 2015. ISBN: 978-1-4842-1669-9

**Web link(s):**

**1** https://python.swaroopch.com/
**2** https://goalkicker.com/PythonBooks/PythonNotesForProfessionals.pdf
**3** https://www.w3schools.com/python/
**4** https://diveintopython3.problemsolving.io/
**5** https://docs.djangoproject.com/en/3.0/intro/tutorial01/
**6** https://docs.python.org/3/

| | | |
|---|---|---|
| Course Code | : | **CS152** |
| Course Title | : | **Programme Core -I / Digital System Design** |
| Number of Credits | : | **3** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **PC** |

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

**CO1** Reduce Boolean Functions using K-Map.

**CO2** Describe the concept of combinational logic.

**CO3** Analyze sequential circuits using flipflops, counters and registers.

**CO4** Design a VLSI circuit for an application.

**CO5** Devise complicated digital systems using Verilog.

**Course Content:**

**Boolean Algebra and Logic Gates:** Binary codes - Weighted and non-weighted - Binary arithmetic conversion algorithms, Canonical and standard boolean expressions - Truth tables, K-map reduction - Don't care conditions - Adders / Subtractors - Carry look-ahead adder - Code conversion algorithms - Design of code converters - Equivalence functions.

**Combinational Logic:** Binary/Decimal Parallel Adder/Subtractor for signed numbers - Magnitude comparator - Decoders / Encoders - Multiplexers / Demultiplexers - Boolean function implementation using multiplexers

**Synchronous and Asynchronous Sequential Logic:** Sequential logic - Basic latch - Flip-flops (SR, D, JK, T and Master-Slave) - Triggering of flip-flops - Counters - Design procedure - Ripple counters - BCD and Binary - Synchronous counters, Registers - Shift registers - Registers with parallel load, Reduction of state and flow tables - Race-free state assignment - Hazards.

**VLSI:** Introduction to VLSI design - Basic gate design - Digital VLSI design - Design of general boolean circuits using CMOS gates. Verilog Concepts – Basic concepts – Modules & ports & Functions – useful modeling techniques – Timing and delays – user defined primitives. Modeling Techniques.

**Advanced VLSI:** Advanced Verilog Concepts – Synthesis concepts – Inferring latches and flip-flops – Modeling techniques for efficient circuit design. Design of high-speed arithmetic circuits – Parallelism Pipelined Wallace tree tipliers - Systolic algorithms - Systolic matrix multiplication.

**Text Books:**


1. Morris Mano and Michael D. Ciletti, "Digital Design", 6th Edition, Pearson Education, 2018
2. Samir Palnitkar, "Verilog HDL", 2nd Edition, Pearson Education, 2003
3. R. P. Jain, “Modern Digital Electronics”, Fourth Edition, McGraw Hill Education, 2009.


**Reference Books:**


1. Michael D. Ciletti, "Advanced Digital Design with the Verilog HDL”, 2nd Edition, Pearson Education, 2010
2. Charles H. Roth Jr, Larry L. Kinney, “Fundamentals of Logic Design”, Sixth Edition, CENGAGE Learning, 2013.
3. Malvino and Leach, “Digital Principles and Applications”, Eighth Edition, McGrawHill, 2014.


**Web link(s):**

1. https://nptel.ac.in/courses/117/105/117105080/
2. https://freevideolectures.com/course/2319/digital-systems-design

| | | |
|---|---|---|
| Course Code | : | **CS153** |
| Course Title | : | **Programme Core – II / Data Structures** |
| Number of Credits | : | **3** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **PC** |

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

**CO1** Compare Time Complexity and Space Complexity for algorithm.
**CO2** Develop programs to implement linear data structures such as stacks, queues, linked lists, etc.
**CO3** Apply the concept of trees and graph data structures in real world scenarios.
**CO4** Review sorting and searching algorithms.
**CO5** Decide appropriate data structure for any practical problem.

**Course Content:**

**Introduction:** Development of Algorithms - Notations and analysis - Storage structures for arrays - Sparse matrices - Stacks and Queues: Representations and applications.

**Linked list, Stacks, and Queues:** Linked Lists - Linked stacks and queues - Operations on polynomials - Doubly linked lists - Circularly linked lists - Dynamic storage management - Garbage collection and compaction.

**Trees:** Binary Trees - Binary search trees - Tree traversal - Expression manipulation - Symbol table construction - Height balanced trees – AVL trees - Red-black trees.

**Graphs:** Graphs - Representation of graphs - BFS, DFS - Topological sort. String representation and manipulations - Pattern matching.

**Sorting and Searching:** Sorting Techniques - Selection, Bubble, Insertion, Merge, Heap, Quick, and Radix sort - Address calculation - Linear search - Binary search - Hash table methods.

**Text Books:**


**1** J. P. Tremblay and P. G. Sorenson, "An Introduction to Data Structures with applications", Second Edition, Tata McGraw Hill, 1981
**2** M. Tenenbaum and Augestien, "Data Structures using C", Third Edition, Pearson Education 2007.
**3** Mark Allen Weiss, "Data Structures and Algorithm Analysis in C", 2nd edition, Addison-Wesley Educational Publishers, 2006.

**Reference Books:**


1. Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman, "Data Structure and Algorithms", Second Edition, Pearson Education, 2009
2. Sara Baase and Allen Van Gelder, "Computer Algorithms - Introduction to Design and Analysis", Third Edition, Pearson Education, 2008.
3. Sartaj Sahni, "Data Structures, Algorithms and Applications in C++", Universities Press (I) Pvt. Ltd.


**Web link(s):**

1. https://courses.cs.washington.edu/courses/cse373/20sp/
2. https://nptel.ac.in/courses/106/102/106102064/

| | | |
|---|---|---|
| Course Code | : | **SH151** |
| Course Title | : | **Energy and Environmental Science** |
| Number of Credits | : | **0** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **GIR** |

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

**CO1** Familiar with the foundational science concepts and terminology needed to understand the energy and the environment.
**CO2** Analyze the demand for solar and thermal energy.
**CO3** Examine the various types of pollution.
**CO4** Understand the environment pollution along with social issues and acts.
**CO5** Analyze the consequences of today's energy consumption for environmental degradation

**Course Content:**

**Sources of Energy:** A brief survey of various energy sources, present and future needs, energy conservation, renewable and non-renewable energy sources of the world. Estimated reserves of on renewable energy sources.

**Thermodynamics of Energy Conversion:** Principles of energy conversion, conversion between different forms of energy, Thermodynamics of various conversion processes and their comparison in terms of efficiency, Thermodynamic engine cycles and their efficiency.

**Direct Electrical Conversion of Solar Energy:** Photo voltaic effect, solar photo emissive and photo voltaic cell, Solar cell characteristics, efficiency and spectral response of solar cells, Description and comparison of different types of solar cells.

**Environmental Pollution and Control**: Environmental pollution (Air, water, soil, thermal, and noise): causes, effects, and controls; Primary and secondary air pollutants, Air and water quality standards, Nuclear hazards and human health risks.

**Global and Regional Environmental Issues:** Global effects of air pollution – Greenhouse gases, global warming, climate change, urban heat islands, acid rain, ozone hole, Factors influencing increase in population, energy consumption, and environmental degradation.

**Text Books:**


**1** J. Andrews, and N. Jelley, "Energy Science: Principles, Technologies and Impacts", Oxford Universities Press, 2013.
**2** J.A. Fay and D.S. Golomb, "Energy and Environment", Oxford Universities Press, 2011.
**3** An Introduction to Solar Energy for Scientist and Engineers: Sol Wieder.

**Reference Books:**


**1** C S Rao, "Environment pollution control Engineering", New Age International reprint 2015, 2nd edition.

**2** W. C. Turner, S. Doty, and W. C. Truner, "Energy Management Hand book", Fairmont Press 7 th Edition 2009.

**3** G. Boyle, "Renewable energy: Power for a sustainable future", Oxford University press, 2004.

| | | |
|---|---|---|
| Course Code | : | **CS154** |
| Course Title | : | **Digital Laboratory** |
| Number of Credits | : | **2** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **ELR** |

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

**CO1** Devise simplified combinational circuits using basic logic gates

**CO2** Practice combinational circuits using MSI devices

**CO3** Construct sequential circuits like registers and counters

**CO4** Design multipliers using Verilog

**CO5** Simulate combinational and sequential circuits using Verilog

**Course Content:**

Design and implementation of combinational circuits using basic gates for arbitrary functions, code converters.

Design and implement Half/Full Adder and Subtractor using MSI devices.

Design of a 32-bit carry look-ahead adder with logarithmic depth using Verilog.

Design and implement synchronous and asynchronous counters.

Design of a Wallace tree multiplier using Verilog

Design of a 4-bit DSP processor using Verilog

Burning the 4-bit DSP processor on a FPGA

| | | |
|---|---|---|
| Course Code | : | **CS155** |
| Course Title | : | **Data Structures Laboratory** |
| Number of Credits | : | **2** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **ELR** |

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

**CO1** Demonstrate data structures such as stacks, queues, linked lists, trees and graphs.
**CO2** Practice the applications of data structures.
**CO3** Design and analyze the time and space efficiency of the data structure.
**CO4** Identity the appropriate sorting and searching techniques for the given problem.
**CO5** Discuss different data structures to solve real-world problems.

**Course Content:**

Problems in C/C++/ Java using data structures involving arrays, stacks, queues, strings, linked lists, trees, graphs.

Operations on stacks, queues and linked lists

Conversion of infix expressions to postfix and evaluation of postfix expressions

Implementation of priority queue

Implementation of Binary Tree and Binary Search Tree

Implementation of Sorting Techniques

# THIRD SEMESTER

| | | |
|---|---|---|
| Course Code | : | **CS201** |
| Course Title | : | **Programme Core -III / Discrete Structures** |
| Number of Credits | : | **4** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **PC** |

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

**CO1** State basic set equalities, mappings and identify the type of relations.
**CO2** Identify techniques to test the logic of a program and discuss the counting principles.
**CO3** Analyze the computer algorithms using recurrence relations.
**CO4** Examine the properties of algebraic structures such as groups, rings and fields.
**CO5** Apply graph theory to solve the real world problems.

**Course Content:**

**Set, Functions and Relations:** Sets: Set Operations, Countable and Uncountable Set, Functions: Mapping, Inverse, Composition, Partial and Total Function, Binary and N-Ary Operations, Relations: Properties, Representation, Closure, Equivalence Relation, Partial Orderings, Poset, Well Ordered Set, Hasse Diagram, Maximal and Minimal Elements, Lattices, Topological Sorting

**Logic and Proofs:** Propositional Logic, Truth Tables, Tautologies, Contradictions, and Contingencies, Normal Forms, Propositional Equivalences, Predicates and Quantifiers, Nested Quantifiers, Rules of Inference, Introduction to Proofs, Proof Methods and Strategy.

**Induction and Combinatorics:** Peano's Axioms, Mathematical Induction, Strong Induction and Well Ordering; The Basics of Counting, The Pigeonhole Principle, Permutations and Combinations, Binomial Coefficients and Identities, Generalized Permutations and Combinations, Generating Permutations and Combinations.

**Recurrence Relations (RR):** Introduction, Homogenous and Non-homogenous Recurrences and Their Solutions, Solving Linear RR, Linear Homogeneous RR, Linear Nonhomogeneous RR, Generating Functions: Counting Problems and Generating Functions, Using Generating Functions to Solve RR.

**Group, Ring and Field:** Semi-Groups, Monoids, Groups, Subgroups and Their Properties – Cyclic Groups – Generators and Evaluation of Powers- Cosets and Lagrange's Theorem – Permutation Groups and Burnside's Theorem– Isomorphisms and Automorphisms- Homomorphisms and Normal Subgroups– Introduction to Rings, Integral Domains, and Fields.

**Graph Theory:** Definitions, Representation of graph by a matrix and Adjacency list, Trees, Cycles, Properties, Paths and Connectedness, Subgraphs, Graph Isomorphism, Vertex and Edge cuts, Vertex and Edge connectivity, Euler and Hamilton paths, Operations on Graphs

**Text Books:**


**1** Kenneth H.Rosen, "Discrete Mathematics and its Applications", McGraw Hill, Seventh Edition, 2012 (Indian Adaptation by Kamala Krithivasan, IIT Madras)

**2** C.L.Liu and D.P.Mohapatra, "Elements of Discrete Mathematics: A Computer oriented Approach", McGraw Hill, Third Edition, 2012.

3 Tremblay J.P. and Manohar R, "Discrete Mathematical Structures with Applications to Computer Science", Tata McGraw Hill Pub. Co. Ltd, New Delhi, 30th Reprint, 2011.


**Reference Books:**


**1** J.L. Mott, A. Kandel, T.P. Baker, "Discrete Mathematics for Computer Scientists and Mathematicians", Second Edition, PHI, 2008

**2** Ralph. P. Grimaldi, "Discrete and Combinatorial Mathematics: An Applied Introduction", Pearson Education Asia, Delhi, 4th Edition, 2007.

**3** Susanna S. Epp, "Discrete Mathematics with Applications" Cengage Learning, New Delhi, 8th Edition, 2016.


**Web link(s):**

**1** NPTEL video lectures by Prof. Kamala Krithivasan, IIT Madras
https://www.youtube.com/playlist?list=PL8FA5147BB09B2B03

**2** https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-042j-mathematics-for-computer-science-fall-2010/video-lectures/

| | | |
|---|---|---|
| Course Code | : | **CS202** |
| Course Title | : | **Programme Core -IV / Design and Analysis of Algorithms** |
| Number of Credits | : | **3** |
| Prerequisites (Course code) | : | **Data structures** |
| Course Type | : | **PC** |

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

**CO1** Explain the time and space complexity for any algorithm.
**CO2** Describe divide-and-conquer techniques for solving problems.
**CO3** Analyze the graphical problems related to greedy paradigm.
**CO4** Apply dynamic programming, backtracking and branch & bound design technique for solving real-world problems
**CO5** Evaluate NP class of problems and propose approximation algorithms for the same.

**Course Content:**

**Introduction**: What is an algorithm?, Fundamentals of Algorithmic Problem Solving, Analysis of Algorithm Efficiency -The Analysis Framework, Asymptotic Notations and Basic Efficiency Classes, Mathematical Analysis of Nonrecursive and Recursive Algorithms, Example: Computing the nth Fibonacci Number, Empirical Analysis of Algorithms, Algorithm Visualization, Recurrences- Substitution, Recursion-tree and Master method.

**Divide and Conquer**: Merge sort, Quicksort, Binary Tree Traversals and Related Properties, Karatsuba's Large Integer Multiplication and Strassen's Matrix Multiplication, The Closest-Pair and Convex-Hull Problems, Binary Search, Finding Maximum and Minimum, O(n) median finding algorithm.

**Greedy Algorithms**: Prim's and Kruskal's Algorithms for finding Minimum Spanning Tree, Dijkstra's single source shortest path algorithm, Huffman trees and codes, Knapsack Problem, Tree Vertex Splitting, Job sequencing with deadlines.

**Dynamic Programming**: Three Basic Examples, Knapsack Problem and Memory Functions, Optimal Binary Search Trees, Warshall's and Floyd's Algorithms, String Editing, Travelling salesperson problem, Assembly-line scheduling, Matrix-chain multiplication, Longest common subsequence.

**Backtracking**: n-Queens Problem, Hamiltonian Circuit Problem, Subset-Sum Problem- graph coloring problem. Branch and Bound: Assignment Problem, Knapsack Problem, Traveling Salesman Problem. Complexity classes – P, NP and NP-Complete Problems, NP-Hard, Cook's theorem- Clique Decision Problem, NP-Completeness reductions - Vertex cover- 3-CNF-clique, Hamiltonian cycle, Approximation Algorithm, Planar Graph Coloring.

**Text Books:**


1. Anany Levitin, “Introduction to the Design and Analysis of Algorithms”, Third Edition, Pearson Education, 2012.
2. Ellis Horowitz, Sartaj Sahni and Sanguthevar Rajasekaran, “Fundamentals of Computer Algorithms”, Second Edition, Universities Press, 2011.
3. Thomas H Cormen, Charles E Leiserson, Ronald L Rivest, Clifford Stein, “Introduction to Algorithms”, MIT Press, Third Edition, 2010.


**Reference Books:**


1. Steven S. Skiena, “The Algorithm Design Manual”, Second Edition, Springer, 2008.
2. M. Tenenbaum and Augestien, “Data Structures using C”, Third Edition, Pearson Education 2007.
3. Jon Kleinberg, Eva Tardos, “Algorithm Design” ,Pearson Addison, Wesley, 2013.


**Web link(s):**

1. https://www.geeksforgeeks.org/fundamentals-of-algorithms/
2. https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-046j-introduction-to-algorithms-sma-5503-fall-2005/video-lectures/

| | | |
|---|---|---|
| Course Code | : | **CS203** |
| Course Title | : | **Programme Core -V / Programming Paradigms** |
| Number of Credits | : | **3** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **PC** |

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

**CO1** Describe syntax and semantics of programming languages.
**CO2** Explain data, data types, and basic statements of programming languages.
**CO3** Design and implement subprogram constructs.
**CO4** Apply object-oriented, concurrency, and event handling programming constructs.
**CO5** Develop programs in Scheme, ML, and Prolog.

**Course Content:**

**Names, Scopes and Bindings, Control Flow**: Names and Scopes, Binding Time, Scope Rules, Storage Management, Binding of Referencing Environments. Control Flow: - Expression Evaluation, Structured and Unstructured Flow, Sequencing, Selection, Iteration, Recursion, Non-determinacy.

**Data Types**: Type Systems, Type Checking, Records and Variants, Arrays, Strings, Sets, Pointers and Recursive Types, Lists, Files and Input/Output, Equality Testing and Assignment.

**Subroutines and Control Abstraction**: Subroutines and Control Abstraction: - Static and Dynamic Links, Calling Sequences, Parameter Passing, Generic Subroutines and Modules, Exception Handling, Co-routines.

**Functional and Logic Languages**: Lambda Calculus, Overview of Scheme, Strictness and Lazy Evaluation, Streams and Monads, Higher-Order Functions, Logic Programming in Prolog, Limitations of Logic Programming.

**Data Abstraction, Object Orientation and Concurrency:** Encapsulation, Inheritance, Constructors and Destructors, Aliasing, Overloading, Polymorphism, Dynamic Method Binding, Multiple Inheritance. Innovative features of Scripting Languages: -Scoping rules, String and Pattern Manipulation, Data Types, Object Orientation. Concurrency: - Threads, Synchronization. Run-time program Management:- Virtual Machines, Late Binding of Machine Code, Reflection, Symbolic Debugging, Performance Analysis.

**Text Books:**


**1** 1. Michael L. Scott, “Programming Language Pragmatics”, Third Edition, Morgan Kaufmann,2009.
**2** 2. Robert W. Sebesta, “Concepts of Programming Languages”, Tenth Edition, Addison Wesley, 2012.
**3** R. Kent Dybvig, “The Scheme programming language”, Fourth Edition, MIT Press, 2009.

**Reference Books:**


1. Jeffrey D. Ullman, “Elements of ML “, Second Edition, Prentice Hall, 1998.
2. Richard A. O, Keefe, “The craft of Prolog”, MIT Press, 2009.
3. W. F. Clocksin and C. S. Mellish, “Programming in Prolog: Using the ISO Standard”, Fifth Edition, Springer, 2003.


**Web link(s):**

1. https://www.ktustudents.in/2018/11/cs403-programming-paradigms-notes-textBooks-syllabus-question-papers-s7-cse.html
2. https://www.cs.bgu.ac.il/~mira/ppl-Books-full.pdf

| | | |
|---|---|---|
| Course Code | : | **CS204** |
| Course Title | : | **Programme Core -VI / Computer Organization and Architecture** |
| Number of Credits | : | **3** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **PC** |

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

**CO1** Explain the functions of a computer.
**CO2** Differentiate between representations of data.
**CO3** Describe the hierarchical organization of memory.
**CO4** Define the interaction between CPU and peripheral devices.
**CO5** Apply parallel execution mechanism for improving CPU performance.

**Course Content:**

**Basic functional blocks of a computer:** CPU, memory, input-output subsystems, control unit. Instruction set architecture of a CPU - registers, instruction execution cycle, RTL interpretation of instructions, addressing modes, instruction set. Case study - instruction sets of some common CPUs.

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

**CPU Design**: Control unit design - hardwired and micro-programmed design approaches, Case study - design of a simple hypothetical CPU. Memory system design - semiconductor memory technologies, memory organization. Memory interleaving, concept of hierarchical memory organization, cache memory, cache size vs block size, mapping functions, replacement algorithms, write policy. Performance enhancement techniques.

**Peripheral devices and their characteristics**: Input-output subsystems, I/O transfers - program controlled, interrupt driven and DMA, privileged and non-privileged instructions, software interrupts and exceptions. Programs and processes - role of interrupts in process state transitions.

**Parallel Organization**: Pipelining- Basic concepts of pipelining, throughput and speedup, pipeline hazards. Multiple Processor Organizations, Closely and Loosely coupled multiprocessors systems, Symmetric Multiprocessors, Clusters, UMA NUMA, Vector Computations, RISC: Instruction execution characteristics, RISC architecture and pipelining. RISC Vs CISC.

**Text Books:**


1. William Stallings, “Computer Organization and Architecture”, 8 th Edition, Pearson Education, 2010.
2. V. Carl Hamacher, Zvonko G. Varanesic, and Safat G. Zaky, “Computer Organization”, 6th edition, McGraw-Hill Inc, 2012.
3. Smruti Ranjan Sarangi, “Computer Organization and Architecture”, McGraw Hill Education, 2015.


**Reference Books:**


1. David A. Patterson and John L. Hennessey, “Computer organization and design, The Hardware/Software interface”, Morgan Kauffman / Elsevier, Fifth edition, 2014.
2. G. George, “Computer Organization: Hardware and Software”, 2nd Edition, Prentice Hall of India, 1986.
3. J. Hays, “Computer Architecture and Organization”, 2nd Edition, McGraw-Hill, 1988.


**Web link(s):**

1. https://nptel.ac.in/courses/106/105/106105163/
2. https://nptel.ac.in/courses/106/103/106103068/

| | | |
|---|---|---|
| Course Code | : | **CS205** |
| Course Title | : | **Programme core – VII / Object oriented programming (C++)** |
| Number of Credits | : | **3** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **PC** |

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

**CO1** Develop programs efficiently using basic features of C++.

**CO2** Employ object oriented concepts using classes and objects.

**CO3** Explain advanced features of C++ specifically Polymorphism and Inheritance.

**CO4** Design programs with dynamic binding to handle the memory efficiently.

**CO5** Apply standard templates available in C++.

**Course Content:**

**Introduction:** Introduction to Objects, Encapsulation, Polymorphism, Inheritance, Dynamic binding, Message Passing, Abstract Classes, Access Modifiers. Basics of a Typical C++ Environment, Pre-processor Directives, Header Files and Namespaces, Library files.

**Classes and Data Abstraction**: Introduction, Structures - Class - Constructors - Destructors, Const Object And Const Member Functions - Friend Function and Friend Classes, Using This Pointer, Dynamic Memory Allocation, Static Class Members, Container Classes And Integrators, Proxy Classes.

**Polymorphism and Inheritance**: Polymorphism - Function Overloading, Operator Overloading, Inheritance and its types, Casting - Overriding.

**Virtual Functions and Files handling**: Introduction to Virtual Functions - Abstract Base Classes and Concrete Classes - virtual base class - dynamic binding - pure virtual functions. Streams and formatted I/O- File handling - object serialization, namespaces - String - STL.

**Templates and Exception Handling:** Function Templates, Overloading Template Functions, Class Template. Exception Handling: Try, Throw, Catch, Rethrow - Exception specifications.

**Text Books:**

**1** Bjarne Stroustrup, "The C++ Programming Language", Third Edition, Pearson Education, 2000.

**2** Robert Lafore, "Object Oriented Programming in C++", Fourth Edition, Sams Publishers, 2001.

**3** P.J. Deitel, “C++ How to Program”, Prentice-Hall of India Pvt Ltd., Sixth edition, 2013.

**Reference Books:**

**1** E. Balagurusamy, “Object Oriented Programming with C++”, McGraw Hill Company Ltd., 2013.

**2** B. Trivedi, “Programming with ANSI C++”, Oxford University Press, 2012.

**3** Ira Pohl, “Object Oriented Programming using C++”, Pearson Education, Second Edition, Reprint 2013.

**Web link(s):**

**1** https://nptel.ac.in/courses/106/105/106105151/

**2** www.w3schools.com

| | | |
|---|---|---|
| Course Code | : | **CS206** |
| Course Title | : | **Design and Analysis of Algorithms Laboratory** |
| Number of Credits | : | **2** |
| Prerequisites (Course code) | : | **DS** |
| Course Type | : | **ELR** |

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

**CO1** Evaluate the complexity of algorithms.
**CO2** Analyze internal and external sorting algorithms using divide and conquer technique.
**CO3** Demonstrate graph and string algorithms.
**CO4** Apply fundamental algorithms and data structures to real-world problems.
**CO5** Design, develop, and optimize algorithms in different paradigms.

**Course Content:**

Estimating worst-case/average-case complexity of algorithms via programs,
a) Linear and binary search algorithm
b) Basic sorting methods
Implementation of Sorting Algorithms using Divide Conquer Technique.
i) Quick Sort
ii) Merge Sort
iii) Heap Sort
Implementation of Binary Search Tree Algorithm using Divide Conquer Technique.
Implementation of Minimum Spanning Tree using Prim's and Kruskal's Algorithm.
Implementation of Knapsack Problem using Greedy method.
Implementation of Single source shortest path algorithm using greedy method.
Implementation of All Pair Shortest Path Algorithm using Floyd's Algorithm.
Implementation of Matrix-chain multiplication and Longest common subsequence
Implementation of Travelling Salesman Problem using Dynamic Programming
Implementation of 0/1 Knapsack using dynamic programming.
Implementation of 8 Queen's Problem using Backtracking Algorithm.
Implementation of Subset-Sum Problem using Backtracking Algorithm.
Implementation of 0/1 Knapsack using branch and bound programming.

| | | |
|---|---|---|
| Course Code | : | **CS207** |
| Course Title | : | **Programming Paradigms Laboratory** |
| Number of Credits | : | **2** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **ELR** |

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

**CO1** Describe syntax and semantics of programming languages.
**CO2** Explain data, data types, and basic statements of programming languages.
**CO3** Design and implement subprogram constructs.
**CO4** Apply object-oriented, concurrency, and event handling programming constructs.
**CO5** Develop programs in Scheme, ML, and Prolog.

**Course Content:**

Programs to determine type compatibility rules of a C compiler
Program to determine the scope of variables having the same name and different names declared within a while / for loop
Program that behaves differently if name equivalence is used against structural equivalence
Program to determine the factors in passing a large array by reference and the same using value
Program that determines whether it is legal to call a function that has been passed by passing a pointer to it to another function
Devise a subprogram and calling code in which pass-by-reference and pass-by-value-result of one or more parameters produces different results
Design a skeletal program and a calling sequence that results in an activation record instance in which the static and dynamic links point to different activation recorded instances in the run-time stack
Implementation of Functional programming (Scheme, Lisp, ML)
Implementation of Logic programming (Prolog)
Implementation of Imperative programming (ALGOL-60, Pascal, C)

| | | |
|---|---|---|
| Course Code | : | **CS208** |
| Course Title | : | **OOPS laboratory** |
| Number of Credits | : | **2** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **ELR** |

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

| | |
|---|---|
| **CO1** | Design and test programs to solve mathematical and scientific problems |
| **CO2** | Practice code reusability with inheritance property |
| **CO3** | Perform various operations on operators with polymorphism |
| **CO4** | Use standard template library for efficient programming. |
| **CO5** | Apply exception handling techniques to manage runtime exceptions. |

**Course Content:**

Programs on basic C++ features.
Programs on Classes and Objects.
Programs on Inheritance.
Programs on Polymorphism and Virtual Functions.
Programs on File handling.
Programs on Templates
Programs on Exception Handling.

# FOURTH SEMESTER

| | | |
|---|---|---|
| Course Code | : | **MA251** |
| Course Title | : | **Probability, Statistics and Queuing theory** |
| Number of Credits | : | **4** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **GIR** |

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

**CO1** Describe the fundamental concepts of probability.
**CO2** Interpret the concept of random variable to perform regression analysis.
**CO3** Examine the distribution of data through various methods.
**CO4** Test the hypothesis for large and small samples
**CO5** Solve queuing theory problems relevant to engineering applications.

**Course Content:**

**Introduction**: Review of fundamental concepts of probability, Conditional Probability - Baye's Rule - illustrations

**Random Variable**: Concept of a random variable - probability functions, density and distribution functions, mean and variance, Moments and Moment generating function. Probability mass function, Density function, Distribution function – function of two random variables - Covariance – Correlation and Regression analysis

**Distributions**: Bernoulli Trials – Binomials Distribution – Poisson Distribution–Geometric and Hypergeometric distributions. Continuous Uniform Distribution, Exponential distribution, Gamma distribution, Normal Distribution, Areas under the Normal Curve, Applications of the Normal Distribution.

**Estimation & Tests of Hypotheses**: Introduction, Statistical Inference, Classical Methods of Estimation.: Estimating the Mean, Standard Error of a Point Estimate, Prediction Intervals, Tolerance Limits, Estimating the Variance, Estimating a Proportion for single mean, Difference between two Means, between two Proportions for two Samples and Maximum Likelihood Estimation. Statistical Hypotheses: General Concepts, Testing a Statistical Hypothesis, Tests Concerning a Single Mean, Tests on two Means, Test on a Single Proportion, Tests on Two Proportions - Time series analysis

**Queuing theory:** Elements of Queuing model, Exponential distribution, Pure Birth and Pure Death Models, M/M/1 model with finite capacity and infinite capacity.

**Text Books:**


1. S C Gupta and V K Kapoor, “Fundamentals of Mathematical statistics”, Khanna publications.
2. Ronald E. Walpole, Raymond H. Myers, Sharon L. Myers, Keying Ye, “Probability & Statistics for Engineers & Scientists”, 9th edition, Pearson Publishers.


**Reference Books:**


1. T.T. Soong, “Fundamentals of Probability And Statistics For Engineers”, John Wiley & Sons Ltd, 2004.
2. S. D. Sharma, “Operations Research”, Kedarnath and Ramnath Publishers, Meerut, Delhi
3. A. O. ALLEN, “Introduction to Probability, Statistics and Queueing Theory with Computer Science Applications”, Academic Press, 2006 reprint.


**Web link(s):**

1. https://nptel.ac.in/courses/111/104/111104032/

2. https://nptel.ac.in/courses/111/104/111104098/

| | | |
|---|---|---|
| Course Code | : | **HM251** |
| Course Title | : | **Economics for Engineers** |
| Number of Credits | : | **3** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **GIR** |

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

| | |
|---|---|
| **CO1** | Explain the various forms of Business and define the impact of economic variables. |
| **CO2** | Perform demand and supply analysis. |
| **CO3** | Analyze production function, cost analysis, pricing methods suitable for different market structures. |
| **CO4** | Review the elements of Financial Statements and prepare Final Accounts. |
| **CO5** | Discuss and interpret the framework for financial analysis through ratios. |

**Course Content:**

**Introduction to Business and Economics:** Business: Structure of Business Firm, Theory of Firm, Types of Business Entities, Limited Liability Companies, Sources of Capital for a Company, Non-Conventional Sources of Finance. Economics: Significance, Micro and Macro Economic Concepts, Concepts and Importance of National Income, Inflation, Money Supply in Inflation, Business Cycle, Features and Phases of Business Cycle. Nature and Scope of Business Economics, Role of Business Economist, Multidisciplinary nature of Business Economics.

**Demand and Supply Analysis:** Demand and Supply Analysis: Determinants, Law of Demand and supply and its exceptions. Elasticity of Demand and Supply: Definition, Types, Measurement and Significance of Elasticity of Demand and Supply. Demand and Supply Forecasting, Methods of forecasting, Factors governing forecasting

**Production, Cost, Market Structures & Pricing**: Production Function - Isoquants and Isocosts, MRTS, Least Cost Combination of Inputs, Cobb-Douglas Production function, Laws of Returns, Economies of Scale. Cost analysis: Concepts, Types, Short run and Long run Cost Functions, Break Even Analysis (BEA), Determination and Limitations. Market Structures: Nature of Competition and Markets, Features of Perfect competition, Monopoly, and Monopolistic Competition. Pricing: Types of Pricing, Product Life Cycle based Pricing

**Financial Accounting:** Financial accounting objectives, functions, importance, Accounting concepts and Conventions, Double-Entry system of Accounting, Rules for maintaining Bookss of Accounts, Journal, Posting to Ledger, Preparation of Trial Balance, Elements of Financial Statements, Preparation of Final Accounts.

**Financial Analysis through Ratios:** Concept of Ratio Importance, Analysis, and interpretation of Liquidity Ratios, Activity ratio, Turnover Ratios, Profitability Ratios, Proprietary Ratios, Leverage Ratios – Analysis and Interpretation (simple problems).

**Text Books:**


1. Dhanesh K Khatri, “Financial Accounting”, Tata McGraw Hill, 2011.
2. Robert Pindyck, and Daniel Rubinfeld, “Microeconomics”, 9th Edition, Pearson, 2018
3. Geethika Ghosh, Piyali Gosh, Purba Roy Choudhury, Managerial Economics, 2e, Tata McGraw Hill Education Pvt. Ltd. 2012.


**Reference Books:**


1. Paresh Shah, “Financial Accounting for Management”, 2e, Oxford Press, 2015.
2. Lipsey & Chrystel, “Economics”, Oxford University Press, 2012.
3. S. N. Maheshwari, Sunil K Maheshwari, Sharad K Maheshwari, Financial Accounting, 5e, Vikas Publications, 2013.


**Web link(s):**

1. https:// thenthata.web4kurd.net/mypdf/managerial-economics-and- financial analysis
2. https://open.umn.edu/opentextBookss/textBookss/principles-of-microeconomics

| | | |
|---|---|---|
| Course Code | : | **CS251** |
| Course Title | : | **Programme Core -VIII / Data Communication and Networks** |
| Number of Credits | : | **3** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **PC** |

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

**CO1** Describe the basic concepts of data communication and computer networks
**CO2** Articulate the components required to build different types of networks.
**CO3** Apply routing and congestion control algorithms.
**CO4** Identify the lacunae in the existing protocols of various layers of the protocol Stack and propose mechanisms to overcome the gaps.
**CO5** Create communication between hosts.

**Course Content:**

**Data Communications and Computer Networks:** Data Communications, Networks, Protocols and Standards – Analog and Digital, Periodic Analog Signals, Digital Signals, Transmission Impairment, Data Rate Limits, Performance, Uses of Computer Networks, Network Hardware, Network Software, Reference Models, Example Networks

**Physical Layer and Data Link Layer:** Digital Transmission, Analog Transmission, Multiplexing, Guided Transmission Media, Wireless Transmission, Communication Satellites, Switching, PSTN, Mobile Telephone System. Data Link Layer Design Issues, Error Detection and Correction, Elementary Data Link Protocols, Sliding Window Protocols.

**MAC and Network Layer:** Channel Allocation Problem, Multiple Access Protocols, Ethernet, Wireless LANs, Broadband Wireless, Bluetooth, RFID, Data Link Layer Switching. Network Layer Design Issues, Routing Algorithms, Congestion Control Algorithms, Quality of Service, Internetworking, Network Layer in the Internet.

**Transport Layer:** Transport Service, Elements of Transport Protocols, Congestion Control, UDP, TCP, Performance Issues, Delay-Tolerant Networking.

**Application Layer:** DNS, Electronic Mail, World Wide Web, Streaming Audio And Video, Content Delivery

**Text Books:**


**1** Andrew S. Tanenbaum and David J. Wetherall, “Computer Networks”, 5th edition, Prentice Hall, 2011
**2** Behrouz A. Foruzan, “Data Communication and Networking”, 5th edition, Science Engineering& Math Publications, 2013

**3** Larry L. Peterson and Bruce S. Davie, “Computer Networks: A Systems Approach”, Fifth Edition, Morgan Kaufmann Publishers Inc., 2012.

**Reference Books:**

**1** W. Stallings, "Data and Computer Communication", 10th Edition, Pearson Education, 2014

**Web link(s):**

**1** Lecture Series on Data Communication by Prof.A. Pal, Department of Computer Science Engineering,IIT Kharagpur: https://nptel.ac.in/courses/106/105/106105082/
**2** https://nptel.ac.in/courses/106/106/106106091/

| | | |
|---|---|---|
| Course Code | : | **CS252** |
| Course Title | : | **Programme Core - IX / Operating Systems** |
| Number of Credits | : | **3** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **PC** |

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

**CO1** Explain the services rendered by operating systems.
**CO2** Apply the various scheduling policies and provide solutions for critical section and deadlock problems.
**CO3** Describe the various memory management techniques.
**CO4** Discuss the file-system design and its implementation issues.
**CO5** Examine the advantages of distributed systems.

**Course Content:**

**Introduction:** Need for Operating Systems – Computer Systems – OS Structures - OS Operations, -Abstract view of OS – Computing Environments – OS Services - System Calls– Building and Booting OS, Virtualization.

**Process Management**: Process –Threads –Multithreading - Process Scheduling- Process Co-ordination –Synchronization –Semaphores –Monitors –Deadlocks –Methods for Handling Deadlocks.

**Memory Management**: Memory Management Strategies –Contiguous memory allocation – Segmentation – Paging -Virtual memory Management –Demand Paging- Page Replacement Policies – Allocation of frames – Thrashing – Memory mapped files – Allocating kernel memory.

**Storage Management**: Mass Storage Structure –Disk Scheduling –Disk Management – RAID - File System –Basic concepts - File System design and Implementation —I/O Systems- Kernel I/O subsystem - System Protection and Security.

**Distributed Systems**: Distributed Systems –Distributed operating systems –Distributed file systems –Distributed Synchronization, Case study on LINUX and Windows OS.

**Text Books:**


**1** Silberschatz, Galvin, Gagne, "Operating System Concepts", John Wiley and Sons, Tenth edition, 2018.
**2** William Stallings, "Operating Systems –Internals and Design Principles", 8/E, Pearson Publications, 2014.
**3** Andrew S. Tanenbaum, "Modern Operating Systems", 4/E, Pearson Publications, 2014.

**Reference Books:**


1. Dhananjay M. Dhamdhere, “Operating Systems - A Concept-Based Approach”, Tata McGraw-Hill Education, Third Edition, 2012.
2. Charles Crowley, “Operating Systems -Design Oriented Approach”, Mc. Graw Hill Education, First edition, 2017.


**Web link(s):**

1. https://nptel.ac.in/courses/106/105/106105214/
2. https://www.youtube.com/playlist?list=PLGvfHSgImk4Y6htSDkXHZWTaC99rz_ApY

| | | |
|---|---|---|
| Course Code | : | **CS253** |
| Course Title | : | **Programme Core - X / Automata and Formal Languages** |
| Number of Credits | : | **3** |
| Prerequisites (Course code) | : | **Discrete Structures** |
| Course Type | : | **PC** |

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

**CO1** Classify languages based on Chomsky hierarchy
**CO2** Design finite automata or Type x grammar
**CO3** Identify the equivalence of the different language representations within a class of the Chomsky hierarchy
**CO4** Devise Turing machine for any language.
**CO5** Conclude the decidable / undecidable nature of any language.

**Course Content:**

**Finite Automata:** Alphabets, Strings and Languages; Automata and Grammars, Deterministic finite Automata (DFA)-Formal Definition, Simplified notation: State transition graph, Transition table, Language of DFA, Nondeterministic finite Automata (NFA), NFA with epsilon transition, Language of NFA, Equivalence of NFA and DFA, Minimization of Finite Automata, Distinguishing one string from other, Myhill-Nerode Theorem.

**Regular Expression (RE):** Definition, Operators of regular expression and their precedence, Algebraic laws for Regular expressions, Kleen's Theorem, Regular expression to FA, DFA to Regular expression, Arden Theorem, Non Regular Languages, Pumping Lemma for regular Languages. Application of Pumping Lemma, Closure properties of Regular Languages, Decision properties of Regular Languages, FA with output: Moore and Mealy machine, Equivalence of Moore and Mealy Machine, Applications and Limitation of FA.

**Context Free Grammar (CFG) and Context Free Languages:** Definition, Examples, Derivation, Derivation trees, Ambiguity in Grammar, Inherent ambiguity, Ambiguous to Unambiguous CFG, Useless symbols, Simplification of CFGs, Normal forms for CFGs: Chomsky Normal Form (CNF) and Greibach Normal Form (GNF), Closure properties of CFLs, Decision Properties of CFLs: Emptiness, Finiteness and Membership, Pumping lemma for CFLs.

**Push Down Automata (PDA):** Description and definition, Instantaneous Description, Language of PDA, Acceptance by Final state, Acceptance by empty stack, Deterministic PDA, Equivalence of acceptance by empty stack and final state, Conversion of CFG to PDA and PDA to CFG.

**Turing machines (TM) and Undecidability:** Basic model, definition and representation, Instantaneous Description, Language acceptance by TM, Variants of Turing Machine, TM as Computer of Integer functions, Universal TM, Church's Thesis, Recursive and recursively

enumerable languages, Halting problem, Introduction to Undecidability, Undecidable problems about TMs. Post correspondence problem (PCP), Modified PCP and undecidable nature of post correspondence problem. Introduction to recursive function theory.

**Text Books:**


**1** John Hopcroft, Rajeev Motwani and Jeffrey Ullman, "Introduction to Automata Theory, Languages and Computation", Pearson Education, 3rd edition, 2014

**2** Micheal Sipser, "Introduction of the Theory and Computation", Thomson Brokecole, 1997.

**3** K.Krithivasan and R.Rama, "Introduction to Formal Languages, Automata Theory and Computation", Pearson Education, 2009


**Reference Books:**


**1** Mishra K L P and Chandrasekaran N, "Theory of Computer Science - Automata, Languages and Computation", Third Edition, Prentice Hall of India, 2004.

**2** Peter Linz, "An Introduction to Formal Language and Automata", Narosa Pub. House, 2011

**3** Martin J. C., "Introduction to Languages and Theory of Computations", TMH, 4th edition, 2010


**Web link(s):**

**1** NPTEL videos by Prof. Kamala Krithivasan, Department of Computer Science and Engineering, IIT Madras: https://www.youtube.com/playlist?reload=9&list=PLwi7ySzY5bVKn6gPAEHzWC0hQ0Q0nvsA6

**2** NPTEL videos by IIT Guwahati: https://freevideolectures.com/course/3379/formal-languages-and-automata-theory/10

| | | |
|---|---|---|
| Course Code | : | **CS254** |
| Course Title | : | **Programme Core - XI / Embedded Systems** |
| Number of Credits | : | **3** |
| Prerequisites (Course code) | : | **Digital Systems Design** |
| Course Type | : | **PC** |

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

CO1 Describe the overall landscape and characteristics of embedded systems.

CO2 Summarize the architecture and programming aspects of the embedded processor.

CO3 Develop application software for embedded systems using RTOS functions.

CO4 Review Linux capabilities and develop embedded Linux systems.

CO5 Analyze various embedded systems applications.

**Course Content:**

Introduction to Embedded Computing: Characteristics of Embedding Computing Applications, Concept of Real time Systems, Challenges in Embedded System Design, Design Process. Embedded System Architecture: Instruction Set Architecture, CISC and RISC instruction set architecture, Basic Embedded Processor/Microcontroller Architecture (ATOM processor, Introduction to Tiva family etc.)

Designing Embedded Computing Platform: Bus Protocols, Bus Organization, Memory Devices and their Characteristics, Memory mapped I/O, I/O Devices, I/O mapped I/O, Timers and Counters, Watchdog Timers, Interrupt Controllers, Interrupt programming, DMA Controllers, GPIO control, A/D and D/A Converters, Need of low power for embedded systems, Mixed Signals Processing

Programming Embedded Systems: Basic Features of an Operating System, Kernel Features, Real-time Kernels, Processes and Threads, Context Switching, Scheduling, Shared Memory Communication, Message-Based Communication, Real-time Memory Management, Dynamic Allocation, Device Drivers, Real-time Transactions and Files, Realtime OS ( VxWorks, RT-Linux, Psos).

Network Based Embedded Applications: Embedded Networking Fundamentals, Layers and Protocols, Distributed Embedded Architectures, Internet-Enabled Systems, IoT overview and architecture, Interfacing Protocols (like UART, SPI, I2C, GPIB, FIREWIRE, USB,). Various wireless protocols and its applications: NFC, ZigBee, Bluetooth, Bluetooth Low Energy, WiFi. CAN. Overview of wireless sensor networks and design examples

Case studies: Embedded system design using ATOM processors, Galileo and Tiva based embedded system applications.

**Text Books:**


1 Wayne Wolf, "Computers as Components- Principles of Embedded Computing System Design", Morgan Kaufmann Publishers, Second edition, 2008

2 C.M. Krishna, Kang G. Shin, "Real time systems", Mc- Graw Hill, 2010

3 Raj Kamal, Embedded Systems Architecture, Programming, and Design. (2/e), Tata McGraw Hill, 2008.


**Reference Books:**


1 Tim Wilmshurst, "The design of Small –Scale Embedded Systems, Palgrave, 2003.

2 K.V. Shibu, Introduction To Embedded Systems, Tata McGraw, 2009

3 Marwedel Peter, "Embedded System Design, Kluwer Publications, 2004


**Web link(s):**

1 https://swayam.gov.in/nd1_noc20_ee98/preview (SWAYAM NPTEL Video by Prof.Dhananjay from Netaji Subhas University of Technology)

2 https://nptel.ac.in/courses/106/105/106105159/( NPTEL Video by Prof. Anupam Basu from IIT Kharagpur)

| | | |
|---|---|---|
| Course Code | : | **CS255** |
| Course Title | : | **Operating Systems laboratory** |
| Number of Credits | : | **2** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **ELR** |

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

**CO1** Practice various UNIX commands.
**CO2** Create multithread applications using Pthread.
**CO3** Demonstrate various CPU scheduling algorithms and measure its performance.
**CO4** Apply synchronization techniques to solve critical section problems.
**CO5** Employ memory management techniques efficiently.

**Course Content:**

Hands on Unix Commands
Shell programming for file handling
Shell Script programming using the commands grep, awk, and sed
Programs on Multithread using Pthread
Implementation of CPU scheduling algorithms
Implementation of Synchronization problems using Semaphores, Message Queues and Shared Memory
Implementation of Bankers algorithm for deadlock avoidance
Implementation of Memory Management - Allocation, Placement and replacement Algorithms
Implementation of various Disk scheduling algorithms

| | | |
|---|---|---|
| Course Code | : | **CS256** |
| Course Title | : | **Embedded Systems Laboratory** |
| Number of Credits | : | **2** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **ELR** |

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

CO1 Practice the programs in microcontroller.

CO2 Use various interfacing kits with microcontroller.

CO3 Connect various peripheral devices I/O with microcontroller.

CO4 Perform various process and scheduling in RTOS.

**List of Experiments:**

I Basic programming of micro controllers Study of the architecture and instruction set of popular micro controllers (8 bit, 16 bit, 32 bit processors)

1. Assembler and Embedded Programming

2. High level language programming (C, C++) and porting it on a processor

II. Interfacing experiments using microcontrollers

1. Using interrupts and interfacing clocks.

2. Interfacing peripheral devices / IO.

3. Motor speed control

III. RTOS Experiments

1. Introduction to Real-Time /Embedded Operating Systems.

2. Process Management & Inter Process Communication

3. Memory management

4. I/O subsystem

5. Real Time Scheduling

# FIFTH SEMESTER

| | | |
|---|---|---|
| Course Code | : | **MA301** |
| Course Title | : | **Operations Research** |
| Number of Credits | : | **3** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **GIR** |

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

**CO1** Identify input-output-parameters of the real problems and formulate their relationships.
**CO2** Determine optimal solutions for transportation problem.
**CO3** Solve sequencing problems using graphical method.
**CO4** Analyze queuing theory problems relevant to engineering applications.

**Course Content:**

**Linear Programming:** Formulation of LPP, Simplex method, Big-M method. Solution in case of unrestricted variables. Dual linear programming problem. Solution of the primal problem from the solution of the dual problems.

**Transportation Problems:** Balanced and unbalanced Transportation problems. Initial basic feasible solution using N-W corner rule, row minimum method, column minimum, least cost entry method and Vogel's approximation method. Optimal solutions. Degeneracy in Transportation problems.

**Game Theory And Sequencing:** Two Person Zero Sum Game, Pure and Mixed Strategies, Algebraic Solution Procedure, Graphical Solution, Solving by Linear Programming; Sequencing Problem, Processing of n Jobs Through Two Machines and m Machines, Graphical Method of Two Jobs m Machines Problem

**Queueing Theory:** Poisson process and exponential distribution. Pure Birth Death Model, Poisson queues – Model (M/M/1), (M/M/C) and its characteristics.

**Elements of Inventory Control:** Economic lot size problems, Fundamental problems of EOQ. The problem of EOQ with finite rate of replenishment. Problems of EOQ with shortages, production instantaneous, replenishment of the inventory with finite rate. Stochastic problems with uniform demand.

**Text Books:**


**1** Kanti Swarup, Man Mohan and P.K.Gupta, "Introduction to Operations Research", S.Chand & Co., 2006
**2** J.C.Pant, "Introduction to Operations Research", Jain Brothers, New Delhi, 2008.

**Reference Books:**


1. H.A.Taha, “Operations Research: An Introduction”, 7th edition, Person Education, Asia, New Delhi, 2002.
2. N.S.Kambo, “Mathematical Programming Techniques”, East-West Pub., Delhi, 1991.
3. B.S. Grewal, “Higher Engineering Mathematics”, Khanna Publications, 44nd ed, 2015.


**Web link(s):**

1. https://nptel.ac.in/courses/111/107/111107128/
2. https://nptel.ac.in/courses/111/104/111104027/

| | | |
|---|---|---|
| Course Code | : | **CS301** |
| Course Title | : | **Programme Core - XII / Compiler Design** |
| Number of Credits | : | **3** |
| Prerequisites (Course code) | : | **Automata and Formal Languages** |
| Course Type | : | **PC** |

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

**CO1** Develop scanner and parser using LEX & YACC tool.

**CO2** Apply different parsing algorithms to develop the parsers for a given grammar.

**CO3** Recommend the necessity for appropriate code optimization techniques.

**CO4** Conclude the appropriate code generator algorithm for a given source language.

**CO5** Design a compiler for any programming language.

**Course Content:**

**Introduction to Compiling:** Compilers – Analysis of the source program – Phases of a compiler – Compiler construction tools – Lexical Analysis – Role of Lexical Analyzer – Input Buffering – Specification of Tokens, – Recognition of Tokens – Lex – Finite Automata – Regular Expressions to Automata – Minimizing DFA.

**Syntax Analysis:** Role of the parser – Grammars –Context-Free Grammars – Top Down parsing – Recursive Descent Parsing – Predictive Parsing – Bottom-up parsing – Shift Reduce Parsing – Operator Precedent Parsing – LR Parsers – SLR Parser – Canonical LR Parser – LALR Parser - YACC

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

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

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

**Text Books:**


1. Alfred V. Aho, Jeffrey D Ullman, S. Lam, and Ravi Sethi, “Compilers: Principles, Techniques and Tools”, Pearson Education, 2015
2. Jean Paul Tremblay, Paul G Serenson, "The Theory and Practice of Compiler Writing", BS Publications, 2005
3. Dhamdhere, D. M., "Compiler Construction Principles and Practice", 2nd edition, Macmillan India Ltd., New Delhi,2008


**Reference Books:**


1. Allen I. Holub, “Compiler Design in C”, Prentice Hall of India,2003
2. C. N. Fischer and R. J. LeBlanc, “Crafting a compiler with C”, Benjamin Cummings, 2003
3. HenkAlblas and Albert Nymeyer, “Practice and Principles of Compiler Building with C”, PHI,2001


**Web link(s):**

1. https://nptel.ac.in/courses/106/105/106105190/
2. https://freevideolectures.com/course/3051/compiler-design

| | | |
|---|---|---|
| Course Code | : | **CS302** |
| Course Title | : | **Programme Core - XIII /Database Management Systems** |
| Number of Credits | : | **3** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **PC** |

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

**CO1** Explain the architecture of relational database management system.

**CO2** Apply the basics of SQL and construct queries using SQL.

**CO3** Design the database with normalization techniques.

**CO4** Describe transaction concurrency techniques and employ optimal query processing.

**CO5** Develop an efficient storage scheme for saving and retrieving Records and Files.

**Course Content:**

**Introduction:** Purpose of Database System -– Views of data – data models, database management system, three-schema architecture of DBMS, components of DBMS. E/R Model - Conceptual data modeling, entities, entity types, attributes, relationships, relationship types, E/R diagram notation, examples.

**Relational Model:** Relational Data Model - Concept of relations, schema-instance distinction, keys, referential integrity and foreign keys, relational algebra operators, SQL - Introduction, data definition in SQL, table, Data Manipulation in SQL, Querying in SQL- aggregation functions group by and having clauses, embedded SQL, PL/SQL, Triggers. Introduction to NoSQL.

**Database Design:** Dependencies and Normal forms, dependency theory - functional dependencies, Armstrong's axioms for FD's, closure of a set of FD's, minimal covers, Normal Forms - 1NF, 2NF, 3NF and BCNF, 4NF, 5NF, decompositions and its desirable properties.

**Transactions and Query Processing:** Transaction Management - ACID properties, Concurrency control – Schedules - Serializability, Locking Protocols, Recoverability, Query Processing and optimization, Database Recovery methods.

**Storage and Indexing:** Data Storage and Indexes - file organizations, primary, secondary index structures, various index structures - hash-based, dynamic hashing techniques, multi-level indexes, B+ trees.

**Text Books:**


1. A. Silberschatz, Henry F. Korth, and S. Sudharshan, "Database System Concepts", 7th Edition, Tata McGraw Hill, 2010
2. C. J. Date, A. Kannan and S. Swamynathan, "An Introduction to Database Systems", 8th Edition, Pearson Education, 2006.
3. RamezElmasri and Shamkant B. Navathe, "Fundamentals of Database Systems", 7th Edition, Pearson Education, 2015


**Reference Books:**


1. Raghu Ramakrishnan, "Database Management Systems", Third Edition, McGraw Hill,2002.
2. K. Singh, "Database Systems Concepts, Design and Applications", First Edition,Pearson Education, 2006.
3. Peter Rob and Carlos Coronel, "Database System- Design, Implementation and Management",7th edition, Cengage Learning, 2007.


**Web link(s):**

1. https://nptel.ac.in/courses/106/105/106105175/
2. https://freevideolectures.com/course/2668/database-management-system

| | | |
|---|---|---|
| Course Code | : | **CS303** |
| Course Title | : | **Programme Core - XIV /Artificial Intelligence** |
| Number of Credits | : | **3** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **PC** |

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

**CO1** Describe the basic concepts of Artificial Intelligence.
**CO2** Analyze the structures and search strategies for state space search methods.
**CO3** Explain the implementation of various Stochastic Methods.
**CO4** Employ artificial intelligence based knowledge representation and reasoning techniques for problem solving.
**CO5** Apply the concept of artificial intelligence based expert systems to deal with current challenges of real world applications.

**Course Content:**

**Introduction:** History of AI, Intelligence, Knowledge, and Human artifice, Overview of AI application Areas –Propositional Calculus, Predicate Calculus, Using Inference Rules to Produce Predicate Calculus Expressions, Application: A Logic-Based Financial Advisor.

**Structures and search strategies for state space search:** Defining problem as a state space search, Production system, Problem characteristics, Production system characteristics, Issues in the design of search programs, Graph Theory, Strategies for State Space Search, Using the State Space to Represent Reasoning with the Predicate Calculus, Hill Climbing and Dynamic Programming, The Best-First Search Algorithm, Admissibility, Monotonicity, and Informedness, Using Heuristics in Games, Complexity Issues.

**Stochastic Methods and Implementation:** Elements of Counting, Elements of Probability Theory, Applications of the Stochastic Methodology, Bayes' Theorem, Recursion– Based Search, Production Systems, Blackboard Architecture for Problem Solving

**Representation and Reasoning;** Knowledge Representation issues, Using Predicate logic, Representing knowledge using rules, Symbolic reasoning under uncertainty, A Brief History of AI Representational Systems, Conceptual Graphs: A Network Language, Alternative Representations and Ontologies Agent Based and Distributed Problem Solving.

**Case Study:** Applications of AI, Natural Language Processing, Computer Vision, Robotics.

**Text Books:**

**1** G. Luger, "Artificial Intelligence, Structures and Strategies for Complex Problem Solving", Sixth Edition, Addison-Wesley Pearson, 2008.

**2** Elaine Rich and Kevin Knigh, “Introduction to Artificial Intelligence”, McGraw Hill, Third Edition, 2017.
**3** Stuart Russel and Peter Norvig, “AI – A Modern Approach”, Pearson Education, Fourth Edition, 2020.

**Reference Books:**

**1** Michael Negnevitsley, Artificial Intelligence: A guide to Intelligent Systems, Addison Wesley, Third Edition, 2017.
**2** Peter Jackson, “Introduction to Expert Systems”, 3rd Edition, Pearson Education, 2007
**3** C.S. Krishnamoorthy and S. Rajeev, “Artificial Intelligence and Expert Systems for Engineers”, CRC Press, 1996.

**Web link(s):**

**1** https://towardsdatascience.com/artificial-intelligence/home
**2** https://dzone.com/articles/the-beginners-guide-to-artificial-intelligence

| | | |
|---|---|---|
| Course Code | : | **CS304** |
| Course Title | : | **Compiler Design Laboratory** |
| Number of Credits | : | **2** |
| Prerequisites (Course code) | : | **Compiler Design** |
| Course Type | : | **ELR** |

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

**CO1** Practice the working principles of a compiler.

**CO2** Design and implement lexical analyzer using LEX, YACC tools.

**CO3** Apply various parsing techniques.

**CO4** Demonstrate simple code generation techniques.

**CO5** Design a compiler for any programming language..

**Course Content:**

Implementation of symbol table.

Design of lexical analyzers and parsers like recursive-descent parser for a block structured language with typical constructs .

Exercises using LEX and YACC.

Program on Left Recursion elimination and Left factoring, SLR, and LALR.

Implement Intermediate code generation for simple expressions.

Quadruples/Triples generation using LEX and YACC for a subset of a block structured language.

Implementation of simple code optimization techniques

| | | |
|---|---|---|
| Course Code | : | **CS305** |
| Course Title | : | **DBMS Laboratory** |
| Number of Credits | : | **2** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **ELR** |

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

**CO1** Create and manipulate database using SQL statements.

**CO2** Write queries efficiently according to user needs.

**CO3** Apply PL/SQL blocks for handling database.

**CO4** Design GUI to access database.

**CO5** Develop a real-time database application.

**Course Content:**

Working with DDL,DML and DCL

Inbuilt functions in RDBMS, Nested Queries & Join Queries.

Set operators & Views in SQL

PL/SQL –Control Structures, Procedures and Functions.

Triggers, Dynamic & Embedded SQL

Forms & Reports

Database Design and implementation (Mini Project/ Group project)

# SIXTH SEMESTER

| | | |
|---|---|---|
| Course Code | : | **CS351** |
| Course Title | : | **Programme Core - XV /Software Engineering** |
| Number of Credits | : | **3** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **PC** |

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

**CO1** Discuss the impact of software engineering in contemporary business, economic, environmental and societal context.
**CO2** Interpret the flow of different modules in terms of DFD, UML and ER diagrams.
**CO3** Assess the software modules with various testing techniques.
**CO4** Develop the software project management skills.
**CO5** Employ validation and verification techniques to assure software quality.

**Course Content:**

**Introduction and Software Development Life Cycle Models:** Role of Software Engineer, Software Components, Software Characteristics, Software Crisis, Software Engineering Processes, Similarity and Differences from Conventional Engineering Processes, Quality Attributes.
**Assessment*:*** How Software Engineering Changes? Software Development Life Cycle (SDLC) Models: Water Fall Model, Prototype Model, Spiral Model, Evolutionary Development Models, Iterative Enhancement Models, Choosing a social relevant problem-Summary Team Report.

**Requirement and Design:** Requirement Engineering Process: Elicitation, Analysis, Documentation, Review and Management of User Needs, Feasibility Study, Information Modeling, Data Flow Diagrams, UML diagrams**,** Entity Relationship Diagrams, Designing the architecture, Design concepts, Design Patterns.
**Assessment:** Impact of Requirement Engineering in their problem. Decision Tables, SRS Document, IEEE Standards for SRS, Architectural design, component level design, user interface design, WebApp Design. Submission of SRS Document for Team Project.

**Testing:** Testing Objectives, Unit Testing, Integration Testing, Acceptance Testing, Regression Testing, Testing for Functionality and Testing for Performance, Top-Down and Bottom-Up Testing, Software Testing Strategies - Strategies: Test Drivers and Test Stubs, Structural Testing (White Box Testing), Functional Testing (Black Box Testing), Testing conventional applications, object oriented applications, and Web applications, Formal modeling and verification, Software configuration management, Product metrics.
**Assessment:** Team Analysis in Metrics Calculation.

**Software Management:** Project Management Concepts, Process and Project Metrics, Estimation for Software projects, Project Scheduling, Risk Management, Maintenance and Reengineering.
**Assessment:** Preparation of Risk mitigation plan

**Software Quality Assurance:** Quality concepts, Review techniques, Software Quality Assurance (SQA): Verification and Validation, SQA Plans, Software Quality Frameworks.

**Assessment:** Framing SQA Plan. ISO 9000 Models, SEI-CMM Model and their relevance to project Management-other emerging models like People CMM.

**Text Books:**


**1** R. S. Pressman, "Software Engineering: A Practitioners Approach", McGraw Hill, 8th edition, 2014
**2** Ian Sommerville, "Software Engineering", Tenth Edition, Pearson Education, 2015.
**3** Pankaj Jalote, "Software Project Management in practice", Pearson Education, New Delhi, 2002


**Reference Books:**


**1** Rajib Mall, "Fundamentals of Software Engineering", PHI Publication, 4th edition, 2014
**2** Jalote Pankaj, "An Integrated Approach to Software Engineering", Third Edition, Springer, 2010.
**3** Shari Lawrence Pfleeger and Joanne M. Atlee, "Software Engineering: Theory and Practice", Fourth Edition, Prentice Hall, 2010.


**Web link(s):**

**1** Research in Software Engineering (RiSE): https://www.microsoft.com/en-us/research/group/research-software-engineering-rise/
**2** https://www2.cs.siu.edu/~mengxia/Courses%20PPT/435/435ppt.htm
**3** https://www.educba.com/software-development/software-development-tutorials/software-engineering-tutorial/

| | | |
|---|---|---|
| Course Code | : | **CS352** |
| Course Title | : | **Programme Core - XVIII / Machine Learning** |
| Number of Credits | : | **3** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **PC** |

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

**CO1** Explain the basic concepts of Machine learning.
**CO2** Apply the linear models for regression in Machine learning.
**CO3** Use clustering techniques and graphical models of Machine learning algorithms.
**CO4** Review the various reinforcement models of Machine learning.
**CO5** Employ machine learning algorithms to real data and evaluate their performance.

**Course Content:**

**Introduction** – Well defined learning problems - Designing Learning System - Perspectives & Issues in Machine Learning - Types of Machine learning: supervised learning, unsupervised learning and reinforcement learning - The machine learning process.

**Linear models for regression**–Maximum Likelihood Estimation (MLS)–least squares–regularized least squares–The Bias-Variance Decomposition–Bayesian Linear Regression–Linear models for classification– Discriminant functions–Fisher's linear discriminant–Probabilistic generative models–Probabilistic discriminative models–Bayesian logistic regression- Bayesian learning–maximum a posterior (MAP) estimation.

**Clustering** - Mixture Densities – K-means Clustering – Expectation- Maximization algorithm – Mixtures of Latent variables models – Supervised Learning after clustering– Hierarchal Clustering – Choosing the number of clusters – Spectral Clustering – Dimensionality Reduction – Principal Component Analysis (PCA) – Linear Discriminant Analysis (LDA) – Factor analysis – Independent Component Analysis (ICA).

**Graphical models**– Markov random fields – Hidden Markov Models – Representation – learning – Decoding - Inference in graphical models –Monte Carlo models – Sampling.

**Reinforcement Learning** – Elements of reinforcement learning – Model based– temporal difference learning – Generalization – Partially observable states – The learning task – Q-learning- Deep reinforcement learning.

**Text Books:**


**1** Tom Mitchell, "Machine Learning", McGraw-Hill, 1997
**2** Stephen Marsland, "Machine Learning – An Algorithmic Perspective", Second Edition, Chapman and Hall/CRC Machine Learning and Pattern Recognition Series, 2014.
**3** E. Alpaydin, "Introduction to Machine Learning", 2nd edition, Prentice-Hall of India, 2010

**Reference Books:**


1. R.O. Duda, P.E. Hart and D.G. Stork. “Pattern Classification”, Wiley-Inder S cience, 2nd Edition, 2000.
2. T. Hastie, R. Tibshirani and J. Friedman, “The Elements of Statistical Learning”, Springer, 2011
3. Shai Shalev-Shwartz, Shai Ben-David, “Understanding Machine Learning from Theory to Algorithms”, Cambridge University Press, 2014


**Web link(s):**

1. https://towardsdatascience.com/machine-learning/home
2. https://dzone.com/refcardz/machine-learning-predictive

| | | |
|---|---|---|
| Course Code | : | **CS353** |
| Course Title | : | **Programme Core - XVII / Web Technology** |
| Number of Credits | : | **3** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **PC** |

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

**CO1** Design a basic website using HTML and Cascading Style Sheets.

**CO2** Construct dynamic and interactive web sites using appropriate technologies.

**CO3** Write server side programs for real time applications.

**CO4** Develop real client applications with Angular JS.

**CO5** Create web services and utilize them in web applications.

**Course Content:**

**Website Basics:** The Internet – Basic Internet protocols – World wide web – HTTP Request Message – HTTP Response Message – Web Clients – Web Servers -HTML- XHTML - DHTML, Cascading Style sheets(CSS). XML- Document type definition, XML Schemas, Document Object model - Simple API for XML (SAX), Extensible Style sheet Language (XSL)

**Client side Programming:** JavaScript DOM Model-Date and Objects,- Regular Expressions-Exception Handling-Validation-Built-in objects-Event Handling - DHTML with JavaScript – AJAX - JSON introduction – Syntax – Function Files – Http Request – SQL

**Web Servers and its Applications:** Web servers –IIS (XAMPP, LAMPP) and Tomcat Servers – Model View Controller (MVC) architecture. Java Web Technologies in Netbeans - Servlets, Java Server Pages (JSP), Java Server Faces (JSF), JSF Components, Session Tracking, Cookies – PHP - Database Connectivity with MySQL

**Angular JS:** Introduction - Data Binding – Modules – Scopes – Controllers – Expressions – Filters – Directives - Module Loading - Multiple Views and Routing - Dependency Injection – Services – XHR – Server Communication – Testing – Events – Caching – Security – Optimization

**Webservices:** Introduction- Service Oriented Architecture – UDDI, SOAP - Java web services Basics – Creating, Publishing, Testing and Describing a Web services (WSDL)-Consuming a web service, Database Driven web service from an application, Web 2.0 technologies, Introduction to semantic web.

**Text Books:**


1. Paul J. Deitel, Harvey M. Deitel, Abbey Deitel, "Internet & World Wide Web How to Program", Deitel series, 5th edition, 2018
2. Ari Lerner, "ng-Books The Complete Books on AngularJS", Fullstack.io, 2013.
3. Ron Schmelzer, Travis Vandersypen, Jason Bloomberg, Madhu Siddalingaiah, Sam hunting, Micheal D.Qualls, David Houlding, Chad Darby, Diane Kennedy, "XML and Web Services", Sams, 2002.


**Reference Books:**


1. Robert W. Sebesta, "Programming with World Wide Web", Addison Wesley, 7th edition, 2013
2. Jeffrey C and Jackson, — "Web Technologies - A Computer Science Perspective", Pearson Education, 2011.
3. Jason Gilmore, "Beginning PHP and MySQL From Novice to Professional", 4th Edition, Apress Publications, 2010


**Web link(s):**

1. www.w3schools.com
2. https://web.stanford.edu/class/cs142/lectures.html

| | | |
|---|---|---|
| Course Code | : | **HM351** |
| Course Title | : | **Technical English** |
| Number of Credits | : | **2** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **GIR** |

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

**CO1** Develop competence in English for independent and effective professional communication.
**CO2** Apply thinking strategies to convince people.
**CO3** Evaluate the scenario and decide the suitable writing style.
**CO4** Identify suitable language to persuade and to reasonably present the analysis of a situation related to his/her profession.

**Course Content:**

**Listening: Barriers to listening:** Physical & psychological – Steps to overcome them – Purposive listening practice – Active listening and anticipating the speaker – Use of technology in the professional world - online listening

**Speaking:** Fluency & accuracy in speech – Positive thinking – Kinds of thinking – Improving self-expression – Tonal variations – Listener oriented speaking – Group discussion practice– Interpersonal Conversation – Developing persuasive speaking skills.

**Reading:** Speed reading practice – Use of extensive readers –Trans-coding: verbal and nonverbal – Analytical and critical reading practice – Introduction to ethics & values through case-study materials.

**Writing:** Professional Correspondence – Formal letters – CV/Resume – Argument Writing – Perspectives in writing – Narrative writing – Different registers – Tone in formal writing – Report Writing – Writing SOP - online tools to effective writing - publishing online - blog writing

**Study Skills:** Reference Skills - Use of dictionary, thesaurus etc. – Importance of contents page, cover & back pages – Bibliography - use of online resources

**Text Books:**

**1** Herta A Murphy, Herbert W Hildebrandt, and Jane P Thomas, “Effective Business Communication”, 7th Edition, McGraw Hill, Irwin, 1997.
**2** Martin Hewings, “Advanced Grammar in Use”, 2nd Edition, Cambridge University Press, 2008.
**3** Michae Swan, “Practical English Usage”, Oxford University Press, Oxford, 1995.

**Reference Books:**

1. Perelman, Leslie C, James Paradis, and Edward Barrett, “The Mayfield HandBooks of Technical & Scientific Writing”, Mountain View, Calif: Mayfield Pub. Co, 1998.
2. Robert Gannon, “Best Science Writing: Readings and Insights”, University Press, Hyderabad, 2000.
3. Shirley Taylor, “Communication for Business”, Longman, New Delhi, 1999.


**Web link(s):**

1. https://nptel.ac.in/courses/109/106/109106094/ (NPTEL Course by Prof. Aysha Iqbal, IITM)
2. https://www.youtube.com/watch?v=lQrj_7xkeNI - Technical Presentation (Part of an NPTEL Course by Prof. Prathap Haridoss, IITM)
3. https://www.youtube.com/watch?v=9SB4tfD0hxM - Technical Writing
4. https://writingcenter.fas.harvard.edu/pages/strategies-essay-writing

| | | |
|---|---|---|
| Course Code | : | **CS354** |
| Course Title | : | **Machine Learning Laboratory** |
| Number of Credits | : | **2** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **ELR** |

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

**CO1** Analyze the performance of supervised learning algorithms.
**CO2** Apply clustering techniques on real data.
**CO3** Compare the performance of dimensionality reduction techniques.
**CO4** Demonstrate reinforcement learning algorithm with real world datasets.

**Course Content:**

Implementation of supervised learning classification and regression using few datasets.
Implementation of naive bayes classifier.
Implementation of clustering using K-Means algorithm.
Implementation of Dimensionality reduction using LDA.
Implementation of Dimensionality reduction using PDA.
Implementation of Reinforcement learning.

| | | |
|---|---|---|
| Course Code | : | **CS355** |
| Course Title | : | **Web Technology Laboratory** |
| Number of Credits | : | **2** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **ELR** |

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

**CO1** Develop web applications with suitable technologies

**CO2** Change the layout of web pages according to user needs.

**CO3** Establish database access from the web pages.

**CO4** Create web services and use them in web applications.

**CO5** Integrate advanced technologies easily with the existing web applications.

**Course Content:**

Designing a static web page using HTML. 3

Designing a dynamic webpage using JAVASCRIPT.

Implement three-tier architecture using Servlets / JSP / PHP with MySQL as database.

Parsing an XML document

Programming using Angular JS.

Programming using AJAX, UDDI, SOAP, WSDL.

Mini Project

# SEVENTH SEMESTER

| | | |
|---|---|---|
| Course Code | : | **CS402** |
| Course Title | : | **Programme Core - XVI / Internetworking Protocols** |
| Number of Credits | : | **3** |
| Prerequisites (Course code) | : | **Data Communication** |
| Course Type | : | **PC** |

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

**CO1** Review the networks, topologies, and layered communication architecture.
**CO2** Explain various concepts of IPv4 comprehensively.
**CO3** Distinguish between IPv4 and IPv6.
**CO4** Apply TCP for efficient communication.
**CO5** Design and develop Mobile IP.

**Course Content:**

**Internet Address Architecture:** Review of TCP/IP Reference Model, Topology and switching, IEEE Standard 802 from Ethernet, Token Bus, Token Ring and Wireless LAN, Connecting Devices, Internet Address Architecture, Link Layer, ARP: Address Resolution Protocol.

**IPV4:** IPv4 headers, IP forwarding, Mobile IP, Host Processing of IP datagrams, DHCP and Autoconfiguration, Firewalls and NAT, ICMPv4, Broadcasting and Local Multicasting (IGMP and MLD), User Datagram Protocol (UDP) and IP Fragmentation, DNS.

**IPV6:** IPv6 Transition issues, IP Security Protocol (Ipsec), Protocol basics, IPv6 Addressing, IPv6 Options and Extension Headers, IPv6 Multicast, IPv6 Anycast, IPv6 Internet Control Message Protocol (ICMPv6), Neighbor Discovery, Routing, Quality of Service, Auto configuration, Mobile IPv6, Multicast Listener Discovery (MLD), IPv6 and DNS, Next Generation Protocols.

**TCP:** TCP Preliminaries, TCP Connection Management, TCP Timeout and Retransmission, TCP Data Flow and Window Management, TCP Congestion Control, TCP Keepalive, Stream Control Transmission Protocol (SCTP), Services, SCTP Association management, SCTP flow and error control.

**Overview of Mobile IP:** Need for Mobile IP, Overview of Mobile IP, Details of Mobile IP, Tunneling, and Mobility for IPv6, Applications of Mobile IP – Security primer, Campus Mobility, Internet wide mobility, A service provider perspective.

**Text Books:**

**1** W. Richard Stevens and G. Gabrani, “TCP/IP Illustrated: The Protocols”, Pearson, 2011
**2** Peter Loshin, Morgan Kaufmann, ”IPv6: Theory, Protocol, and Practice”, 2nd Ed,2003
**3** James Solomon, “Mobile IP: The Internet Unplugged”, 1st Ed, Pearson Education, 2008

**Reference Books:**

**1** Kevin R. Fall and W. Richard Stevens, “TCP/IP Illustrated, Vol. 1- The Protocols”, 2nd Edition, Addison-Wesley, 2012
**2** Silvia Hagen, “IPv6 Essentials, 2nd Edition, O'Reilly Media, 2006
**3** Charles E. Perkins, “Mobile IP: Design Principles and Practices”, Pearson Education, 2008

**Web link(s):**

**1** https://nptel.ac.in/courses/106/105/106105183/
**2** https://people.kth.se/~maguire/courses/IK1550/Coursepage-Spring-2013.html

| | | |
|---|---|---|
| Course Code | : | **CS403** |
| Course Title | : | **Networks Laboratory** |
| Number of Credits | : | **2** |
| Prerequisites (Course code) | : | **None** |
| Course Type | : | **ELR** |

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

**CO1** Practice the analytical studies of Computer Networks through network simulation.
**CO2** Design a network using NS-3 toolkit and explain its importance in a real network.
**CO3** Measure and analyze the network parameters for a high throughput network.
**CO4** Compare various routing algorithms.
**CO5** Differentiate the various transport layer protocols.

**Course Content:**

Exercises on Socket Programming using C and Java

Exercises using NS-3 Network Simulator

Hands on experiments on Network equipments

a. Switches, Routers
b. Hardware firewall

Implementation of error detection and correction techniques

Implementation of Stop and Wait Protocol and sliding window

Implementation of IP address configuration

Implementation of routing algorithms

a. Distance vector

b. Link state routing

Client Server Program using TCP sockets

a. Date and Time Server

b. Chat application

Client Server Program using UDP

a. DNS Implementation

b. Chat application