

Syllabus

Savitribai Phule Pune University

Faculty of Engineering

Second Year of Computer Engineering

(Course 2015)

(with effect from June 2016)

Computer Engineering

Program Education Objectives-

- 1) To prepare globally competent graduates having strong fundamentals, domain knowledge, update with modern technology to provide the effective solutions for engineering problems.
- 2) To prepare the graduate to work as a committed professional with strong professional ethics and values, sense of responsibilities, understanding the legal, safety, health, societal, cultural and environmental issues.
- 3) To prepare the committed and motivated graduate with research attitude, lifelong learning, investigative approach, and multidisciplinary thinking.
- 4) To prepare the graduate with strong managerial and communication skills who can work effectively as individual or in team.

Program Outcomes-

Students are expected to know and be able to do by the time of graduation-

1. Ability to apply Knowledge of problem solving, algorithmic analysis, Mathematical Modeling to computer Engineering
2. Ability to analyze the problem by finding its domain and applying the skill related to domain of the problem in computer Engineering
3. Ability to understand the design issues of the product/software and develop effective solutions
4. Ability to find solutions of complex problem by conduct Investigations by applying suitable techniques.
5. Ability to adapt the usage of modern tools and new releases of the software upcoming in the societies.
6. Ability to Contribute towards the computer society by understanding the impact of Engineering on global aspect.
7. Ability to understand environment issues and design a sustainable system
8. Ability to follow the ethics as computer professional and proceed with legal bindings of the society.
9. Ability to furnish performance effectively at individual level and also in a team.
10. Ability to accomplish effective communication at various levels.
11. Ability to apply the Knowledge of Computer Engineering for development of projects, its finance and management.
12. Ability to keep in touch with current technologies and inculcate the practice of lifelong learning.

Savitribai Phule University of Pune, Pune												
Second Year Computer Engineering (2015 Course)												
(With effect from Academic Year 2016-17)												
Semester I												
Course Code	Course	Teaching Scheme Hours / Week			Semester Examination Scheme of Marks						Credit	
		Theory	Tutorial	Practical	In-Sem	End-Sem	TW	PR	OR	Total	TH/TW	PR+OR
210241	Discrete Structures & Graph Theory	04	--	--	50	50	--	--	--	100	04	--
210242	Digital Electronics and Logic Design	04	--	--	50	50	--	--	--	100	04	--
210243	Data Structures and Algorithms	04	--	--	50	50	--	--	--	100	04	--
210244	Computer Organization and Architecture	04	--	--	50	50	--	--	--	100	04	--
210245	Object Oriented Programming	04	--	--	50	50	--	--	--	100	04	--
210246	Digital Electronics Lab	--	--	02	--	--	25	50	--	75	--	01
210247	Data Structures Lab	--	--	04	--	--	25	50	--	75	--	02
210248	Object Oriented Programming Lab	--	--	02	--	--	25	50	--	75	--	01
210249	Soft Skills	--	--	02	--	--	25	--	--	25	--	01
Total											20	05
210250	Audit Course 1	--	--	--	--	--	--	--	--	--	Grade: AP	
Total		20	--	10	250	250	100	150	--	750	25	

Abbreviations:

TW: Term Work

OR: Oral

PR: Practical

TH: Theory

TUT: Tutorial

AP: Passed Audit Course

Savitribai Phule University of Pune, Pune
Second Year Computer Engineering (2015 Course)
 (With effect from Academic Year 2016-17)
Semester II

Course Code	Course	Teaching Scheme Hours / Week			Semester Examination Scheme of Marks						Credit	
		Theory	Tutorial	Practical	In-Sem	End-Sem	TW	PR	OR	Total	TH/TW TUT	PR+OR
207003	Engineering Mathematics III	04	01	--	50	50	25	--	--	125	05	--
210251	Computer Graphics	04	--	--	50	50	--	--	--	100	04	--
210252	Advanced Data Structures	04	--	--	50	50	--	--	--	100	04	--
210253	Microprocessor	04	--	--	50	50	--	--	--	100	04	--
210254	Principles of Programming Languages	03	--	--	50	50	--	--	--	100	03	--
210255	Computer Graphics Lab	--	--	02	--	--	25	50	--	75	--	01
210256	Advanced Data Structures Lab	--	--	04	--	--	25	50	--	75	--	02
210257	Microprocessor Lab	--	--	04	--	--	25	50	--	75	--	02
Total											20	05
210258	Audit Course 2		--	--	--	--	--	--	--	--	Grade: AP	
Total		19	01	10	250	250	100	150	--	750	25	

Abbreviations:

TW: Term Work

TH: Theory

OR: Oral

TUT: Tutorial

PR: Practical

AP: Passed Audit Course

Savitribai Phule Pune University, Pune Second Year of Computer Engineering (2015 Course) 210241: Discrete Structures and Graph Theory		
Teaching Scheme: TH: 04 Hours/Week	Credit 04	Examination Scheme: In-Sem: 50 Marks End-Sem: 50 Marks
Prerequisite:- Basic Mathematics		
Course Objectives: <ul style="list-style-type: none"> • To use appropriate set, function or relation models to understand practical examples, interpret the associated operations and terminology in context • Determine number of logical possibilities of events • Learn logic and proof techniques to expand mathematical maturity • Formulate problems precisely, solve the problems, apply formal proof techniques, and explain their reasoning clearly 		
Course Outcomes: On completion of the course, learner will be able to– <ul style="list-style-type: none"> • Solve real world problems logically using appropriate set, function, or relation models and interpret the associated operations and terminology in context • Analyze and synthesize the real world problems using discrete mathematics 		
Course Contents		
Unit I	Set Theory and Logic	08 Hours
Discrete Mathematics, Significance of Discrete Mathematics in Computer Engineering, Sets – Naïve Set Theory (Cantorian Set Theory), Axiomatic Set Theory, Need for Sets, Representation of Sets, Set Operations, cardinality of set , Types of Sets – Bounded and Unbounded Sets, Countable and Uncountable Sets, Finite and Infinite Sets, Countably Infinite and Uncountably Infinite Sets, Countability of Rational Numbers Using Cantor Diagonalization Argument, power set, Propositional Logic - logic, Propositional Equivalences, Application of Propositional Logic- Translating English Sentences, Proof by mathematical induction and strong mathematical induction.		
Unit II	Relations and Functions	08 Hours
Relations and Their Properties, n-ary Relations and Their Applications , Representing Relations , Closures of Relations, Equivalence Relations, Partial Orderings, partitions, Hasse Diagram, Lattices, Chains and Anti-Chains, Transitive Closure and Warshall’s Algorithm, n-Ary Relations and their Applications. Functions - surjective, Injective and Bijective functions, Inverse Functions and Compositions of Functions, The Pigeonhole Principle.		
Unit III	Counting	08 Hours

The Basics of Counting, Permutations and Combinations, Binomial Coefficients and Identities, Generalized Permutations and Combinations, Algorithms for generating Permutations and Combinations.

Unit IV	Graph Theory	08 Hours
----------------	---------------------	-----------------

Graphs and Graph Models, Graph Terminology and Special Types of Graphs, Representing Graphs and Graph Isomorphism, Connectivity, Euler and Hamilton Paths, Shortest-Path Problems, Planar Graphs, Graph Colouring. Applications of Graph Theory in Software – Graph Database, Web Graph.

Unit V	Trees	08 Hours
---------------	--------------	-----------------

Introduction, properties of trees, Binary search tree, decision tree, prefix codes and Huffman coding, cut sets, Spanning Trees and Minimum Spanning- Kruskal's and Prim's algorithms, The Max flow- Min Cut Theorem (Transport network). Application- Game Tree, Min-Max Tree

Unit VI	Algebraic Structures and Coding Theory	08 Hours
----------------	---	-----------------

The structure of algebra, Algebraic Systems, Semi Groups, Monoids, Groups, Homomorphism and Normal Subgroups, and congruence relations, Rings, Integral Domains and Fields, coding theory, Polynomial Rings and polynomial Codes, Galois Theory –Field Theory and Group Theory.

Books:

Text:

1. Kenneth H. Rosen, Discrete Mathematics and its Applications, Tata McGraw-Hill, ISBN 978-0-07-288008-3, 7th Edition.
2. C. L. Liu, Elements of Discrete Mathematics, Tata McGraw-Hill, ISBN: ISBN 10:0-07-066913-9.

Reference:

1. Bernard Kolman, Robert C. Busby and Sharon Ross, Discrete Mathematical Structures, Prentice-Hall of India /Pearson, ISBN: 0132078457, 9780132078450.
2. N. Biggs, "Discrete Mathematics", 3rd Edition, Oxford University Press, ISBN 0 –19 850717 – 8.
3. Narsingh Deo, "Graph with application to Engineering and Computer Science", Prentice Hall of India, 1990, 0 – 87692 – 145 – 4.
4. Dr. K. D. Joshi, Foundations of Discrete Mathematics, New Age International Limited, Publishers, January 1996, ISBN: 8122408265, 9788122408263
5. Eric Gossett, "Discrete Mathematical Structures with Proofs", Wiley India Ltd, ISBN:978-81-265-2758-8.
6. Sriram P & Steven S., "Computational Discrete Mathematics", Cambridge University Press, ISBN 13: 978-0-521-73311-3.

Savitribai Phule Pune University, Pune Second Year of Computer Engineering (2015 Course) 210242: Digital Electronics & Logic Design		
Teaching Scheme: TH: 04 Hours/Week	Credit 04	Examination Scheme: In-Sem: 50 Marks End-Sem: 50 Marks
Prerequisite: - Basic Electronics Engineering		
Course Objectives: <ul style="list-style-type: none"> • To understand the functionality and design of Combinational and Sequential Circuits • To understand and compare the functionalities, properties and applicability of Logic Families. • To understand concept of programmable logic devices and ASM chart and get acquainted with design of synchronous state machines. • To design and implement digital systems using VHDL and microcontroller. 		
Course Outcomes: On completion of the course, learner will be able to– <ul style="list-style-type: none"> • Solve K-MAPs and Boolean Algebraic assignments for designing digital circuits. • Design and implement Sequential and Combinational digital circuits as per the specifications. • Apply the knowledge to select the logic families IC packages as per the design specifications. • Design the minimum systems using VHDL. • Develop minimum embedded system for simple real world application. 		
Course Contents		
Unit I	Combinational Logic Design	08 Hours
<p>Logic minimization: Representation of truth-table, Sum of Product (SOP) form, Product of Sum (POS) form, Simplification of logical functions, Minimization of SOP and POS forms using K-Maps up to 4 variables and Quine- McCluskey Technique, realization of logic gates.</p> <p>Design of Combinational Logic: Code converter - BCD, Excess-3, Gray code, Binary Code. Half- Adder, Full Adder, Half Subtractor, Full Subtractor, Binary Adder (IC 7483), BCD adder, Look ahead carry generator, Multiplexers (MUX): MUX (IC 74153, 74151), MUX tree, Demultiplexers (DEMUX)- Decoder. (IC 74138, IC 74154). DMUX Tree, Implementation of SOP and POS using MUX, DMUX, Comparators, Parity generators and Checker, Priority Encoders.</p>		
Unit II	Sequential Logic Design	08 Hours
<p>Flip- flop: SR, JK, D, T; Preset & Clear, Master and Slave Flip Flops, Truth Tables and Excitation tables, Conversion from one type to another type of Flip Flop.</p>		

<p>Registers: Buffer register, shift register, Applications of shift registers</p> <p>Counters: Asynchronous counter. Synchronous counter, ring counters, BCD Counter, Johnson Counter, Modulus of the counter (IC 7490)</p> <p>Synchronous Sequential Circuit Design: Models – Moore and Mealy, State diagram and State Tables, Design Procedure, Sequence generator and detector</p> <p>Asynchronous Sequential Circuit Design: Difference with synchronous circuit design, design principles and procedure, applications</p>		
Unit III	Algorithmic State Machines	08 Hours
<p>Algorithmic State Machines: Finite State Machines (FSM) and ASM, ASM charts, notations, construction of ASM chart and realization for sequential circuits, Examples: Sequence Generator, Types of Counter.</p> <p>VHDL: Introduction to HDL, Data Objects & Data Types, Attributes., VHDL- Library, Design Entity, Architecture, Modeling Styles, Concurrent and Sequential Statements,</p> <p>Design Examples: VHDL for Combinational Circuits-Adder, MUX, VHDL for Sequential Circuits, Synchronous and Asynchronous Counter.</p>		
Unit IV	Programmable Logic Devices	08 Hours
<p>ROM as PLD, Programmable Logic Array (PLA), Programmable Array Logic (PAL), Designing combinational circuits using PLDs.</p>		
Unit V	Logic Families	08 Hours
<p>Classification of logic families: Unipolar and Bipolar Logic Families, Characteristics of Digital ICs: Speed, power dissipation, figure of merits, fan-out, Current and voltage parameters, Noise immunity, operating temperature range, power supply requirements.</p> <p>Transistor-Transistor Logic: Operation of TTL, Current sink logic, TTL with active pull up, TTL with open collector output, Schottkey TTL, TTL characteristics, TTL 5400/7400 series,</p> <p>CMOS: CMOS Inverter, CMOS characteristics, CMOS configurations- Wired Logic, Open drain outputs, Interfacing: TTL to CMOS and CMOS to TTL. Tristate Logic and Tristate TTL inverter.</p>		
Unit VI	Microcontrollers	08 Hours
<p>Comparison of typical microprocessor and microcontroller.</p> <p>Microcontroller 8051: Features, architecture, Pin description, Programming model– Special Function Registers, addressing modes, instruction set, Timers and Counters, serial communication, interrupts, interfacing with ADC and DAC.</p>		
Books:		

Text:

1. R.P. Jain, “Modern Digital Electronics”, Tata McGraw-Hill, 2012, ISBN–13: 978-0-07-066911-6.
2. Stephen Brown, Zvonko Vranesic, “ Fundamentals of Digital Logic with VHDL Design”, McGraw-Hill, ISBN–13:978-1-25-902597-6.
3. Muhammas Mazidi, Janice Mazidi and Rolin McKinlay, “The 8051 Microcontroller and Embedded Systems using Assembly and C”, Pearson Education, ISBN-13: 9788131758991

Reference:

1. John Yarbrough, “Digital Logic applications and Design”, Cengage Learning, ISBN – 13: 978-81-315-0058-3
2. D. Leach, Malvino, Saha, “Digital Principles and Applications”, Tata McGraw Hill, ISBN – 13:978-0-07-014170-4.
3. Anil Maini, “Digital Electronics: Principles and Integrated Circuits”, Wiley India Ltd, ISBN:978-81-265-1466-3.
4. Norman B & Bradley, “Digital Logic Design Principles, Wiley India Ltd, ISBN:978-81-265-1258-4.
5. Scott Mackenzie, “The 8051 Microcontroller”, Prentice Hall India, ISBN-13: 978-0130195623

Savitribai Phule Pune University, Pune Second Year of Computer Engineering (2015 Course) 210243: Data Structures and Algorithms		
Teaching Scheme: TH: 04 Hours/Week	Credit 04	Examination Scheme: In-Sem: 50 Marks End-Sem: 50 Marks
Prerequisites: - FPL I and FPL II		
Course Objectives: <ul style="list-style-type: none"> • To understand the standard and abstract data representation methods. • To acquaint with the structural constraints and advantages in usage of the data. • To understand the memory requirement for various data structures. • To operate on the various structured data. • To understand various data searching and sorting methods with pros and cons. • To understand various algorithmic strategies to approach the problem solution. 		
Course Outcomes: On completion of the course, learner will be able to– <ul style="list-style-type: none"> • To discriminate the usage of various structures in approaching the problem solution. • To design the algorithms to solve the programming problems. • To use effective and efficient data structures in solving various Computer Engineering domain problems. • To analyze the problems to apply suitable algorithm and data structure. 		
Course Contents		
Unit I	Introduction to Algorithm and Data Structures	08 Hours
Algorithms- Problem Solving, Introduction to Algorithms, Characteristics of algorithms, Algorithm design tools: Pseudo code and flowchart, Analysis of Algorithms, Complexity of algorithms- Space complexity, Time complexity, Asymptotic notation- Big-O, Theta and Omega, standard measures of efficiency. Data Structures- Data structure, Abstract Data Types (ADT), Concept of linear and Non-linear, static and dynamic, persistent and ephemeral data structures, and relationship among data, data structure, and algorithm, From Problem to Program. Algorithmic Strategies- Introduction to algorithm design strategies- Divide and Conquer, and Greedy strategy. Recurrence relation - Recurrence Relation, Linear Recurrence Relations, With constant Coefficients, Homogeneous Solutions. Solving recurrence relations		
Unit II	Linear Data Structures Using Sequential Organization	08 Hours
Sequential Organization, Linear Data Structure Using Sequential Organization, Array as an Abstract Data Type, Memory Representation and Address Calculation, Inserting an element		

into an array, Deleting an element, Multidimensional Arrays, Two-dimensional arrays, n-dimensional arrays, Concept of Ordered List, Single Variable Polynomial, Representation using arrays, Polynomial as array of structure, Polynomial addition, Polynomial multiplication, Sparse Matrix, Sparse matrix representation, Sparse matrix addition, Transpose of sparse matrix, String Manipulation Using Array

Unit III	Linked Lists	08 Hours
-----------------	---------------------	-----------------

Concept, Comparison of sequential and linked organizations, Primitive operations, Realization of Linked Lists, Realization of linked list using arrays, Dynamic Memory Management, Linked list using dynamic memory management, Linked List Abstract Data Type, Linked list operations, Head pointer and header node, **Types of linked list**- Linear and circular linked lists, Doubly Linked List and operations, Circular Linked List, Singly circular linked list, Doubly circular linked list, **Polynomial Manipulations** - Polynomial addition, Multiplication of two polynomials using linked list. **Generalized Linked List (GLL)**-concept, representation of polynomial and sets using GLL.

Case Study- Garbage Collection

Unit IV	Stacks	08 Hours
----------------	---------------	-----------------

Concept of Stacks and Queues, **Stacks**- concept, Primitive operations, Stack Abstract Data Type, Representation of Stacks Using Sequential Organization, stack operations, Multiple Stacks, Applications of Stack- Expression Evaluation and Conversion, Polish notation and expression conversion, Need for prefix and postfix expressions, Postfix expression evaluation, **Recursion**- concept, **variants of recursion**- direct, indirect, tail and tree, Linked Stack and Operations, Backtracking algorithmic strategy, use of stack in backtracking.

Case **Study**- 4 queens problem, Android- multiple tasks/multiple activities and back stack.

Unit V	Queues	08 Hours
---------------	---------------	-----------------

Concept, Queue as Abstract Data Type, Realization of Queues Using Arrays , Circular Queue, Advantages of using circular queues, Multi-queues, Deque, Priority Queue, Array implementation of priority queue, Linked Queue and operations.

Case study- Job scheduling

Unit VI	Sorting and Searching	08 Hours
----------------	------------------------------	-----------------

Searching- Search Techniques, Sequential search, variant of sequential search- sentinel search, Binary search, Fibonacci search.

Sorting- Types of sorting-Internal and external sorting, General sort concepts-sort order, stability, efficiency, number of passes, Sorting methods- Bubble sort, Insertion sort, Selection sort, Quick sort, Heap sort, Shell sort, Bucket sort, Radix sort, Comparison of All Sorting Methods.

Books:

Text:

1. Brassard & Bratley, “Fundamentals of Algorithmics”, Prentice Hall India/Pearson Education, ISBN 13-9788120311312.
2. Horowitz and Sahani, “Fundamentals of Data Structures in C++”, University Press, ISBN 10: 0716782928 ISBN 13: 9780716782926.
3. Goodrich, Tamassia, Goldwasser, “Data Structures and Algorithms in C++”, Wiley publication, ISBN-978-81-265-1260-7

Reference:

1. R. Gillberg, B. Forouzn, “Data Structures: A Pseudo code approach with C”, Cenage Learning, ISBN 9788131503140.
2. Horowitz, Sahani and Rajshekaran, “Fundamentals of Computer Algorithms”, University Press, ISBN-13, 9788175152571.
3. Yedidyah Langsam, Moshe J Augenstein, Aron M Tenenbaum, “Data Structures using C and C++”, Pearson Education, ISBN 81-317-0328-2.
4. A Michael Berman, “Data Structures via C++: Objects by Evolution”, Oxford University Press, ISBN:0-19-510843-4.
5. M. Weiss, “Data Structures and Algorithm Analysis in C++”, 2nd edition, Pearson Education, 2002, ISBN-81-7808-670-0.

Savitribai Phule Pune University, Pune Second Year of Computer Engineering (2015 Course) 210244: Computer Organization and Architecture		
Teaching Scheme: TH: 04 Hours/Week	Credit 04	Examination Scheme: In-Sem: 50 Marks End-Sem: 50 Marks
Prerequisites: - Fundamentals of Programming Languages-I & II and Basics of Electronics Engineering		
Course Objectives: <ul style="list-style-type: none"> To understand the structure, function and characteristics of computer systems To understand the design of the various functional units of digital computers To learn basics of Parallel Computer Architecture 		
Course Outcomes: On completion of the course, learner will be able to– <ul style="list-style-type: none"> Demonstrate computer architecture concepts related to design of modern processors, memories and I/Os Analyze the principles of computer architecture using examples drawn from commercially available computers Evaluate various design alternatives in processor organization 		
Course Contents		
Unit I	Computer Evolution and Performance	08 Hours
Unit I Computer Evolution and Performance Computer organization and architecture structure and function, Evolution (a brief history) of computers, Designing for Performance, Evolution of Intel processor architecture-n 4 bit to 64 bit, performance assessment. A top level view of Computer function and interconnection- Computer Components, Computer Function, Interconnection structure, bus interconnection, Computer Arithmetic- The Arithmetic and Logic Unit, addition and subtraction of signed numbers, design of adder and fast adder, carry look ahead addition, multiplication of positive numbers, signed operand multiplication, booths algorithm, fast multiplication, integer division. Floating point representation and operations – IEEE standard, arithmetic operations, guard bits and truncation.		
Unit II	Computer Memory System	08 Hours
Characteristics of memory system, The memory hierarchy. Cache Memory- Cache memory principles, Elements of cache design- cache address, size, mapping functions, replacement algorithms, write policy, line size, number of cache, one level and two level cache, performance characteristics of two level cache- locality & operations.		

<p>Case Study- pentium4 cache organization. Internal Memory- semiconductor main memory, advanced DRAM organization External Memory- Hard Disk organization, RAID- level 1 to level 6</p>		
Unit III	Input and Output System	08 Hours
<p>External devices, I/O modules- Module function and I/O module structure, Programmed I/O- overview, I/O commands, I/O instructions, Interrupt driven I/O- interrupt processing, design issues Case Study- Programmable Interrupt Controller 82C59A Direct Memory Access- drawbacks of programmed and interrupt driven I/O, DMA functions, Case Study- DMA Controller Intel 8237A, I/O channels and processors- evolution and characteristics, The external Interface- Thunderbolt and Infinite Band.</p>		
Unit IV	Instruction Sets	08 Hours
<p>Characteristics and Functions- machine instruction characteristics, types of operands, Case Study-Intel x86, Types of operations- data transfer, arithmetic, logical, conversion, input-output, system control, and transfer of control, Case Study-Intel 8086 operation types. Addressing modes and Formats- Addressing modes- immediate, direct, indirect, register, register indirect, displacement and stack, Case Study-8086 addressing modes, Instruction Formats- instruction length, allocation of bits, variable length instructions. Case Study- 8086 instruction formats.</p>		
Unit V	Processor Organization	08 Hours
<p>Processor organization, Register organization- user visible registers, control and status registers, Case Study- register organization of microprocessor 8086. Instruction Cycle- The indirect cycle and Data flow Instruction Pipelining- Pipelining Strategy, pipeline performance, pipeline hazards, dealing with branches, Case Study- one of Intel 486 or Intel 586 pipelining. Instruction level parallelism and superscalar processors- Super scalar verses super pipelined, constraints, Design Issues- instruction level and machine parallelism, Instruction issue policy, register renaming, machine parallelism, branch prediction, superscalar execution and implementation. Case study- Pentium 4.</p>		
Unit VI	Basic Processing Unit	08 Hours
<p>Fundamental Concepts- register transfer, performing arithmetic or logic operations, fetching a word from memory, storing a word in memory, Execution of a complete instruction- branch instructions, Hardwired control, Micro programmed control- micro instructions, micro program sequencing, wide branch addressing, microinstruction with next address field, pre fetching microinstructions and emulation.</p>		
Books:		

Text:

1. W. Stallings, “Computer Organization and Architecture: Designing for performance”, Pearson Education/ Prentice Hall of India, 2003, ISBN 978-93-325-1870-4, 7th Edition.
2. Zaky S, Hamacher, “Computer Organization”, 5th Edition, McGraw-Hill Publications, 2001, ISBN- 978-1-25-900537-5, 5th Edition.

Reference:

1. John P Hays, “Computer Architecture and Organization”, McGraw-Hill Publication, 1998, ISBN:978-1-25-902856-4, 3rd Edition.
2. Miles Murdocca and Vincent Heuring, “Computer Architecture and Organization- an integrated approach, Wiley India Pvt. Ltd, ISBN:978-81-265-1198-3, 2nd Edition
3. A. Tanenbaum, “Structured Computer Organization”, Prentice Hall of India, 1991 ISBN: 81 – 203 – 1553 – 7. 4th Edition
4. Patterson and Hennessy, “Computer Organization and Design”, Morgan Kaufmann Publishers In, ISBN 978-0-12-374750-1, 4th Edition.

Savitribai Phule Pune University, Pune Second Year of Computer Engineering (2015 Course) 210245: Object Oriented Programming		
Teaching Scheme: TH: 04 Hours/Week	Credit 04	Examination Scheme: In-Sem: 50 Marks End-Sem: 50 Marks
Prerequisites: Fundamentals of Programming Languages-I and II		
Course Objectives: <ul style="list-style-type: none"> • To explore the principles of Object Oriented Programming (OOP). • To understand object-oriented concepts such as data abstraction, encapsulation, inheritance, dynamic binding, and polymorphism. • To use the object-oriented paradigm in program design. • To lay a foundation for advanced programming. • Provide programming insight using OOP constructs. 		
Course Outcomes: On completion of the course, learner will be able to– <ul style="list-style-type: none"> • Analyze the strengths of object oriented programming • Design and apply OOP principles for effective programming • Develop programming application using object oriented programming language C++ • Percept the utility and applicability of OOP 		
Course Contents		
Unit I	Classes and Objects	08 Hours
Need of Object-Oriented Programming (OOP), Object Oriented Programming Paradigm, Basic Concepts of Object-Oriented Programming, Benefits of OOP, C++ as object oriented programming language. C++ Programming- C++ programming Basics, Data Types, Structures, Enumerations, control structures, Arrays and Strings, Class, Object, class and data abstraction, class scope and accessing class members, separating interface from implementation, controlling access to members. Functions- Function, function prototype, accessing function and utility function, Constructors and destructors, Copy Constructor, Objects and Memory requirements, Static Class members, data abstraction and information hiding, inline function		
Unit II	Polymorphism and Inheritance	08 Hours

Operator Overloading- concept of overloading, operator overloading, Overloading Unary Operators, Overloading Binary Operators, Data Conversion, Type casting (implicit and explicit), Pitfalls of Operator Overloading and Conversion, Keywords explicit and mutable.

Inheritance- Base Class and derived Class, protected members, relationship between base Class and derived Class, Constructor and destructor in Derived Class, Overriding Member Functions, Class Hierarchies, Inheritance, Public and Private Inheritance, Levels of Inheritance, Multiple Inheritance, Ambiguity in Multiple Inheritance, Aggregation, Classes Within Classes.

Polymorphism- concept, relationship among objects in inheritance hierarchy, abstract classes, polymorphism.

Unit III	Templates and Exception Handling	08 Hours
-----------------	---	-----------------

Templates- function templates, Function overloading, overloading Function templates, class templates, class template and Nontype parameters, template and inheritance, template and friends Generic Functions, Applying Generic Function, Generic Classes, The typename and export keywords, The Power of Templates.

Exception Handling- Fundamentals, other error handling techniques, simple exception handling- Divide by Zero, rethrowing an exception, exception specifications, processing unexpected exceptions, stack unwinding, constructor, destructor and exception handling, exception and inheritance,

Unit IV	Virtual Functions	08 Hours
----------------	--------------------------	-----------------

Virtual Functions- Pointers- indirection Operators, Memory Management: new and delete, Pointers to Objects, A Linked List Example, accessing Arrays using pointers, Function pointers, Pointers to Pointers, A Parsing Example, Debugging Pointers, Dynamic Pointers, smart pointers, shared pointers, Case Study : Design of Horse Race Simulation

Virtual Function- Friend Functions, Static Functions, Assignment and Copy Initialization, The this Pointer, virtual function, dynamic binding, Virtual destructor,

Unit V	Files and Streams	08 Hours
---------------	--------------------------	-----------------

Data hierarchy, Stream and files, Stream Classes, Stream Errors, Disk File I/O with Streams, File Pointers, and Error Handling in File I/O, File I/O with Member Functions, Overloading the Extraction and Insertion Operators, memory as a Stream Object, Command-Line Arguments, Printer output, Early vs. Late Binding.

Unit VI	Standard Template Library (STL)	08 Hours
----------------	--	-----------------

Introduction to STL, Containers, algorithms and iterators, Containers- Sequence container and associative containers, container adapters, Algorithms- basic searching and sorting algorithms, min-max algorithm, set operations, heap sort, Iterators- input, output, forward, bidirectional and random access.

Object Oriented Programming – a road map to future

Books:**Text:**

1. Bjarne Stroustrup, “The C++ Programming language”, Third edition, Pearson Education. ISBN 9780201889543.
2. **Deitel**, “C++ How to Program”, 4th Edition, Pearson Education, ISBN:81-297-0276-2

Reference:

1. Robert Lafore, “Object-Oriented Programming in C++”, fourth edition, Sams Publishing, ISBN:0672323087 (ISBN 13: 9780672323089)
2. Herbert Schildt, “C++ The complete reference”, Eighth Edition, McGraw Hill Professional, 2011, ISBN:978-00-72226805
3. Matt Weisfeld, “The Object-Oriented Thought Process”, Third Edition Pearson ISBN-13:075-2063330166
4. Cox Brad, Andrew J. Novobilski, “Object –Oriented Programming: An Evolutionary Approach”, Second Edition, Addison–Wesley, ISBN:13:978-020-1548341

<p style="text-align: center;">Savitribai Phule Pune University, Pune Second Year of Computer Engineering (2015 Course) 210246 Digital Electronics Lab</p>		
Teaching Scheme:	Credit	Examination Scheme:
PR: 02 Hours/Week	01	Practical: 50 Marks Term work: 25 Marks
Guidelines for Instructor's Manual		
<p>The instructor's manual is to be developed as a hands-on resource and reference. The instructor's manual need to include prologue (about University/program/ institute/ department/foreword/ preface etc), University syllabus, conduction & Assessment guidelines, topics under consideration- concept, objectives, outcomes, Data sheets of various ICs used to perform assignments, information about downloading and installing 8051 simulator, assignment wise explanation of statements with related theory and all the necessary information which is helpful for students to understand the implementation of assignments and references.</p>		
Guidelines for Student's Lab Journal		
<p>The laboratory assignments are to be submitted by student in the form of journal. Journal consists of prologue, Certificate, table of contents, and handwritten write-up of each assignment (Title, Objectives, Problem Statement, Outcomes, software & Hardware requirements, Date of Completion, Assessment grade/marks and assessor's sign, Theory- Concept, circuit diagram, pin configuration, conclusion/analysis).</p> <p style="background-color: #90EE90;">As a conscious effort and little contribution towards Green IT and environment awareness, attaching printed papers as part of write-ups and program listing to journal may be avoided.</p>		
Guidelines for Lab /TW Assessment		
<p>Continuous assessment of laboratory work is to be done based on overall performance and lab assignments performance of student. Each lab assignment assessment will assign grade/marks based on parameters with appropriate weightage. Suggested parameters for overall assessment as well as each lab assignment assessment include- timely completion, performance, innovation, efficiency, punctuality and neatness.</p>		
Guidelines for Laboratory Conduction		
<p>Student will submit the term work in the form of Journal consisting of minimum of 14 experiments with all experiments from group A and any 4 assignments from group B (at least one must be VHDL assignment) and one from group C assignments.</p>		
Suggested List of Laboratory Assignments		
Sr. No.	Group A (Mandatory)	
1	T.T.L Characteristics (Study and Write up only).	
2	Design and implement Code converters e.g. Excess-3 to BCD and vice versa	

3	Design and implement a system to realize tuning of one FM channel among many.
4	Design and implement Demultiplexers: Applications like Realization of ROM using Demultiplexer.
5	Design and implement BCD adder/Subtractor using 4 bit binary adder 7483.
6	Design and implement a system to detect an error during digital data transmission in a communication system.
7	Design and implement a system using flip-flops, to monitor number of vehicles entering and exiting from a car parking area with maximum capacity of 15 and having separate entry and exit gates.
8	Design and implement Sequence generator using JK flip-flop.
9	Design and implement Sequence detector using JK flip-flop.
Group B (At least 4)	
1	Assume a scenario of a hall where students are entering to attend seminar. Design and implement a system which will increment count if student is entering in the hall and will decrement count if student is exiting the hall. Assume seating capacity of a hall is 63.
2	Assume a scenario of conveyor belt carrying objects to pack in a box. Design and implement a system which will put 'N' objects in a box and then reset the counting to pack next box. You can use ICs 7490 & 74190 (N>10).
3	Design and implement Pseudo random number generator.
4	Design and implement a barrel shifter.
5	Write VHDL code to implement Arithmetic & Logic Unit.
6	Write VHDL code to implement Asynchronous and Synchronous Counter.
Group C (At least One)	
1	Implement simple ASM chart using multiplexer controller method.
2	Implementation of combinational logic using PLAs.
3	Simulate interfacing of LCD with 8051 to display a message. Write program using 8051 assembly language. Use any available 8051 simulator.

Savitribai Phule Pune University, Pune
Second Year of Computer Engineering (2015 Course)
210247: Data Structures Lab

Teaching Scheme	Credit	Examination Scheme
PR: 04 Hours/Week	02	TW: 25 Marks PR: 50 Mark

Guidelines for Instructor's Manual

The instructor's manual is to be developed as a hands-on resource and reference. The instructor's manual need to include prologue (about University/program/ institute/ department/foreword/ preface etc), University syllabus, conduction & Assessment guidelines, topics under consideration-concept, objectives, outcomes, set of typical applications/assignments/ guidelines, and references.

Guidelines for Student Journal

The laboratory assignments are to be submitted by student in the form of journal. Journal consists of prologue, Certificate, table of contents, and **handwritten write-up** of each assignment (Title, Objectives, Problem Statement, Outcomes, software & Hardware requirements, Date of Completion, Assessment grade/marks and assessor's sign, **Theory-Concept in brief, algorithm, flowchart, test cases, conclusion/analysis.**

Program codes with sample output of all performed assignments are to be submitted as softcopy.

As a conscious effort and little contribution towards Green IT and environment awareness, attaching printed papers as part of write-ups and program listing to journal may be avoided. Use of DVD containing students programs maintained by lab In-charge is highly encouraged. For reference one or two journals may be maintained with program prints at Laboratory.

Guidelines for Assessment

Continuous assessment of laboratory work is to be done based on overall performance and lab assignments performance of student. Each lab assignment assessment will assign grade/marks based on parameters with appropriate weightage. Suggested parameters for overall assessment as well as each lab assignment assessment include- timely completion, performance, innovation, efficient codes, punctuality and neatness.

Guidelines for Laboratory Conduction

The instructor is expected to frame the assignments by understanding the prerequisites, technological aspects, utility and recent trends related to the topic. The assignment framing policy need to address the average students and inclusive of an element to attract and promote the intelligent students. The instructor may set multiple sets of assignments and distribute among batches of students. It is appreciated if the assignments are based on real world problems/applications. Encourage students for appropriate use of Hungarian notation, Indentation and comments. Use of open source software is to be encouraged. For all lab assignments instruct students to minimize the number of operations, test against all test cases.

In addition to these, instructor may assign one real life application in the form of a mini-project based on the concepts learned. Instructor may also set one assignment or mini-project that is suitable to respective branch beyond the scope of syllabus.

Set of suggested assignment list is provided in groups- A, B, C, D, and E. Each student must perform at least 13 assignments as at least 3 from group A, 3 from group B, 2 from group C, 2 from group D and 3 from group E.

Operating System recommended :- 64-bit Open source Linux or its derivative

Programming tools recommended: - Open Source C++ Programming tool like G++/GCC

Suggested List of Laboratory Assignments

Group A

- | | |
|----|---|
| 1. | <p>Second year Computer Engineering class, set A of students play cricket and set B of students play badminton. Write C/C++ program to find and display-</p> <ol style="list-style-type: none"> i. Set of students who play either cricket or badminton or both ii. Set of students who play both cricket and badminton iii. Set of students who play only cricket iv. Set of students who play only badminton v. Set of students who play neither cricket nor badminton <p>(Note- While realizing the set duplicate entries are to avoided)</p> |
| 2. | <p>Write C/C++ program to store marks scored for first test of subject 'Data Structures and Algorithms' for N students. Compute</p> <ol style="list-style-type: none"> I. The average score of class ii. Highest score and lowest score of class iii. Marks scored by most of the students iv. list of students who were absent for the test |
| 3. | <p>Department library has N books. Write C/C++ program to store the cost of books in array in ascending order. Books are to be arranged in descending order of their cost. Write function for</p> <ol style="list-style-type: none"> a) Reverse the contents of array without using temporary array. b) Copy costs of books those with cost less than 500 in new array c) Delete the duplicate entries using temporary array d) Delete duplicate entries without using temporary array e) Count number of books with cost more than 500. |
| 4. | <p>It is decided that weekly greetings are to be furnished to wish the students having their birthdays in that week. The consolidated sorted list with desired categorical information is to provide to authority. Write C++ program for array of structures to</p> |

	store students PRNs with date and month of birth. Let Array_A and Array_B be the two arrays for two SE Computer divisions. Arrays are sorted on date and month. Merge these two arrays into third array Array_SE_Comp_DOB resulting in sorted information about Date of Birth of SE Computer students.																									
5.	<p>A magic square is an $n * n$ matrix of the integers 1 to n^2 such that the sum of each row, column, and diagonal is the same. The figure given below is an example of magic square for case $n=5$. In this example, the common sum is 65. Write C/C++ Program for magic square.</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>15</td><td>8</td><td>1</td><td>24</td><td>17</td></tr> <tr><td>16</td><td>14</td><td>7</td><td>5</td><td>23</td></tr> <tr><td>22</td><td>20</td><td>13</td><td>6</td><td>4</td></tr> <tr><td>3</td><td>21</td><td>19</td><td>12</td><td>10</td></tr> <tr><td>9</td><td>2</td><td>25</td><td>18</td><td>11</td></tr> </table>	15	8	1	24	17	16	14	7	5	23	22	20	13	6	4	3	21	19	12	10	9	2	25	18	11
15	8	1	24	17																						
16	14	7	5	23																						
22	20	13	6	4																						
3	21	19	12	10																						
9	2	25	18	11																						
6.	An $m \times n$ matrix is said to have a saddle point if some entry $a[i][j]$ is the smallest value in row i and the largest value in j . Write C/ C++ function that determines the location of a saddle point if one exists.																									
7.	Write C/C++ program for storing matrix. Write functions for <ol style="list-style-type: none"> Check whether given matrix is upper triangular or not Compute summation of diagonal elements Compute transpose of matrix Add, subtract and multiply two matrices 																									
8.	Write C++ program with class for String. Write a function <ul style="list-style-type: none"> <i>frequency</i> that determines the frequency of occurrence of particular character in the string. <i>deletethat</i> accepts two integers, <i>start</i> and <i>length</i>. The function computes a new string that is equivalent to the original string, except that <i>length</i> characters being at <i>start</i> have been removed. <i>chardeletethat</i> accepts a character <i>c</i>. The function returns the string with all occurrences of <i>c</i> removed. <i>replace</i> to make an in-place replacement of a substring <i>w</i> of a string by the string <i>x</i>. note that <i>w</i> may not be of same size of <i>x</i> <i>palindrome</i> to check whether given string is palindrome or not 																									
9.	Write C++ program for sparse matrix realization and operations on it- Transpose, Fast Transpose and addition of two matrices																									
10.	Write a C++ program to realize polynomial equation and perform operations. Write function <ol style="list-style-type: none"> To input and output polynomials represented as $b_m x^{em} + b_{m-1} x^{em-1} + \dots + b_0 x^{e0}$. Your functions should overload the << and >> operators. Evaluates a polynomial at given value of x Add two polynomials Multiplies two polynomials 																									
Group B																										
11.	Department of Computer Engineering has student's club named 'Pinnacle Club'. Students of Second, third and final year students of department can be granted membership on request. Similarly one may cancel the membership of club. The list is sorted on PRN of student. First node is reserved for president of club and last node is reserved for secretary of club. Write C++ program to maintain club member's																									

	<p>information using singly linked list. Store student PRN and Name. Write functions to</p> <ol style="list-style-type: none"> Add and delete the members as well as president or even secretary. Compute total number of members of club Display members Display list in reverse order using recursion Two linked lists exists for two divisions. Concatenate two lists.
12.	<p>The ticket booking system of Cinemax theater has to be implemented using C++ program. There are 10 rows and 7 seats in each row. Doubly circular linked list has to be maintained to keep track of free seats at rows. Assume some random booking to start with. Use array to store pointers (Head pointer) to each row. On demand</p> <ol style="list-style-type: none"> The list of available seats is to be displayed The seats are to be booked The booking can be cancelled.
13.	<p>Write C++ program for storing appointment schedule for day booked randomly. Set start and end time and min and max duration for visit slot such as 20 minutes or 1 hour . Write functions for-</p> <ol style="list-style-type: none"> Display free slots Book appointment Cancel appointment (check validity, time bounds, availability etc) Sort list based on time Sort list based on time using pointer manipulation
14.	<p>Second year Computer Engineering class, set A of students like Vanilla Ice-cream and set B of students like butterscotch ice-cream. Write C/C++ program to store two sets using linked list. compute and display-</p> <ol style="list-style-type: none"> Set of students who like either vanilla or butterscotch or both Set of students who like both vanilla and butterscotch Set of students who like only vanilla not butterscotch Set of students who like only butterscotch not vanilla Set of students who like neither vanilla nor butterscotch
15.	<p>Write C++ program to store set of negative and positive numbers using linked list. Write functions</p> <ol style="list-style-type: none"> Insert numbers Delete nodes with negative numbers To create two more linked lists using this list, one containing all positive numbers and other containing negative numbers For two lists that are sorted; Merge these two lists into third resultant list that is sorted
16.	<p>Write C++ program for storing binary number using doubly linked lists. Write functions</p> <ol style="list-style-type: none"> to compute 1's and 2's complement add two binary numbers
17.	<p>Let $x = (x_1, x_2, \dots, x_n)$ and $y = (y_1, y_2, \dots, y_m)$ be two doubly linked lists. Assume that in each linked list, the nodes are in non-decreasing order of their data-field values. Write an algorithm to merge the two lists to obtain a new linked list z in which the nodes are also in this order. Following the merge, x and y should represent empty lists because each node initially in x or y is now in z. No additional nodes may be used.</p>
18.	<p>Design a linked allocation system to represent and manipulate univariate polynomials with integer coefficients (use circular linked lists with head nodes). Each term of the</p>

polynomial will be represented as a node Thus. a node in this system will have three data members as below:

Exponent	Link
Coefficient	

To erase polynomials efficiently, we need to use an available-space list and associated functions. The external (i.e.. for input or output) representation of a univariate polynomial will be assumed to be a sequence of integers of the form: $n, c_1, e_1, c_2, e_2, c_3, e_3, \dots, c_n, e_n$ where e_i represents an exponent and c_i a coefficient; n gives the number of terms in the polynomial. The exponents are in decreasing order — i.e., $e_1 > e_2 > \dots > e_n$.

Write and test the following functions:

1. **istream&operator >>**(istream& is, Polynomial& x): Read in an input polynomial and convert it to its circular list representation using a head node.
2. **ostream&operator <<** (ostream&os, Polynomial& x): Convert x from its linked list representation to its external representation and output it.
3. **Polynomial:: Polynomial(const Polynomial& a)** [Copy Constructor]: Initialize the polynomial *this to the polynomial a.
4. **const Polynomial& Polynomial :: operator=(const Polynomial& a)** [Assignment Operator]: Assign polynomial a to *this.
5. **Polynomial:: Polynomial ()** [Destructor]: Return all nodes of the polynomial *this to the available-space list.
6. **Polynomial operator+ (const Polynomial& a, const Polynomial& b)** [Addition]: Create and return the polynomial a + b. a and b are to be left unaltered.
7. **Polynomial operator- (const Polynomial& a, const Polynomial& b)** [Subtraction] : Create and return the polynomial a - b. a and b are to be leftunaltered.
8. **Polynomial operator* (constPolynomial& a, constPolynomial& b)** [Multiplication]: Create and return the polynomial a * b. a and b are to be left unaltered.
9. **floatPolynomial ::Evaluate(float x): Evaluate the polynomial *this at x and return the result.**

19. Write C++ program to realize Set using Generalized Liked List (GLL) e.g A ={ a, b, {c, d,e, { }, {f,g}, h, I, {j,k}, l, m}. Store and print as set notation.

Group C

20. A palindrome is a string of character that's the same forward and backward. Typically, punctuation, capitalization, and spaces are ignored. For example, "Poor Dan is in a droop" is a palindrome, as can be seen by examining the characters "poor danisina droop" and observing that they are the same forward and backward. One way to check for a palindrome is to reverse the characters in the string and then compare with them the original-in a palindrome, the sequence will be identical. Write C++ program with functions
1. to check whether given string is palindrome or not that uses a stack to determine whether a string is a palindrome.
 2. to remove spaces and punctuation in string, convert all the Characters to lowercase, and then call above Palindrome checking function to check for a palindrome
 3. to print string in reverse order using stack
21. In any language program mostly syntax error occurs due to unbalancing delimiter such

	as (), {}, []. In a given expression while ($m < (n[8] + 0)$), the first opening parenthesis matched with the last closing parenthesis, but this is done after the second opening parenthesis is matched with the next to last closing parenthesis, this, in turn, is done after the opening square bracket is matched with the closing bracket. Here, identify the data structure needed to implement delimiter matching program to display the given statement is balanced/ well- parenthesis or not. Write C++ program using stack to check whether given expression is well parenthesized or not.
22.	Implement C++ program for expression conversion as infix to postfix and its evaluation using stack based on given conditions i. Operands and operator, both must be single character. ii. Input Postfix expression must be in a desired format. iii. Only '+', '-', '*' and '/' operators are expected.
23.	Implement C++ program for expression conversion as infix to prefix, prefix to postfix, prefix to infix, postfix to infix and postfix to prefix.
24.	A classic problem that can be solved by backtracking is called the Eight Queens problem, which comes from the game of chess. The chess board consist of 64 square arranged in an 8 by 8 grid. The board normally alternates between black and white square, but this is not relevant for the present problem. The queen can move as far as she wants in any direction, as long as she follows a straight line, Vertically, horizontally, or diagonally. Write C++ program with recursive function for generating all possible configurations for 4-queen's problem.
Group D	
25.	Queues are frequently used in computer programming, and a typical example is the creation of a job queue by an operating system. If the operating system does not use priorities, then the jobs are processed in the order they enter the system. Write C++ program for simulating job queue. Write functions to add job and delete job from queue.
26.	In a priority queue, items are added in queue along with a priority, and serves (deletes from queue) the item with the highest priority. If two items have the same priority then they can be deleted from queue in either order. Priority queues are heavily used in computer system software- for example when keeping track of jobs waiting to print on a printer. Implement a priority queue in C++ using an inorder List to store the items in the queue. Create a class that includes the data items(which should be template) and the priority (which should be int)The inorder list should contain these objects ,with operator <= overloaded so that the items with highest priority appear at the beginning of the list (which will make it relatively easy to retrieve the highest item.)
27.	A double-ended queue(deque) is a linear list in which additions and deletions may be made at either end. Obtain a data representation mapping a deque into a one-dimensional array. Write C++ program to simulate deque with functions to add and delete elements from either end of the deque.
28.	Pizza parlor accepting maximum M orders. Orders are served in first come first served basis. Order once placed can not be cancelled. Write C++ program to simulate the system using circular queue using array.
Group E	
29.	a) Write C++ program to store roll numbers of student in array who attended training program in random order. Write function for searching whether particular student attended training program or not using linear search and sentinel search.

	b) Write C++ program to store roll numbers of student array who attended training program in sorted order. Write function for searching whether particular student attended training program or not using binary search.
30.	Write C++ program to store names and mobile numbers of your friends in sorted order on names. Search your friend from list using binary search (recursive and non recursive). Insert friend if not present in phonebook.
31.	Write C++ program to store names and mobile numbers of your friends in sorted order on names. Search your friend from list using Fibonacci search. Insert friend if not present in phonebook.
32.	Write C++ program to maintain club members, store roll numbers in ascending order. Write function 'Ternary_Search' to search whether particular student is member of club or not. Ternary search is modified binary search that divides array into 3 halves instead of two.
33.	Write C++ program to store first year percentage of students in array. Write function for sorting array of floating point numbers in ascending order using a) Selection Sort b) Bubble sort and
34.	Write C++ program to store first year percentage of students in array. Write function for sorting array of floating point numbers in ascending order using a) Insertion sort b) Shell Sort
35.	Write C++ program to store first year percentage of students in array. Write function for sorting array of floating point numbers in ascending order using quick sort
36.	Write C++ program to store first year percentage of students in array. Write function for sorting array of floating point numbers in ascending order using bucket sort
37.	Write C++ program to store first year percentage of students in array. Write function for sorting array of floating point numbers in ascending order using radix sort

Savitribai Phule Pune University, Pune Second Year of Computer Engineering (2015 Course) 210248: Object Oriented Programming Lab		
Teaching Scheme	Credit	Examination Scheme
PR: 02 Hours/Week	01	TW: 25 Marks PR: 50 Mark
Guidelines for Instructor's Manual		
<p>The instructor's manual is to be developed as a hands-on resource and reference. The instructor's manual need to include prologue (about University/program/ institute/ department/foreword/preface etc), University syllabus, conduction & Assessment guidelines, topics under consideration concept objectives, outcomes, set of typical applications/assignments/ guidelines, references.</p>		
Guidelines for Student's Lab Journal		
<p>The laboratory assignments are to be submitted by student in the form of journal. Journal consists of prologue, Certificate, table of contents, and write-up of each assignment (Title, Objectives, Problem Statement, Outcomes, software & Hardware requirements, Date of Completion, Assessment grade/marks and assessor's sign, Theory- Concept, algorithm, flowchart, test cases, conclusion/analysis).</p> <p><u>Program codes with sample output of all performed assignments are to be submitted as softcopy.</u></p> <p>As a conscious effort and little contribution towards Green IT and environment awareness, attaching printed papers as part of write-ups and program listing to journal may be avoided. Use of DVD containing students programs maintained by lab In-charge is highly encouraged. For reference one or two journals may be maintained with program prints at Laboratory.</p>		
Guidelines for Lab /TW Assessment		
<p>Continuous assessment of laboratory work is to be done based on overall performance and lab assignments performance of student. Each lab assignment assessment will assign grade/marks based on parameters with appropriate weightage. Suggested parameters for overall assessment as well as each lab assignment assessment include- timely completion, performance, innovation, efficient codes, punctuality and neatness.</p>		

Guidelines for Laboratory Conduction

The instructor is expected to frame the assignments by understanding the prerequisites, technological aspects, utility and recent trends related to the topic. The assignment framing policy need to address the average students and inclusive of an element to attract and promote the intelligent students. The instructor may set multiple sets of assignments and distribute among batches of students. Encourage students for the use of industry coding standards such as appropriate use of Hungarian notation, Indentation and comments.

Use of open source software is encouraged. In addition to these, instructor may assign one real life application in the form of a mini-project. Based on the concepts learned. Instructor may also set one assignment or mini-project that is suitable to respective branch beyond the scope of syllabus.

Operating System recommended :- 64-bit Open source Linux or its derivative

Programming tools recommended: - Open Source C++ Programming tool like G++/GCC.

First assignment is compulsory. Set of suggested assignment list is provided in 3 groups- A, B, and C. Instructor is suggested to design assignments list by selecting/designing at least **10** suitable assignments from group A, B, and C along with **compulsory assignment**. Select at least **5** from group A, **4** from group B, **1** from group C.

Suggested List of Laboratory Assignments

Compulsory Assignment

- | | |
|-----------|---|
| 1. | Install and Configure 64 bit Linux Operating Systems, |
|-----------|---|

Group A

- | | |
|-----------|--|
| 2. | <p>Implement a class Complex which represents the Complex Number data type. Implement the following operations:</p> <ol style="list-style-type: none"> 1. A constructor (including a default constructor which creates the complex number 0+0i). 2. Overloaded operator+ to add two complex numbers. 3. Overloaded operator* to multiply two complex numbers. 4. Overloaded << and >> to print and read Complex Numbers. To do this, you will need to decide what you want your input and output format to look like. |
| 3. | <p>Implement a class Quadratic that represents two degree polynomials i.e., polynomials of type ax^2+bx+c. Your class will require three data members corresponding to a, b and c. Implement the following operations:</p> <ol style="list-style-type: none"> 1. A constructor (including a default constructor which creates the 0 polynomial). 2. Overloaded operator+ to add two polynomials of degree 2. 3. Overloaded << and >> to print and read polynomials. To do this, you will need to decide what you want your input and output format to look like. 4. A function eval that computes the value of a polynomial for a given value of x. 5. A function that computes the two solutions of the equation $ax^2+bx+c=0$. |

4.	<p>Implement a class CppArray which is identical to a one-dimensional C++ array (i.e., the index set is a set of consecutive integers starting at 0) except for the following :</p> <ol style="list-style-type: none"> 1. It performs range checking. 2. It allows one to be assigned to another array through the use of the assignment operator (e.g. cp1= cp2) 3. It supports a function that returns the size of the array. 4. It allows the reading or printing of array through the use of cout and cin.
5.	<p>Write a Program to create four-function calculator using C++. The program should ask the user to enter a number, an operator, and another number. (Use floating point.) It should then carry out the specified arithmetical operation: adding, subtracting, multiplying, or dividing the two numbers. Use a switch statement to select the operation. Finally, display the result. When it finishes the calculation, the program should ask whether the user wants to do another calculation. The response can be 'y' or 'n'. Some sample interaction with the program might look like this:</p> <p><i>Enter first number, operator, second number: 10 / 3</i> <i>Answer = 3.333333</i> <i>Do another (y/n)? y</i> <i>Enter first number, operator, second number: 12 + 100</i> <i>Answer = 112</i> <i>Do another (y/n)? n</i></p>
6.	<p>Create student database with appropriate data members that should use the following features of object oriented programming in C++. Class, Object, array of objects, new, delete, default constructor to initialize student class fields, parameterized constructor to set the values into the objects, access specifiers, this pointer.</p>
7.	<p>Implement C++ program to write a class template to represent a generic vector. Include following member functions:</p> <ul style="list-style-type: none"> • To create the vector. • To modify the value of a given element • To multiply by a scalar value • To display the vector in the form (10,20,30,...)
8.	<p>Create a class RationalNumber (fractions) with the following capabilities:</p> <ol style="list-style-type: none"> a) Create a constructor that prevents a 0 denominator in a fraction, reduces or simplifies fractions that are not in reduced form and avoids negative denominators. b) Overload the addition, subtraction, multiplication and division operators for this class. c) Overload the relational and equality operators for this class.
9.	<p>Imagine a publishing company that markets both book and audiocassette versions of its works. Create a class publication that stores the title (a string) and price (type float) of a publication. From this class derive two classes: book, which adds a page count (type int), and tape, which adds a playing time in minutes (type float). Each of these three classes should have a getdata() function to get its data from the user at the keyboard, and a putdata() function to display its data.</p> <p>Write a main() program to test the book and tape classes by creating instances of them, asking the user to fill in data with getdata(), and then displaying the data with putdata().</p>

10.	<p>Write a function in C++ to count and display the number of lines not starting with alphabet 'A' present in a text file "STORY.TXT".</p> <p>Example: If the file "STORY.TXT" contains the following lines,</p> <p style="padding-left: 40px;">The roses are red. A girl is playing there. There is a playground. An aeroplane is in the sky. Numbers are not allowed in the password. The function should display the output as 3.</p>
11.	<p>Write C++ Program with base class convert declares two variables, val1 and val2, which hold the initial and converted values, respectively. It also defines the functions getinit() and getconv(), which return the initial value and the converted value. These elements of convert are fixed and applicable to all derived classes that will inherit convert. However, the function that will actually perform the conversion, compute(), is a pure virtual function that must be defined by the classes derived from convert. The specific nature of compute() will be determined by what type of conversion is taking place.</p>
Group B	
12.	<p>Crete User defined exception to check the following conditions and throw the exception if the criterion does not met.</p> <ol style="list-style-type: none"> a. User has age between 18 and 55 b. User stays has income between Rs. 50,000 – Rs. 1,00,000 per month c. User stays in Pune/ Mumbai/ Bangalore / Chennai d. User has 4-wheeler <p>Accept age, Income, City, Vehicle from the user and check for the conditions mentioned above. If any of the condition not met then throw the exception.</p>
13.	<p>Write a menu driven program that will create a data file containing the list of telephone numbers in the following form</p> <p style="padding-left: 40px;">John 23456 Ahmed 9876</p> <p>Use a class object to store each set of data, access the file created and implement the following tasks</p> <ol style="list-style-type: none"> I. Determine the telephone number of specified person II. Determine the name if telephone number is known III. Update the telephone number, whenever there is a change.
14.	<p>Write a C++ program that creates an output file, writes information to it, closes the file and open it again as an input file and read the information from the file.</p>
15.	<p>Write a C++ program using command line arguments to search for a word in a file and replace it with the specified word. The usage of the program is shown below.</p> <p><i>\$ change <old word> <new word> <file name></i></p>
16.	<p>Using standard template library (STL) list container implement following member functions of list class: empty, insert, merge, reverse, sort, Unique, using iterator</p>

17.	Write a function template selectionSort. Write a program that inputs, sorts and outputs an int array and a float array.																								
18.	<p>You are the owner of a hardware store and need to keep an inventory that can tell you what different tools you have, how many of each you have on hand and the cost of each one. Write a program that initializes the random-access file hardware.dat to 100 empty records, lets you input the data concerning each tool, enables you to list all your tools, lets you delete a record for a tool that you no longer have and lets you update any information in the file. The tool identification number should be the record number. Use the following information to start your file:</p> <table border="1"> <thead> <tr> <th>Record #</th> <th>Tool name</th> <th>Quantity</th> <th>Cost</th> </tr> </thead> <tbody> <tr> <td>3</td> <td>Electric sander</td> <td>7</td> <td>57.98</td> </tr> <tr> <td>17</td> <td>Hammer</td> <td>76</td> <td>11.99</td> </tr> <tr> <td>24</td> <td>Jig saw</td> <td>21</td> <td>11.00</td> </tr> <tr> <td>39</td> <td>Lawn mower</td> <td>3</td> <td>79.50</td> </tr> <tr> <td>56</td> <td>Power saw</td> <td>18</td> <td>99.99</td> </tr> </tbody> </table>	Record #	Tool name	Quantity	Cost	3	Electric sander	7	57.98	17	Hammer	76	11.99	24	Jig saw	21	11.00	39	Lawn mower	3	79.50	56	Power saw	18	99.99
Record #	Tool name	Quantity	Cost																						
3	Electric sander	7	57.98																						
17	Hammer	76	11.99																						
24	Jig saw	21	11.00																						
39	Lawn mower	3	79.50																						
56	Power saw	18	99.99																						
Group C																									
19.	Design and develop the Tic-Tac-Toe Game using C++																								
20.	<p>Develop a Supermarket Billing System in C++ to demonstrate the use OOPS concepts and file handling. The key features of this application are listed below :</p> <ul style="list-style-type: none"> • Bill Report : It shows the bill report of all the items added in supermarket billing system. • Add, Remove or Edit items: With this feature one can add, remove and modify item details. In add items, one can add information or details such as item no., item name, manufacturing date, price, quantity, tax percent, and many more. • Show item details: This feature allows users to see the items and the corresponding details given for the item while adding the item. 																								
21.	Design an E-mail Verifier which accepts the email address from the user. Depending upon the input displays appropriate results. Check for all possibilities. Use the following concepts in the Project – Constructor, Destructor, new, delete, exceptional handling, string handling functions, etc.																								
22.	Design and Develop Library Management system using OOP Concepts.																								

Savitribai Phule Pune University, Pune Second Year of Computer Engineering (2015 Course) 210249: Soft Skills		
Teaching Scheme:	Credit	Examination Scheme:
PR: 02 Hours /Week	01	TW: 25 Marks
Course Objectives: <ul style="list-style-type: none"> • To encourage the all round development of students by focusing on soft skills. • To make the engineering students aware of the importance, the role and the content of soft skills through instruction, knowledge acquisition, demonstration and practice. • To develop and nurture the soft skills of the students through individual and group activities. • To expose students to right attitudinal and behavioral aspects and to build the same through activities 		
Course Outcomes: On completion of the course, learner will be able to– <ul style="list-style-type: none"> • Effectively communicate through verbal/oral communication and improve his listening skill. • Write precise briefs or reports and technical documents. • Actively participate in group discussion / meetings / interviews and prepare & deliver presentations. • Follow Ethics as an engineering professional and adopt good standards & norms of engineering practice. • Become more effective individual through goal/target setting, self motivation and practicing creative thinking. • Function effectively in multi-disciplinary teams through the knowledge of team work, Inter-personal relationships, conflict management and leadership quality. 		
Course Contents		
Unit I	Self Development	
Introduction to soft skills, Self-Management: Self-Evaluation, Self-Discipline, Self-Criticism, Self-Awareness, Self-Esteem, Positive Thinking, Perceptions and Attitudes, Values and Belief Systems, Personal success factors, Handling failure, Knowing Yourself, identifying one's strengths and weaknesses, SWOT analysis, Johari's Window, Career Planning & Goal setting, prioritization, Managing self – emotions, ego, pride, stress; Personality development.		
Unit II	Communication Skills	

Significance of Communication- types, barriers of communication, effective communication, Verbal and non-verbal Communication, Speaking Skills – Importance of speaking effectively, speech process, message, audience, speech.

style, feedback, conversation and oral skills, fluency and self expression, body language phonetics and spoken English, speaking techniques, word stress, correct stress patterns, voice quality, correct tone, types of tones, positive image projection techniques, Public Speaking, Group discussion, Listening Skills: Virtues of Listening, Barriers and filters, Fundamentals of Good Listening, Reading Skills : Comprehension, reading research papers, Communication in a Digital World

Unit III

Language and Writing Skills

Vocabulary: One - Word Substitutes, Words often Confused - Pairs of Words, Synonyms and Antonyms, Foreign Phrases, Phrasal verbs derived from the dynamic verbs, Business Writing: Note Making, Letter writing, Writing Formal Letters. Technical Report Writing, Memo, Notices/Circulars Agenda and Minutes of a Meeting, E-Mail, Essay writing. Employment Communication: Job Application, Preparation of C.V and Resume writing. Presentation skills: Professional Presentation, Nature of Oral Presentation, Planning a Presentation, Preparing the Presentation, Delivering the Presentation.

Unit IV

Leadership and Team Building

Introduction, Leader and Leadership, Leadership Traits, Culture and Leadership : Salient Features of Corporate Culture, Leadership Styles ,Leadership Trends, Team Building: Team Development Stages, Types of Teams : Cross-functional Team, Problem-solving Team, Inter-personal relations: Types of feelings, steps to deal with complex feelings. Assertiveness and Confidence building. Types of Conflict and resolutions. Emotions, emotional empathy and emotional intelligence

Unit V

Stress and Time Management

Introduction, Stress in Today's Time: Identify the Stress Source, Signs of Stress, Ways to Cope with Stress : Healthier Ways to Combat Stress, Steps to be Taken in the Organizations : Open communication, Time Management, Working towards Your Goals, Smart Work, Prioritize your Tasks, 4 Ds of Email Decision Making

Unit VI

Ethics, Etiquette and Mannerism

Professional Etiquette: Etiquette at Meetings, Etiquette at Dining. Involuntary Awkward Actions, Public Relations Office(PRO)'s Etiquettes, Technology Etiquette : Phone Etiquette, Email Etiquette, Social Media Etiquette, Video Conferencing Etiquette, Interview Etiquette, Dressing Etiquettes : for Interview, offices and social functions, Ethical Values: Importance of Work Ethics, Problems in the Absence of Work Ethics.

Books:

Text:

1. Gajendra Singh Chauhan, Sangeeta Sharma : Soft Skills – An Integrated Approach to Maximise Personality, WILEY INDIA, ISBN:13:9788126556397.

Reference:

1. Indrajit Bhattacharya, An Approach to Communication Skills, Delhi, Dhanpat Rai, 2008.
2. Simon Sweeney ,English for Business Communication ,Cambridge University Press, ISBN 13:978-0521754507.
3. Sanjay Kumar and Pushpa Lata, Communication Skills, Oxford University Press, ISBN 10:9780199457069.
4. Atkinson and Hilgard's, Introduction to Psychology, 14th Edition, Geoffrey Loftus, ISBN-10:0155050699 © 2003
5. Kenneth G. Mcgee, Heads Up: How to Anticipate Business Surprises & Seize Opportunities First, Harvard Business School Press, Boston, Massachusetts, 2004, ISBN 10:1591392993.
6. Ananta Kumar Giri, Values, Ethics and Business: Challenges for Education and Management, Jaipur/New Delhi: Rawat Publications, ISBN: 9781591392996.
7. Krishnaswami, N. and Sriraman, T., Creative English for Communication, Macmillan.

Guidelines for Instructor's Manual

The instructor's manual is to be developed as a hands-on resource and reference. The instructor's manual need to include prologue (about University/program/ institute/ department/foreword/preface etc), University syllabus, conduction & Assessment guidelines, topics under consideration concept objectives, outcomes, guidelines, references.

Guidelines for Student's Lab Journal

The laboratory assignments are to be submitted by student in the form of journal.

Guidelines for Lab /TW Assessment

Continuous assessment of laboratory work is to be done based on overall performance and lab assignments performance of student. Each lab assignment assessment will assign grade/marks based on parameters with appropriate weightage. Suggested parameters for overall assessment as well as each lab assignment assessment include- timely completion, performance, innovation, efficiency, punctuality and neatness.

Guidelines for Laboratory Conduction	
Suggested List of Laboratory Assignments	
1.	<p>SWOT analysis The students should be made aware of their goals, strengths and weaknesses, attitude, moral values, self confidence, etiquettes, non-verbal skills, achievements etc. through this activity. SWOT Analysis, Confidence improvement, values, positive attitude, positive thinking and self esteem. The teacher should prepare a questionnaire which evaluate students in all the above areas and make them aware about these aspects.</p>
2.	<p>Personal & Career Goal setting – Short term & Long term The teacher should explain to them on how to set goals</p>
3.	<p>Public Speaking Any one of the following activities may be conducted :</p> <p>1. Prepared speech (Topics are given in advance, students get 10 minutes to prepare the speech and 5 minutes to deliver.)</p> <p>2. Extempore speech (Students deliver speeches spontaneously for 5 minutes each on a given topic)</p> <p>3. Story telling (Each student narrates a fictional or real life story for 5 minutes each)</p> <p>4. Oral review (Each student orally presents a review on a story or a book read by them)</p>
4.	<p>Reading and Listening skills The batch can be divided into pairs. Each pair will be given an article (any topic) by the teacher. Each pair would come on the stage and read aloud the article one by one. After reading by each pair, the other students will be asked questions on the article by the readers. Students will get marks for correct answers and also for their reading skills. This will evaluate their reading and listening skills. The teacher should give them guidelines on improving their reading and listening skills. The teacher should also give passages on various topics to students for evaluating their reading comprehension.</p>
5.	<p>Group discussion Group discussions could be done for groups of 5-8 students at a time Two rounds of a GD for each group should be conducted and teacher should give them feedback .</p>
6.	<p>Letter/Application writing Each student will write one formal letter, and one application. The teacher should teach the students how to write the letter and application. The teacher should give proper format and layouts.</p>

7.	Report writing The teacher should teach the students how to write report .The teacher should give proper format and layouts. Each student will write one report based on visit / project / business proposal etc.
8.	Resume writing
9.	Presentation Skill Students should make a presentation on any informative topic of their choice. The topic may be technical or non-technical. The teacher should guide them on effective presentation skills. Each student should make a presentation for at least 10 minutes.
10.	Team games for team building
11.	Situational games for role playing as leaders
12.	Faculty may arrange one or more sessions from following: Yoga and meditation. Stress management, relaxation exercises, and fitness exercises. Time management and personal planning sessions.
13.	Mock interviews

Savitribai Phule Pune University, Pune
Second Year of Computer Engineering (2015 Course)
210250: Audit Course 1

Preamble:

In addition to credits, it is recommended that there should be audit courses in preferably in each semester. Student will be awarded the bachelor's degree if he/she earns 190 credits and clears all the audit courses specified in the syllabus. The student will be awarded grade as AP. List of audit courses offered is provided below. Each student has to choose one audit course from the list. Further for each audit course - course objectives, course outcomes, outline of contents, Suggested method of conduction, method of assessment and references is provided.

Criteria-**AP Grade - 'Passed Audit Course'**

The student registered for audit course shall be awarded the grade AP and shall be included such AP grade in the Semester grade report for that course, provided student has the minimum attendance as prescribed by the Savitribai Phule Pune University and satisfactory in-semester performance and secured a passing grade in that audit course. No grade points are associated with this 'AP' grade and performance in these courses is not accounted in the calculation of the performance indices SGPA and CGPA. Evaluation of audit course will be done at institute level itself. (Ref- http://www.unipune.ac.in/Syllabi_PDF/revise-2015/engineering/UG_RULE_REGULATIONS_FOR_CREDIT_SYSTEM-2015_18June.pdf)

Guidelines for Conduction (Any one or more of following but not limited to)

- Lectures/ Guest Lectures
- Visits (Social/Field) and reports
- Demonstrations
- Assist authorities like RTO for audits (e.g. Particular road safety audit as critical on-site assessment of the shortcomings in the various elements of the road) and report
- Mini Project
- Hands on experience on specific focused topic

Guidelines for Assessment(Any one or more of following but not limited to)

- Written Test
- Demonstrations/ Practical Test
- Presentations
- IPR/Publication (IPR/Paper/book/chapter)
- Report

210250 Audit Course 1 Options	
Audit Course Code	Audit Course Title
AC1-I	Road Safety
AC1-II	Humanities and Social Sciences
AC1-III	Environmental Studies
AC1-IV	Engineering Economics
AC1-V	Professional Ethics and Morals
AC1-VI	Intellectual Property Rights and Patents
210250 Audit Course 1 AC1-I: Road Safety	
<p>Road transport remains the least safe mode of transport, with road accidents representing the main cause of death of people. The boom in the vehicle population without adequate road infrastructure, poor attention to driver training and unsatisfactory regulation has been responsible for increase in the number of accidents. India's vehicle population is negligible as compared to the World statistics; but the comparable proportion for accidents is substantially large.</p> <p>The need for stricter enforcement of law to ensure greater safety on roads and an environment-friendly road transport operation is of paramount importance. Safety and security are growing concerns for businesses, governments and the traveling public around the world, as also in India. It is, therefore, essential to take new initiatives in raising awareness, skill and knowledge of students as one of the ibid stake holders who are expected to follow the rules and policies of the government in order to facilitate safety of individual and safe mobility of others.</p>	
Course Contents	
<ol style="list-style-type: none"> 1. Existing Road Transport Scenario 2. Accident Causes & Remedies 3. Road Accident Investigation & Investigation Methods 4. Vehicle Technology – CMVR & Road Safety 5. Regulatory / Legislative Provisions for Improving Road Safety 6. Behavioral Training for Drivers for Improving Road Safety 7. Road Safety Education 8. Road Engineering Measures for Improving Road Safety 	

Reference:

1. “Road Accidents in India Issues & Dimensions”, Ministry of Road Transport & Highways Government of India (www.unescap.org/sites/default/files/2.12.India_.pdf)
2. “Road Safety in India- Insights and analysis”, http://indiatransportportal.com/wp-content/uploads/2012/11/Road_safety_2012.pdf
3. Road User’s Handbook, ROADS & MARITIME PUBLICATIONS
4. “Road Safety and Traffic Management”, The Committee Report, Planning Commission, Government of India (http://planningcommission.nic.in/sectors/ppp_report/3.Reports%20of%20Committees%20&%20Task%20force/Power/14.Road_Safety.pdf)
5. Road Safety Annual Reports, http://www.oecd-ilibrary.org/transport/road-safety-annual-report_23124571
6. “Improving Road Safety in Developing Countries”, The national Academic Press

210250 Audit Course 1**AC1-II: Humanities and Social Sciences**

Objective of Humanities and Social Science(HSS) is to produce well-rounded engineers, not only having good technological skills but also with the ability to interact with different organs of an organization.

HSS is concerned with society and the relationships among individuals within a society. It in turn has many branches, each of which is considered a "social science". The main social sciences include economics, political science, human geography, demography and sociology. In a wider sense, social science also includes some fields in the humanities such as anthropology, archaeology, psychology, history, law and linguistics.

Course Objectives:

- Human and social development;
- Contemporary national and international affairs;
- Emergence of Indian society and Economics
- Sectorial development and Economic development and related issues (such as international economics, WTO, RBI, etc).

Course Outcomes:

On completion of the course, learner will be able to–

- Making engineering and technology students aware of the various issues concerning man and society.
- These issues will help to sensitize students to be broader towards the social, cultural, economic and human issues, involved in social changes
- Able to understand the nature of the individual and the relationship between the self and the community
- Understanding major ideas, values, beliefs, and experiences that have shaped human history and cultures

Course Contents

1. **Indian Society :** Structure of Indian Society, Indian Social Demography– Social and Cultural, Differentiations: caste, class, gender and tribe; Institutions of marriage, family and kinship- Secularization –Social Movements and Regionalism- Panchayatraj Institutions; Indian constitution; Affirmative Action Programme of the Government- various reservations and commissions.
2. **Social Development:** Scientific approach to the study of human beings. Evolution of human kind, social change and evolution. Industrial revolution. National policy on education, health and health care and human development.
3. **Sectoral Development: Agriculture:** Technology changes, Green revolutions, Employment Rural & Urban, Government Schemes. Industrial Development: Strategies, Public & Private Sectors, Categories, infrastructure, transport & communication, Consumer Awareness.
4. **Environment & Ecology:** Ecosystems: Structure, Working, components. Pollution: Water & Air Pollution, Global Warming, Control Strategies, International Treaties. Energy Sources: Renewable & Non Renewable, Hydro power, Biomass, Ocean, Geothermal & Tidal. Global Environmental Issues: Population Growth, Soil Degradation, Loss of Biodiversity.

References:

1. Krugman, “ International Economics”, Pearson Education, ISBN-13:000-01334-23646
2. Prakash, “The Indian Economy”, Pearson Education, ISBN-8131758931
3. Thursen Gerald, “Engineering Economics”, Prentice Hall, ISBN-10:0138221227
4. C.S. Rao, “Environmental Pollution Control Engineering”, New Age International Pvt. Ltd, ISBN-812241835X
5. Rangarajan, “Environmental Issues in India, Pearson Education”, ISBN-10:8131708101
6. University of Delhi, “The Individual & Society”, Pearson Education. ISBN-8131704173
7. Wikipedia.org / wiki /social studies.
8. M. N. Srinivas, “Social change in modern India, 1991”, Orient Longman, ISBN-10:812500422X
9. David Mandelbaum, “Society in Indi”a, 1990, Popular, ISBN-10:8171540139
10. David Newman,”Exporing the architecture of everyday life”, Pine Forge Press, 7th edition, ISBN-10:1452275947

210250 Audit Course 1 **AC1-III: Environmental Studies**

Environmental studies are the field that examines this relationship between people and the environment. An environmental study is an interdisciplinary subject examining the interplay between the social, legal, management, and scientific aspects of environmental issues.

Course Objectives:

- Understanding the importance of ecological balance for sustainable development.
- Understanding the impacts of developmental activities and mitigation measures.
- Understand and realize the multi-disciplinary nature of the environment, its components, and inter-relationship between man and environment
- Understand the relevance and importance of the natural resources in the sustenance of life on earth and living standard

Course Outcomes:

On completion of the course, learner will be able to–

- Comprehend the importance of ecosystem and biodiversity
- To correlate the human population growth and its trend to the environmental degradation and develop the awareness about his/her role towards environmental protection and prevention
- Identify different types of environmental pollution and control measures
- To correlate the exploitation and utilization of conventional and non-conventional resources

Course Contents

1. **Natural Resources:** Introduction, Renewable and non-renewable, Forest, water, mineral, food, energy and land resources, Individual and conservation of resources, Equitable use of resources.
2. **Ecosystems:** Concept, Structure, Function, Energy flow, Ecological succession, Forest, grassland, desert and aquatic ecosystems - Introduction, characteristic features, structure and function.
3. **Biodiversity:** Genetic, Species and ecological diversity, Biogeographical classification of India, Value and hot spots, Biodiversity at global, national and local levels, India as mega-biodiversity nation, Threats to biodiversity, Endangered and endemic species of India, Conservation of Biodiversity, Endangered and endemic species, Conservation of biodiversity.
4. **Pollution:** Definition, Causes, effects and control measures of the pollution – Air, soil, Noise, Water, Marine and Thermal and Nuclear Pollution, Solid waste management, Role of Individual in Prevention of Pollution, Pollution case studies, Disaster management

Reference:

1. Bharucha, E., “Textbook of Environmental Studies”, Universities Press (2005), ISBN-10:8173715408
2. Chapman, J.L. and Reiss, M.J., “Ecology- Principles and Application”, Cambridge University Press (LPE) (1999), ISBN-9780521588027
3. Joseph, B., “Environmental Studies”, Tata McGraw-Hill (2006), ISBN-0070590923
4. Miller, G.T., “Environmental Science- Working with Earth”, Thomson (2006), ISBN-13:978-0-495-55671-8
5. Wright, R.T., “Environmental Science-Towards a sustainable Future”, Prentice Hall (2008), ISBN-13:978-0130936547, 9th Edition.

210250 Audit Course 1
AC1-IV : Engineering Economics

Course Objective:

- Understand the fundamental concepts of engineering economy

Course Outcomes:

On completion of the course, learner will be able to–

- Analysis of current economic parameters /indicators/ policy debates.
- Carry out and evaluate benefit/cost, life cycle and Breakeven analyses on one or more economic alternatives.
- Perform and evaluate present worth, future worth and annual worth analyses on one of more economic alternatives.
- Perform and evaluate payback period and capitalized Cost on one or more economic alternatives

Course Contents

- 1. Basic Principles and Methodology of Economics: Demand/Supply** – elasticity – Government Policies and Application. Theory of the Firm and Market Structure. Basic Macro-economic Concepts (including GDP/GNP/NI/Disposable Income) and Identities for both closed and open economies. Aggregate demand and Supply (IS/LM). Price Indices (WPI/CPI), Interest rates, Direct and Indirect Taxes.
- 2. Public Sector Economics:** Welfare, Externalities, Labour Market. Components of Monetary and Financial System, Central Bank –Monetary Aggregates; Commercial Banks & their functions; Capital and Debt Markets. Monetary and Fiscal Policy Tools & their impact on the economy – Inflation and Phillips Curve.
- 3. Elements of Business/Managerial Economics and forms of organizations:** Cost & Cost Control –Techniques, Types of Costs, Budgets, Break even Analysis, Capital Budgeting, Application of Linear Programming. Investment Analysis – NPV, ROI, IRR, Payback Period, Depreciation, Time value of money. Business Forecasting – Elementary techniques. Statements – Cash flow, Financial. Case Study Method.
- 4. Indian economy Brief overview of post independence period Brief overview of post independence period** – plans. Post reform Growth, Structure of productive activity. Issues of Inclusion – Sectors, States/Regions, Groups of people (M/F), Urbanization. Employment–Informal, Organized, Unorganized, 22 Public, Private. Challenges and Policy Debates in Monetary, Fiscal, Social, External sectors

References-

1. Mankiw Gregory N.,” Principles of Economics”, Thompson Asia, 2002, ISBN-10:0538477229
2. V. Mote, S. Paul, G. Gupta, “Managerial Economics”, Tata McGraw Hill, 2004 .
3. Misra, S.K. and Puri (2009), Indian Economy, Himalaya, ISBN-10:8126115394
4. Pareek Saroj (2003), “Textbook of Business Economics”, Sunrise Publishers, ISBN-1841124397.

210250 Audit Course 1

AC1-V: Professional Ethics and Morals

Professional ethics encompass the personal, organizational and corporate standards of behaviour expected of professionals. This course introduces the concepts, theory and practice of engineering ethics. It will allow students to explore the relationship between ethics and engineering and apply classical moral theory and decision making to engineering issues encountered in academic and professional careers. Professionals and those working in acknowledged professions, exercise specialist knowledge and skill. How the use of this knowledge should be governed when providing a service to the public can be considered a moral issue and is termed professional ethics.

Our society places a great deal of responsibility on its professionals and requires that they conduct themselves in a manner fitting to the place of prominence accorded to them by the community. Studying and understanding professional ethics is as much a part of your development as an engineer as is the study of higher order mathematics. You must be able to broaden your mind and be open to society’s ever changing character. It is important that you learn to share ideas and concepts regardless of the fact that you may not always agree.

Course Objectives:

- An understanding of their duties and responsibilities as professionals through gaining knowledge of the philosophies of ethics, professional practice, and world culture.
- Basic knowledge to make informed ethical decisions when confronted with problems in the working environment.
- An understanding of how a societal moral varies with culture and how this influences ethical thought and action.
- Understand what morality is and how it connects to professional ethics

Course Outcomes:

On completion of the course, learner will be able to–

- Understanding of professional and ethical responsibility.
- Ability to communicate effectively ethical issues
- Understand the nature of professional responsibility and be able to identify the ethical elements in decisions
- Develop a professional ethical identity to carry forward in their life

Course Contents

- 1. What is profession?** - Engineering and Professionalism - Two models of Professionalism - Three Types of Ethics or Morality – The Negative face of Engineering Ethics - The Positive Face of Engineering Ethics - Responsibility in Engineering - Engineering Standards - The Standard Care – Blame-Responsibility and causation.
- 2. Engineering Ethics** – Variety of moral issues – types of inquiry moral dilemmas – moral autonomy – The problems of Many Hands – Kohlburg’s theory – Gilligan’s theory Impediments to Responsible Action.
- 3. Engineering as social experimentation** – Framing the problem – Determining the facts codes of ethics – clarifying Concepts – Application issues – Common Ground – General principles – Utilitarian thinking respect for persons.
- 4. Engineer’s Responsibility for Safety** – Social and Value dimensions of Technology – Technology Pessimism – The Perils of Technological Optimism – The Promise of Technology – Computer Technology Privacy and Social Policy – Risk Benefit Analysis – Collegiality and loyalty.

References:

1. Mike Martin and Roland Schinzinger, "Ethics in Engineering" McGraw Hill, ISBN-10:0072831154
2. Charles E Harris, Micheal J Rabins, "Engineering Ethics, Cengage Learning", ISBN-13:978-1133934684.
3. Edmund G Seebauer and Robert L Barry, "Fundamentals of Ethics for Scientists and Engineers", Oxford University Press, ISBN-10:0195134885
4. PSR Murthy, "Indian Culture Values and Professional Ethics", BS Publications, ISBN-10:9381075700.
5. Caroline Whitback Ethics in Engineering Practice and Research, Cambridgs University Press, ISBN:9780521723985
6. Mike Martin and Roland Schinzinger, "Ethics in Engineering" McGraw Hill, ISBN:978-0-07-248311-6
7. Charles D Fleddermann, "Engineering Ethics", Prentice Hall, ISBN-13:978-0-13-214521-3.
8. George Reynolds, "Ethics in Information Technology", Cengage Learning, ISBN-10:1285197151.

210250 Audit Course 1**AC1-VI: Intellectual Property Rights and Patents**

Intellectual property is the area of law that deals with protecting the rights of those who create original works. It covers everything from original plays and novels to inventions and company identification marks. The purpose of intellectual property laws is to encourage new technologies, artistic expressions and inventions while promoting economic growth.

Innovation and originality have great potential value. Whatever line of activity you are engaged in, future success depends on them. The last few years have seen intellectual property rights become an issue of general interest: the smart phone "patent wars", the introduction of digital

Rights management (DRM) and the rise of generic pharmaceuticals and open-source software are just some examples that have been in the public eye. Protecting your intellectual rights appropriately should be a top priority. Yet too many people embark on their chosen professions without even a basic awareness of intellectual property.

Teaching Intellectual Property is a very challenging task. It is nothing like teaching any other area of law. First and foremost, IP law covers many areas. Unlike Tort Law or Criminal Law, IP deals not only with Patents, Copyrights, Trademarks and Trade Secrets but also with Unfair Competition, Right of Publicity and more. Moreover, IP interacts with many other areas of law such as Antitrust, Torts, Property Law, and Contracts.

Course Objectives:

- To encourage research, scholarship, and a spirit of inquiry, thereby generating new knowledge.
- To facilitate the transfer of knowledge and technology to intending users to promote utilization of such resources for benefit of the society.
- To encourage students at all levels to develop patentable technologies.
- To provide environment to the students of the Institute for creation, protection, and commercialization of intellectual property and to stimulate innovation.

Course Outcomes:

On completion of the course, learner will be able to–

- Understand the fundamental legal principles related to confidential information, copyright, patents, designs, trademarks and unfair competition
- Identify, apply and assess principles of law relating to each of these areas of intellectual property
- Apply the appropriate ownership rules to intellectual property you have been involved in creating
- Identify which of the four main different types of intellectual property rights may be presented by an output

Course Contents

1. **Introduction to Intellectual Property Law** – The Evolutionary Past - The IPR Tool Kit-Para -Legal Tasks in Intellectual Property Law – Ethical obligations in Para Legal Tasks in Intellectual Property Law - Introduction to Cyber Law – Innovations and Inventions Trade related Intellectual Property Right
2. **Introduction to Trade mark** – Trade mark Registration Process – Post registration Procedures – Trade mark maintenance - Transfer of Rights – Inter partes Proceeding – Infringement - Dilution Ownership of Trade mark – Likelihood of confusion - Trademarks claims – Trademarks Litigations – International Trade mark Law
3. **Introduction to Copyrights** – – Principles of Copyright Principles -The subjects Matter of Copy right – The Rights Afforded by Copyright Law – Copy right Ownership, Transfer and duration – Right to prepare Derivative works – Rights of Distribution – Rights of Perform the work Publicity Copyright Formalities and Registrations - Limitations - Copyright disputes and International Copyright Law – Semiconductor Chip Protection Act
4. **Introduction to Trade Secret** – Maintaining Trade Secret – Physical Security – Employee Limitation - Employee confidentiality agreement - Trade Secret Law - Unfair Competition – Trade Secret Letigation – Breach of Contract – Applying State Law

Reference:

1. Debirag E.Bouchoux: “Intellectual Property”. Cengage learning , New Delhi, ISBN-10:1111648573
2. M. Ashok Kumar and Mohd.Iqbal Ali: “Intellectual Property Right” Serials Pub.
3. Ferrera, Reder, Bird, Darrow, ”Cyber Law. Texts & Cases”, South-Western’s Special Topics Collections, ISBN:0-324-39972-3
4. Prabhuddha Ganguli: “Intellectual Property Rights” Tata Mc-Graw –Hill, New Delhi, ISBN-10:0070077177