

19CS2111: COMPUTER ORGANIZATION

B.Tech. II Year I Sem.

L T P C

3 - - 3

Course Objectives

- To introduce principles of computer organization and the basic architectural concepts.
- To understand the design concepts of control memory and central processing unit.
- To explain the computer arithmetic set of operations and instruction set design.
- To understand the representation of data at the machine level and how computations are performed at machine level.
- To understand the memory organization and I/O organization.
- To become familiar with pipelining, vector processing and memory organization.

Course Outcomes

- Able to explain the principles of computer organization and simple register transfer language to specify various computer operations.
- Able to compare different addressing modes and instruction formats.
- Able to make use of all computer arithmetic operations.
- Able to decide the type of data representations.
- Able to identify the types of memory organizations.
- Able to compare various types of pipeline and processing.

UNIT - I

Digital Computers: Introduction, Block Diagram of Digital Computer, Definition of Computer Organization, Computer Design and Computer Architecture.

Register Transfer Language and Micro operations: Register Transfer language, Register Transfer, Bus and Memory Transfers, Arithmetic Micro Operations, Logic Micro Operations, Shift Micro Operations, Arithmetic Logic Shift Unit.

Basic Computer Organization and Design: Instruction Codes, Computer Registers, Computer

Instructions, Timing and Control, Instruction Cycle, Memory Reference Instructions, Input-Output and Interrupt.

UNIT - II

Micro Programmed Control: Control Memory, Address Sequencing, Micro Program example, Design of Control Unit.

Central Processing Unit: General Register Organization, Instruction Formats, Addressing Modes, Data Transfer and Manipulation, Program Control.

UNIT - III

Data Representation: Data Types, Complements, Fixed Point Representation, Floating Point Representation.

Computer Arithmetic: Addition and Subtraction, Multiplication Algorithms, Division Algorithms, Floating– point Arithmetic Operations. Decimal Arithmetic Unit, Decimal Arithmetic Operations

UNIT - IV

Input-Output Organization: Input-Output Interface, Asynchronous Data Transfer, Modes of Transfer, Priority Interrupt, Direct Memory Access.

Memory Organization: Memory Hierarchy, Main Memory, Auxiliary Memory, Associate Memory, Cache Memory.

UNIT - V

Reduced Instruction Set Computer: CISC Characteristics, RISC Characteristics.

Pipeline and Vector Processing: Parallel Processing, Pipelining, Arithmetic Pipeline, Instruction Pipeline, RISC Pipeline, Vector Processing, Array Processor.

Multi Processors: Characteristics of Multiprocessors, Interconnection Structures, Inter processor Arbitration, Inter processor communication and synchronization, Cache Coherence.

TEXT BOOK:

1. Computer System Architecture – M. Morris Mano, 3rd Edition, Pearson/PHI.

REFERENCES:

1. Computer Organization – Carl Hamacher, Zvonko Vranesic, Safwat Zaky, 5th Edition, McGraw-Hill.
2. Computer Organization and Architecture – William Stallings 6th Edition, Pearson/PHI.
3. Structured Computer Organization – Andrew S. Tanenbaum, 4th Edition, PHI/Pearson

19CY2111: PYTHON PROGRAMMING

B.Tech. II Year I Sem.

L T P C

3 0 0 3

Prerequisites: A course on “Programming for Problem Solving using C”.

Course Objectives:

- Learn Syntax and Semantics and create Functions in Python.
- Handle Strings and Files in Python.
- Understand Lists, Dictionaries and Regular expressions in Python.
- Implement Object Oriented Programming concepts in Python.
- Implement Network Programming in Python.

Course Outcomes:

- Examine Python syntax and semantics and be fluent in the use of Python flow control and functions.
- Demonstrate proficiency in handling Strings and File Systems.
- Create, run and manipulate Python Programs using core data structures like Lists, Dictionaries and use Regular Expressions.
- Interpret the concepts of Object-Oriented Programming as used in Python.
- Implement exemplary applications related to Network Programming

UNIT – I

Introduction to Python: Motivation for learning Python in scenarios like rapid prototyping. Installing Python: basic syntax, interactive shell, editing, saving, and running a script.

Python Basics: Objects- Python Objects, Standard Types, Other Built-in Types, Internal Types, Standard Type Operators, Standard Type Built-in Functions, Categorizing the Standard Types, Unsupported Types ,Numbers - Introduction to Numbers, Integers, Floating Point Real Numbers, Complex Numbers, Operators, Built-in Functions, Related Modules Sequences - Strings, Lists, and Tuples, Mapping and Set Types

UNIT - II

FILES: File Objects, File Built-in Function [open()], File Built-in Methods, File Built-in Attributes, Standard Files, Command-line Arguments, File System, File Execution, Persistent Storage Modules, Related Modules

Exceptions: Exceptions in Python, Detecting and Handling Exceptions, Context Management,*Exceptions as Strings, Raising Exceptions, Assertions, Standard Exceptions, *Creating Exceptions, Why Exceptions (Now)?, Why Exceptions at All?, Exceptions and the sys Module, Related Modules

Modules: Modules and Files, Namespaces, Importing Modules, Importing Module Attributes, Module Built-in Functions, Packages, Other Features of Modules

UNIT - III

Regular Expressions: Introduction, Special Symbols and Characters, Res and Python Multithreaded Programming: Introduction, Threads and Processes, Python, Threads, and the Global Interpreter Lock, Thread Module, Threading Module, Related Modules

UNIT - IV

GUI Programming: Introduction, Tkinter and Python Programming, Brief Tour of Other GUIs, Related Modules and Other GUIs

WEB Programming: Introduction, Web Surfing with Python, Creating Simple Web Clients, Advanced Web Clients, CGI-Helping Servers Process Client Data, Building CGI Application Advanced CGI, Web (HTTP) Servers

UNIT - V

Database Programming: Introduction, Python Database Application Programmer's Interface (DB-API), Object Relational Managers (ORMs), Related Modules

Use of popular Python packages for Scientific Computing: Exercises to understand usage of libraries like Numpy, SciPy, Pandas, Scikit-learn in interpreted and script modes.

TEXT BOOK:

1. Core Python Programming, Wesley J. Chun, Second Edition, Pearson.
2. Guttag, John. Introduction to Computation and Programming Using Python: With Application to Understanding Data Second Edition. MIT Press, 2016. ISBN: 9780262529624.
3. William McKinney, Python for Data Analysis: Data Wrangling with Pandas, NumPy, and Ipython, Second edition (27 October 2017).

REFERENCE BOOKS:

1. Think Python, Allen Downey, Green Tea Press
2. Introduction to Python, Kenneth A. Lambert, Cengage
3. Python Programming: A Modern Approach, Vamsi Kurama, Pearson
4. Learning Python, Mark Lutz, O'Really. Python Programming

19CS2113: DATA STRUCTURES

B.Tech. II Year I Sem.

L T P C

3 1 - 4

Prerequisites:

- A course on “Programming for Problem Solving”.

Course Objectives

- Exploring basic data structures such as stacks and queues.
- Introduces a variety of data structures such as hash tables, search trees, tries, heaps, graphs.
- Introduces searching, sorting and pattern matching algorithms

Course Outcomes

- Ability to design and analyze the time and space complexity and understand the concepts Linked List of data structures
- Understand data structure concepts of Stacks and Queues
- Understand data structure concepts of Trees, Graphs
- Understand the concepts of Searching and Sorting.
- Understand the concepts of Search Trees.
- Understand the concepts of Pattern Matching and Tries.

UNIT- I

Basic concepts : Algorithm Specification-Introduction, Recursive algorithms, Data Abstraction Performance Analysis- time complexity and space complexity, Asymptotic Notation-Big O, Omega and Theta notations, Introduction to Linear and Non Linear data structures.

Linked Lists: Singly Linked Lists-Operations-Insertion, Deletion, Concatenating singly linked lists, Circularly linked lists- operations for Circularly linked lists, Doubly Linked Lists- Operations-Insertion, Deletion. Representation of single, two dimensional arrays, sparse matrices-array and linked representations.

UNIT- II

Introduction to Data Structures - Stack ADT, definition, operations, array and linked implementations in C, applications-infix to postfix conversion, Postfix expression evaluation, recursion implementation, Queue ADT, definition and operations, array and linked Implementations in C, Circular Queues -Insertion and deletion operations, Deque (Double ended Queue)ADT, array and linked implementations in C.

UNIT- III

Trees – Terminology, Representation of Trees, Binary tree ADT, Properties of Binary Trees, Binary Tree Representations-array and linked representations, Binary Tree traversals, Threaded binary trees, Max Priority Queue ADT-implementation-Max Heap-Definition, Insertion into a Max Heap, Deletion from a Max Heap.

Graphs – Introduction, Definition, Terminology, Graph ADT, Graph Representations- Adjacency matrix, Adjacency lists, Graph traversals- DFS and BFS.

UNIT- IV

Searching- Linear Search, Binary Search, Static Hashing-Introduction, hash tables, hash functions, Overflow Handling.

Sorting-Insertion Sort, Selection Sort, Radix Sort, Quick sort, Heap Sort, Comparison of Sorting methods.

UNIT- V

Search Trees-Binary Search Trees, Definition, Operations- Searching, Insertion and Deletion, AVL Trees- Definition and Examples, Insertion into an AVL Tree ,B-Trees, Definition, B-Tree of order m, operations-Insertion and Searching, Introduction to Red-Black and Splay Trees(Elementary treatment-only Definitions and Examples), Comparison of Search Trees.

Pattern matching algorithm- The Knuth-Morris-Pratt algorithm, Tries (examples only).

TEXT BOOKS:

1. Fundamentals of Data structures in C, 2nd Edition, E.Horowitz, S.Sahni and Susan Anderson-Freed, Universities Press.
2. Data structures A Programming Approach with C, D.S.Kushwaha and A.K.Misra, PHI.

REFERENCES:

1. Data structures: A Pseudocode Approach with C, 2nd edition, R.F.Gilberg And B.A.Forouzan, Cengage Learning.
2. Data structures and Algorithm Analysis in C, 2nd edition, M.A.Weiss, Pearson.
3. Data Structures using C, A.M.Tanenbaum,Y. Langsam, M.J.Augenstein, Pearson.
4. Data structures and Program Design in C, 2nd edition, R.Kruse, C.L.Tondo and B.Leung,Pearson.
5. Data Structures and Algorithms made easy in JAVA, 2nd Edition, Narsimha Karumanchi, CareerMonk Publications.
6. Data Structures using C, R.Thareja, Oxford University Press.
7. Data Structures, S.Lipscutz,Schaum's Outlines, TMH.

8. Data structures using C, A.K.Sharma, 2nd edition, Pearson.
9. Data Structures using C &C++, R.Shukla, Wiley India.
10. Classic Data Structures, D.Samanta, 2nd edition, PHI.
11. Advanced Data structures, Peter Brass, Cambridge

19BS2116: MATHEMATICAL AND STATISTICAL FOUNDATIONS

B.Tech. II Year I Sem

L T P C

3 0 0 3

Prerequisites: Mathematics courses of first year of study

Course Objectives:

- The Number Theory basic concepts useful for cryptography etc.
- The theory of Probability, and probability distributions of single and multiple random variables
- The sampling theory and testing of hypothesis and making inferences
- Stochastic process and Markov chains.

Course Outcomes: After learning the contents of this course, the student must be able to

- Apply the number theory concepts to cryptography domain
- Apply the concepts of probability and distributions to some case studies
- Correlate the material of one unit to the material in other units
- Resolve the potential misconceptions and hazards in each topic of study.

UNIT - I

Greatest Common Divisors and Prime Factorization: Greatest common divisors, The Euclidean algorithm, The fundamental theorem of arithmetic, Factorization of integers and the Fermat numbers

Congruences: Introduction to congruences, linear congruence, The Chinese remainder theorem, Systems of linear congruencies

UNIT - II

Simple Linear Regression and Correlation: Introduction to Linear Regression, The Simple Linear Regression Model, Least Squares and the Fitted Model, Properties of the Least Squares Estimators, Inferences Concerning the Regression Coefficients, Prediction, Simple Linear Regression Case Study

Random Variables and Probability Distributions: Concept of a Random Variable, Discrete Probability Distributions, Continuous Probability Distributions, Statistical Independence. Discrete Probability Distributions: Binomial Distribution, Poisson distribution.

UNIT - III

Continuous Probability Distributions: Normal Distribution, Areas under the Normal Curve, Applications of the Normal Distribution, Normal Approximation to the Binomial Fundamental

Sampling Distributions: Random Sampling, Sampling Distributions, Sampling Distribution of Means and the Central Limit Theorem, Sampling Distribution of S^2 , t-Distribution, F Distribution.

UNIT – IV

Estimation & Tests of Hypotheses: Introduction, Statistical Inference, Classical Methods of Estimation. Estimating the Mean, Standard Error of a Point Estimate, Prediction Intervals, Tolerance Limits, Estimating the Variance, Estimating a Proportion for single mean, Difference between Two Means, between Two Proportions for Two Samples and Maximum Likelihood Estimation.

UNIT - V

Stochastic Processes and Markov Chains: Introduction to Stochastic processes- Markov process. Transition Probability, Transition Probability Matrix, First order and Higher order Markov process, nstep transition probabilities, Markov chain, Steady state condition, Markov analysis.

TEXT BOOKS:

1. Kenneth H. Rosen, Elementary number theory & its applications, sixth edition, AddisonWesley, ISBN 978 0-321-50031-1
2. Ronald E. Walpole, Raymond H. Myers, Sharon L. Myers, Keying Ye, Probability & Statistics for Engineers & Scientists, 9th Ed. Pearson Publishers.
3. S. D. Sharma, Operations Research, Kedarnath and Ramnath Publishers, Meerut, Delhi

REFERENCE BOOK:

1. S C Gupta and V K Kapoor, Fundamentals of Mathematical statistics, Khanna publications
2. T.T. Soong, Fundamentals of Probability and Statistics for Engineers, John Wiley & SonsLtd, 2004.
3. Sheldon M Ross, Probability and statistics for Engineers and scientists, Academic Press

19EC2119: ANALOG AND DIGITAL ELECTRONICS

B.Tech. II Year I Sem.

L T P C

3 0 0 3

Course Objectives:

- To introduce components such as diodes, BJTs and FETs.
- To study the applications of components.
- To learn the understanding of various types of amplifier circuits.
- To learn the basic techniques for the design of digital circuits and fundamental concepts used in the design of digital systems.
- To understand the concepts of combinational logic circuits and sequential circuits.

Course Outcomes: Upon completion of the Course, the students will be able to:

- Understand the characteristics of various components.
- Understand the applications of components.
- Design and analyze small signal amplifier circuits.
- Learn Postulates of Boolean algebra and to minimize combinational functions
- Design and analyze combinational and sequential circuits
- Understand the logic families and realization of logic gates.

UNIT - I

Diodes and its Applications: Junction diode characteristics: Open circuited p-n junction, p-n junction as a rectifier, V-I characteristics, effect of temperature, diode resistance, diffusion capacitance, diode switching times, breakdown diodes, Tunnel diodes, photo diode, LED.

Diode Applications - clipping circuits, comparators, half wave rectifier, Full wave rectifier, rectifier with capacitor filter.

UNIT - II

Bipolar Junction Transistor: Transistor characteristics: The junction transistor, transistor as an amplifier, CB, CE, CC configurations, comparison of transistor configurations, the operating point, self-bias or Emitter bias, bias compensation, thermal runaway and stability, transistor at low frequencies, CE amplifier response, gain bandwidth product, Emitter follower, RC coupled amplifier, two cascaded CE and multi stage CE amplifiers.

UNIT - III

FETs and Digital Circuits: Field Effect Transistors: JFET, V-I characteristics, MOSFET, low frequency CS and CD amplifiers, CS and CD amplifiers.

Digital Circuits: Digital (binary) operations of a system, OR gate, AND gate, NOT, EXCLUSIVE OR gate, De Morgan Laws, NAND and NOR DTL gates, modified DTL gates, HTL and TTL gates, output stages, RTL and DCTL, CMOS, Comparison of logic families.

UNIT - IV

Combinational Logic Circuits: Basic Theorems and Properties of Boolean Algebra, Canonical and Standard Forms, SOP and POS, Digital Logic Gates, Simplification of Boolean expressions using K-Map Method, Implementation of Universal Gates, Binary Adder-Subtractor, Decimal Adder, Binary Multiplier, Magnitude Comparator, Decoders, Encoders, Multiplexers.

UNIT - V

Sequential Logic Circuits: Sequential Circuits, Storage Elements: Latches and flip flops, Analysis of Clocked Sequential Circuits, State Reduction and Assignment, Shift Registers, Ripple Counters, Synchronous Counters, Random-Access Memory, Read-Only Memory.

TEXT BOOKS:

1. Integrated Electronics: Analog and Digital Circuits and Systems, 2/e, Jaccob Millman, Christos Halkias and Chethan D. Parikh, *Tata McGraw-Hill Education*, India, 2010.
2. Digital Design, 5/e, Morris Mano and Michael D. Cilette, *Pearson*, 2011.

REFERENCE BOOKS:

1. Electronic Devices and Circuits, Jimmy J Cathey, *Schaum's outline series*, 1988.
2. Digital Principles, 3/e, Roger L. Tokheim, *Schaum's outline series*, 1994.

19CS2151: DATA STRUCTURES LAB

B.Tech. II Year I Sem.

L T P C

- - 3 1.5

Prerequisites:

- A Course on “Programming for problem solving”.

Course Objectives

- It covers various concepts of C programming language
- It introduces searching and sorting algorithms
- It provides an understanding of data structures such as stacks and queues.

Course Outcomes

- Ability to develop C programs for computing and real-life applications using basic elements like control statements, arrays, functions, pointers and strings, and data structures like stacks, queues and linked lists.
- Ability to implement searching and sorting algorithms

LIST OF EXPERIMENTS:

Week 1: Write a program that uses functions to perform the following operations on Single Linked List:

- i) Creation ii) Insertion iii) Deletion iv) Traversal

Week 2: Write a program that uses functions to perform the following operations on double Linked List :

- i) Creation ii) Insertion iii) Deletion iv) Traversal

Week 3: Write a program that uses functions to perform the following operations on Circular Linked List:

- i) Creation ii) Insertion iii) Deletion iv) Traversal

Week 4: Write a program that implement Stack (its operations) using Arrays and Linked list.

Week 5: Write a C program that uses stack operations to convert a given Infix expression into its Postfix.

Week 6: Write a program that implement Queue (its operations) using Arrays and Linked list.

Week 7: Write a C Program to implement a Double Ended Queue ADT using i) Arrays ii) Doubly Linked List

Week 8: Write a C program to implement all the functions of a Dictionary (ADT) using Hashing.

Week 9: Write a program that use recursive functions to perform the following searching operations for a Key value in a given list of integers:

- i) Linear Search ii) Binary Search

Week 10: Write a program to implement the tree traversal methods.

Week 11: Write a program to implement the graph traversal methods.

Week 12: Write a program that implements the following sorting methods to sort a given list of integers in ascending order

- i) SelectionSort
- ii) Quick Sort

Week 13: Write a program that implements the following sorting methods to sort a given list of integers in ascending order Insertion Sort ii) Merge Sort

Week 14: Write a program that implements the following sorting methods to sort a given list of integers in ascending order

- i) Radix Sort
- ii) Heap Sort

Week 15: Write a C program for implementing Minimum cost spanning tree

- a) Prim's
- b) Kruskal's.

TEXTBOOKS:

1. Fundamentals of Data Structures in C, 2nd Edition, E. Horowitz, S. Sahni and Susan Anderson Freed, Universities Press.
2. Data Structures using C—A.S. Tanenbaum, Y. Langsam, and M. J. Augenstein, PHI/Pearson Education.

REFERENCES:

1. Data Structures: A Pseudocode Approach with C, 2nd Edition, R. F. Gilberg and B.A. Forouzan, Cengage Learning

19CY2152: PYTHON PROGRAMMING LAB

B.Tech. II Year I Sem

L T P C

0 0 3 1.5

Prerequisites: A course on “Programming for Problem Solving”.

Course Objectives

- To be able to introduce core programming basics and program design with functions using Python programming language.
- To understand a range of Object-Oriented Programming, as well as in-depth data and information processing techniques.
- To understand the high-performance programs designed to strengthen the practical expertise.

Course Outcome

- Student should be able to understand the basic concepts scripting and the contributions of scripting language
- Ability to explore python especially the object-oriented concepts, and the built in objects of Python.
- Ability to create practical and contemporary applications such as TCP/IP network programming, Web applications, discrete event simulations

List of Experiments:

1. Write a program to demonstrate different number data types in Python.
2. Write a program to perform different Arithmetic Operations on numbers in Python.
3. Write a program to create, concatenate and print a string and accessing sub-string from a given string.
4. Write a python script to print the current date in the following format “Sun May 29 02:26:23 IST 2017”
5. Write a program to create, append, and remove lists in python.
6. Write a program to demonstrate working with tuples in python.
7. Write a program to demonstrate working with dictionaries in python.
8. Write a python program to find largest of three numbers.
9. Write a Python program to convert temperatures to and from Celsius, Fahrenheit. [Formula : $c/5 = f-32/9$]
10. Write a Python program to construct the following pattern, using a nested for loop

```
*
* *
* * *
* * * *
* * * * *
* * * *
* * *
* *
*
```

11. Write a Python script that prints prime numbers less than 20.
12. Write a python program to find factorial of a number using Recursion.
13. Write a program that accepts the lengths of three sides of a triangle as inputs. The program output should indicate whether or not the triangle is a right triangle (Recall from the Pythagorean Theorem that in a right triangle, the square of one side equals the sum of the squares of the other two sides).
14. Write a python program to define a module to find Fibonacci Numbers and import the module to another program.
15. Write a python program to define a module and import a specific function in that module to another program.
16. Write a script named copyfile.py. This script should prompt the user for the names of two text files. The contents of the first file should be input and written to the second file.
17. Write a program that inputs a text file. The program should print all of the unique words in the file in alphabetical order.
18. Write a Python class to convert an integer to a roman numeral.
19. Write a Python class to implement $\text{pow}(x, n)$
20. Write a Python class to reverse a string word by word.

19EC2156: ANALOG AND DIGITAL ELECTRONICS LAB

B.Tech. II Year I Sem.

L T P C

0 0 2 1

Course Objectives

- To introduce components such as diodes, BJTs and FETs.
- To know the applications of components.
- To give understanding of various types of amplifier circuits
- To learn basic techniques for the design of digital circuits and fundamental concepts used in the design of digital systems.
- To understand the concepts of combinational logic circuits and sequential circuits.

Course Outcomes:

- Understand the characteristics of various components.
- Understand the application of components.
- Design and analyze small signal amplifier circuits.
- Postulates of Boolean algebra and to minimize combinational functions
- Design and analyze combinational and sequential circuits
- Understand the logic families and realization of logic gates.

List of Experiments

1. Full Wave Rectifier with & without filters
2. Common Emitter Amplifier Characteristics
3. Common Base Amplifier Characteristics
4. Common Source amplifier Characteristics
5. Measurement of h-parameters of transistor in CB, CE, CC configurations
6. Input and Output characteristics of FET in CS configuration
7. Realization of Boolean Expressions using Gates
8. Design and realization logic gates using universal gates
9. generation of clock using NAND / NOR gates
10. Design a 4 – bit Adder /Subtractor
11. Design and realization a Synchronous and Asynchronous counter using flip-flops
12. Realization of logic gates using DTL, TTL, ECL, etc.

TEXT BOOKS:

1. Integrated Electronics: Analog and Digital Circuits and Systems, 2/e, Jacob Millman, Christos Halkias and Chethan D. Parikh, Tata McGraw-Hill Education, India, 2010.
2. Digital Design, 5/e, Morris Mano and Michael D. Cilette, Pearson, 2011.

REFERENCES:

1. Electronic Devices and Circuits, Jimmy J Cathey, Schaum's outline series, 1988.
2. Digital Principles, 3/e, Roger L. Tokheim, Schaum's outline series, 1994

19MC0006: UNIVERSAL HUMAN VALUES - II

B.Tech. II Year I Sem.

L T P C

2 - - 0

Pre-requisites:

- None. Universal Human Values 1 (desirable)

Course Objectives

The objective of the course is four fold:

- 1. Development of a holistic perspective based on self-exploration about
- Themselves (human being), family, society and nature/existence.
- Understanding (or developing clarity) of the harmony in the human being,
- family, society and nature/existence
- Strengthening of self-reflection.
- Development of commitment and courage to act.

COURSE TOPICS:

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

Module 1: Course Introduction - Need, Basic Guidelines, Content and Process for Value Education

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

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

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

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

12. Programs to ensure Sanyam and Health.

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

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

13. Understanding values in human-human relationship; meaning of Justice (nine universal values in relationships) and program for its fulfilment to ensure mutual happiness; Trust and Respect as the foundational values of relationship

14. Understanding the meaning of Trust; Difference between intention and competence

15. Understanding the meaning of Respect, Difference between respect and differentiation; the other salient values in relationship

16. Understanding the harmony in the society (society being an extension of family):Resolution, Prosperity, fearlessness (trust) and co-existence as comprehensive Human Goals

17. Visualizing a universal harmonious order in society- Undivided Society, Universal Order- from family to world family.

Include practice sessions to reflect on relationships in family, hostel and institute as extended family, real life examples, teacher-student relationship, goal of education etc. Gratitude as a universal value in relationships. Discuss with scenarios. Elicit examples from students' lives

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

18. Understanding the harmony in the Nature

19. Interconnectedness and mutual fulfillment among the four orders of nature recycle ability and self-regulation in nature

20. Understanding Existence as Co-existence of mutually interacting units in all-pervasive space

21. Holistic perception of harmony at all levels of existence. Include practice sessions to discuss human being as cause of imbalance in nature (film "Home" can be used), pollution, depletion of resources and role of technology etc.

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

22. Natural acceptance of human values

23. Definitiveness of Ethical Human Conduct

24. Basis for Humanistic Education, Humanistic Constitution and Humanistic Universal Order.

25. Competence in professional ethics:

a.Ability to utilize the professional competence for augmenting universal human order

b.Ability to identify the scope and characteristics of people friendly and eco-friendly production systems,

c. Ability to identify and develop appropriate technologies and management patterns for above production systems.

26. Case studies of typical holistic technologies, management models and production systems

27. Strategy for transition from the present state to Universal Human Order:

a. At the level of individual: as socially and ecologically responsible engineers, technologists and managers

b. At the level of society: as mutually enriching institutions and organizations

28. Sum up.

Include practice Exercises and Case Studies will be taken up in Practice (tutorial)Sessions eg. To discuss the conduct as an engineer or scientist etc.

TEXT BOOKS:

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

REFERENCES:

1. Jeevan Vidya: EkParichaya, A Nagaraj, Jeevan VidyaPrakashan, Amarkantak,1999.
2. Human Values, A.N. Tripathi, New Age Intl. Publishers, New Delhi, 2004.
3. The Story of Stuff (Book).
4. The Story of My Experiments with Truth - by Mohandas Karamchand Gandhi
5. Small is Beautiful - E. F Schumacher.
6. Slow is Beautiful - Cecile Andrews
7. Economy of Permanence - J C Kumarappa
8. Bharat Mein Angreji Raj - PanditSunderlal
9. Rediscovering India - by Dharampal
10. Hind Swaraj or Indian Home Rule - by Mohandas K. Gandhi
11. India Wins Freedom - Maulana Abdul Kalam Azad
12. Vivekananda - Romain Rolland (English)
13. Gandhi - Romain Rolland (English)

MODE OF CONDUCT

- Lectures hours are to be used for interactive discussion, placing the proposals about the topics at hand and motivating students to reflect, explore and verify them.
- Tutorial hours are to be used for practice sessions. While analyzing and discussing the topic, the faculty mentor's role is in pointing to essential elements to help in sorting them out from the surface elements. In other words, help the students explore the important or critical elements.
- In the discussions, particularly during practice sessions (tutorials), the mentor encourages the student to connect with one's own self and do self-observation, self-reflection and self-exploration.
- Scenarios may be used to initiate discussion. The student is encouraged to take up "ordinary" situations rather than "extra-ordinary" situations. Such observations and their analyses are shared and discussed with other students and faculty mentor, in a group sitting.
- Tutorials (experiments or practical) are important for the course. The difference is that the laboratory is everyday life, and practical are how you behave and working real life. Depending on the nature of topics, worksheets, home assignment and/or activity are included. The practice sessions (tutorials) would also provide support to a student in performing actions commensurate to his/her beliefs. It is intended that this would lead to development of commitment, namely behaving and working based on basic human values.
- It is recommended that this content be placed before the student as it is, in the form of a basic foundation course, without including anything else or excluding any part of this content. Additional content may be offered in separate, higher courses.
- This course is to be taught by faculty from every teaching department, including HSS faculty.
- Teacher preparation with a minimum exposure to at least one 8-day FDP on Universal Human Values is deemed essential.

ASSESSMENT:

- The assessment is to provide a fair state of development of the student, so participation in classroom discussions, self-assessment, peer assessment etc. will be used in evaluation.

Example:

Assessment by faculty mentor : 10 marks

Self-assessment : 10 marks

Assessment by peers : 10 marks

Socially relevant project/Group Activities/Assignments: 20 marks

The overall pass percentage is 40%. In case the student fails, he/she must repeat the course.

OUTCOME OF THE COURSE:

- By the end of the course, students are expected to become more aware of themselves, and their surroundings (family, society, nature); they would become more responsible in life, and in handling problems with sustainable solutions, while keeping human relationships and human nature in mind.

- They would have better critical ability. They would also become sensitive to their commitment towards what they have understood (human values, human relationship and human society). It is hoped that they would be able to apply what they have learnt to their own self in different day-to-day settings in real life, at least a beginning would be made in this direction.
- This is only an introductory foundational input. It would be desirable to follow it up by faculty-student or mentor-mentee programs throughout their time with the institution. Higher level courses on human values in every aspect of living. E.g. as a professional

II YEAR –SEM-II

19CS2211: OPERATING SYSTEMS

B.Tech. II Year II Sem.

L T P C
3 - - 3

Course Objectives

- Provide an introduction to operating system concepts (i.e, Operating System services, OS Structure)
- Introduces the way an *operating system* can make the computer more productive by the effective **management of processes** (i.e., threads, scheduling, Synchronization)
- Introduce basic UNIX commands, system call interface for process management; inter process communication and I/O in UNIX.
- To understand the basic **memory management** of operating system.
- To elucidate **deadlocks**, present a number of various techniques for preventing or avoiding or recovering from deadlocks in a computer system
- Introduce the issues to be considered in the design and development of operating system with high **protection** and ease of access.

Course Outcomes

- Will be able to explain services & structure of operating systems.
- Demonstrate the knowledge of the components of computer and their respective roles in computing and illustrate various methods of process scheduling, synchronization.
- Ability to recognize and resolve user problems related to memory management with standard operating system techniques.
- Gain practical knowledge of how programming languages, operating systems, and architectures interact and how to use each effectively to implement file system directory Structures.
- Will be able to apply security mechanisms and techniques to handle deadlocks.
- Will be able to do Programming and debugging C code at the system level communicating directly with an operating system via system calls.

UNIT - I

Overview-Introduction: Operating system objectives, User view, System view, Operating system definition ,Computer System Organization, Computer System Architecture, OS Structure, OS Operations, Process Management, Memory Management, Storage Management, Protection and Security, Computing Environments. Operating System services, User and OS Interface, System Calls, Types of System Calls, System Programs, Operating System Design and Implementation, OS Structure.

UNIT - II

Process and CPU Scheduling: Process concepts-The Process, Process State, Process Control Block, Threads, Process Scheduling-Scheduling Queues, Schedulers, Context Switch, Operations on Processes, System calls-fork(),exec(),wait(),exit(),. Process Scheduling-Basic concepts, Scheduling Criteria, Scheduling algorithms, Multiple Processor Scheduling, Real-Time Scheduling, Thread scheduling, Linux scheduling and Windows scheduling.

Inter process communication: Background, IPC using ordinary pipes and named pipes in Unix.

Process Synchronization: Background, The Critical Section Problem, Peterson's solution,

Synchronization Hardware, Semaphores, Classic Problems of Synchronization, Monitors, Synchronization in Linux and Windows.

UNIT - III

Memory Management: Background, Swapping, Contiguous Memory Allocation, Segmentation, Paging, Structure of Page Table.

Virtual Memory Management: Background, Demand Paging, Copy-on-Write, Page Replacement, Page Replacement Algorithms, Allocation of Frames, Thrashing, Virtual memory in Windows.

UNIT - IV

Storage Management-File System: Concept of a File, System calls for file operations- open(), read(), write(), close(), seek(), unlink(), Access methods, Directory, File System Mounting, File Sharing, Protection.

File System Implementation: File System Structure, File System Implementation, Directory Implementation, Allocation methods, Free-space Management, Efficiency, and Performance.

Mass Storage Structure: Overview of Mass Storage Structure, Disk Structure, Disk Scheduling, Disk Management, Swap space Management.

UNIT - V

Deadlocks: System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, and Recovery from Deadlock. **Protection:** System Protection, Goals of Protection, Principles of Protection, Domain of Protection, Access Matrix, Implementation of Access Matrix, Access Control, Revocation of Access Rights.

TEXT BOOKS:

1. Operating System Principles- Abraham Silberchatz, Peter B. Galvin, Greg Gagne 7th Edition, John Wiley
2. Advanced programming in the UNIX environment, W.R. Stevens, Pearson education.

REFERENCES:

1. Operating Systems – Internals and Design Principles Stallings, Fifth Edition–2005,Pearson Education/PHI
2. Operating System A Design Approach- Crowley, TMH.
3. Modern Operating Systems, Andrew S. Tanenbaum 2nd edition, Pearson/PHI
4. UNIX programming environment, Kernighan and Pike, PHI/ Pearson Education
5. UNIX Internal - The New Frontiers U. Vahalia, Pearson Education.

19CY2212: COMPUTER NETWORKS

B.Tech. II Year II Sem.

L T P C

3 1 0 4

Prerequisites

1. A course on “Programming for problem solving”.
2. A course on “Data Structures”.

Course Objectives:

- The objective of the course is to equip the students with a general overview of the concepts and fundamentals of computer networks.
- Familiarize the students with the standard models for the layered approach to communication between machines in a network and the protocols of the various layers.

Course Outcomes:

- Gain the knowledge of the basic computer network technology.
- Gain the knowledge of the functions of each layer in the OSI and TCP/IP reference model.
- Obtain the skills of sub netting and routing mechanisms.
- Familiarity with the essential protocols of computer networks, and how they can be applied in network design and implementation.

UNIT - I

Network hardware, Network software, OSI, TCP/IP Reference models, Example Networks: ARPANET, Internet.

Physical Layer: Guided Transmission media: twisted pairs, coaxial cable, fiber optics, Wireless transmission.

UNIT - II

Data link layer: Design issues, framing, Error detection and correction.

Elementary data link protocols: simplex protocol, a simplex stop and wait protocol for an error-free channel, a simplex stop and wait protocol for noisy channel.

Sliding Window protocols: A one-bit sliding window protocol, a protocol using Go-Back-N, A protocol using Selective Repeat, Example data link protocols.

Medium Access sub layer: The channel allocation problem, Multiple access protocols: ALOHA, Carrier sense multiple access protocols, collision free protocols. Wireless LANs, Data link layer switching.

UNIT - III

Network Layer: Design issues, Routing algorithms: shortest path routing, Flooding, Hierarchical routing, Broadcast, Multicast, distance vector routing, Congestion Control Algorithms, Quality of Service, Internetworking, The Network layer in the internet.

UNIT - IV

Transport Layer: Transport Services, Elements of Transport protocols, Connection management, TCP and UDP protocols.

UNIT - V

Application Layer: Domain name system, SNMP, Electronic Mail; the World WEB, HTTP, Streaming audio and video.

TEXT BOOK:

1. Computer Networks -- Andrew S Tanenbaum, David. j. Wetherall, 5th Edition.
Pearson Education/PHI

REFERENCE BOOKS:

1. An Engineering Approach to Computer Networks-S. Keshav, 2nd Edition, Pearson Education
2. Data Communications and Networking – Behrouz A. Forouzan. Third EditionTMH.

19MB2216: BUSINESS ECONOMICS AND FINANCIAL ANALYSIS

B.Tech. II Year II Sem.

LTP C

300 3

Prerequisites: None

Course Objective:

- To learn the basic Business types, impact of the Economy on Business and Firms specifically. To analyze the Business from the Financial Perspective.

Course Outcome:

- The students will understand the various Forms of Business and the impact of economic variables on the Business. The Demand, Supply, Production, Cost, Market Structure, Pricing aspects are learnt. The Students can study the firm's financial position by analyzing the Financial Statements of a Company.

UNIT – I

Introduction to Business and Economics:

Business: Structure of Business Firm, Theory of Firm, Types of Business Entities, Limited Liability Companies, Sources of Capital for a Company, Non-Conventional Sources of Finance.

Economics: Significance of Economics, Micro and Macro Economic Concepts, Concepts and Importance of National Income, Inflation, Money Supply in Inflation, Business Cycle, Features and Phases of Business Cycle. Nature and Scope of Business Economics, Role of Business Economist, Multidisciplinary nature of Business Economics.

UNIT - II

Demand and Supply Analysis:

Elasticity of Demand: Elasticity, Types of Elasticity, Law of Demand, Measurement and Significance of Elasticity of Demand, Factors affecting Elasticity of Demand, Elasticity of Demand in decision making, Demand Forecasting: Characteristics of Good Demand Forecasting, Steps in Demand Forecasting, Methods of Demand Forecasting.

Supply Analysis: Determinants of Supply, Supply Function & Law of Supply.

UNIT - III

Production, Cost, Market Structures & Pricing:

Production Analysis: Factors of Production, Production Function, Production Function with one variable input, two variable inputs, Returns to Scale, Different Types of Production

Functions.

Cost analysis: Types of Costs, Short run and Long run Cost Functions.

Market Structures: Nature of Competition, Features of Perfect competition, Monopoly, Oligopoly, Monopolistic Competition.

Pricing: Types of Pricing, Product Life Cycle based Pricing, Break Even Analysis, Cost Volume Profit Analysis.

UNIT -IV

Financial Accounting: Accounting concepts and Conventions, Accounting Equation, Double-Entry system of Accounting, Rules for maintaining Books of Accounts, Journal, Posting to Ledger, Preparation of Trial Balance, Elements of Financial Statements, Preparation of Final Accounts.

UNIT - V

Financial Analysis through Ratios: Concept of Ratio Analysis, Liquidity Ratios, Turnover Ratios, Profitability Ratios, Proprietary Ratios, Solvency, Leverage Ratios (simple problems). Introduction to Fund Flow and Cash Flow Analysis (simple problems).

TEXT BOOKS:

1. D.D. Chaturvedi, S.L. Gupta, Business Economics - Theory and Applications, International Book House Pvt. Ltd.2013.
2. Dhanesh K Khatri, Financial Accounting, Tata McGraw Hill,2011.
3. Geethika Ghosh, Piyali Gosh, Purba Roy Choudhury, Managerial Economics, 2e, Tata McGraw Hill Education Pvt. Ltd.2012.

REFERENCE BOOKS:

1. Paresh Shah, Financial Accounting for Management 2e, Oxford Press,2015.
2. S.N. Maheshwari, Sunil K Maheshwari, Sharad K Maheshwari, Financial Accounting, 5e, Vikas Publications,2013.

19DS2214: OBJECT ORIENTED PROGRAMMING THROUGH JAVA

B.TECH II Year II Sem.

L T P C

3 0 0 3

Course Objectives:

- To introduce the object-oriented programming concepts.
- To understand object-oriented programming concepts, and apply them in solving problems.
- To introduce the principles of inheritance and polymorphism; and demonstrate how they relate to the design of abstract classes
- To introduce the implementation of packages and interfaces
- To introduce the concepts of exception handling and multithreading.
- To introduce the design of Graphical User Interface using applets and swing controls.

Course Outcomes:

- Able to solve real world problems using OOP techniques.
- Able to understand the use of abstract classes.
- Able to solve problems using java collection framework and I/O classes.
- Able to develop multithreaded applications with synchronization.
- Able to develop applets for web applications.
- Able to design GUI based applications

UNIT – I

Object-Oriented Thinking: A way of viewing world – Agents and Communities, messages and methods, Responsibilities, Classes and Instances, Class Hierarchies- Inheritance, Method binding, Overriding and Exceptions, Summary of Object-Oriented concepts. Java buzzwords,

An Overview of Java, Data types, Variables and Arrays, operators, expressions, control statements, Introducing classes, Methods and Classes, String handling.

Inheritance–Inheritanceconcept,Inheritancebasics,Memberaccess,Constructors, Creating Multilevel hierarchy, super uses, using final with inheritance, Polymorphism-ad hoc polymorphism, pure polymorphism, method overriding, abstract classes, Object class, forms of inheritance specialization, specification, construction, extension, limitation, combination, benefits of inheritance, costs of inheritance.

UNIT – II

Packages- Defining a Package, CLASSPATH, Access protection, importing packages. Interfaces- defining an interface, implementing interfaces, Nested interfaces, applying interfaces, variables in interfaces and extending interfaces. Stream based I/O (java.io)– The Stream classes-Byte streams and Character streams, Reading console Input and Writing Console Output, File class, Reading and writing Files, Random access file operations, The Console class, Serialization, Enumerations, auto boxing, generics.

UNIT - III

Exception handling- Fundamentals of exception handling, Exception types, Termination or presumptive models, Uncaught exceptions, using try and catch, multiple catch clauses, nested try

statements, throw, throws and finally, built- in exceptions, creating own exception sub classes.
Multithreading- Differences between thread-based multitasking and process-based multitasking, Java thread model, creating threads, thread priorities, synchronizing threads, inter thread communication.

UNIT - IV

The Collections Framework(java.util)- Collections overview, Collection Interfaces, The Collection classes- Array List, Linked List, Hash Set, Tree Set, Priority Queue, Array De-queue. Accessing a Collection via an Iterator, Using an Iterator, The For-Each alternative, Map Interfaces and Classes, Comparators, Collection algorithms, Arrays, The Legacy Classes and Interfaces- Dictionary, Hash table, Properties, Stack, Vector More Utility classes, String Tokenizer, Bit Set, Date, Calendar, Random, Formatter, Scanner

UNIT - V

GUI Programming with Swing – Introduction, limitations of AWT, MVC architecture, components, containers. Understanding Layout Managers, Flow Layout, Border Layout, Grid Layout, Card Layout, Grid Bag Layout. Event Handling- The Delegation event model- Events, Event sources, Event Listeners, Event classes, Handling mouse and keyboard events, Adapter classes, Inner classes, Anonymous Inner classes. A Simple Swing Application, Applets – Applets and HTML, Security Issues, Applets and Applications, passing parameters to applets. Creating a Swing Applet, Painting in Swing, A Paint example, Exploring Swing Controls- JLabel and Image Icon, JText Field, The Swing Buttons JButton, JToggleButton, JCheckBox, JRadioButton, JTabbed Pane, JScrollPane, JList, JComboBox, Swing Menus, Dialogs.

TEXT BOOKS:

1. Java The complete reference, 9th edition, Herbert Schildt, McGraw Hill Education (India) Pvt.Ltd.
2. Understanding Object-Oriented Programming with Java, updated edition, T. Budd, Pearson Education.

REFERENCE BOOKS:

1. An Introduction to programming and OO design using Java, J. Nino and F.A. Hosch, JohnWiley & sons
2. Introduction to Java programming, Y. Daniel Liang, Pearson Education.
3. Object Oriented Programming through Java, P. Radha Krishna, University Press.
4. Programming in Java, S. Malhotra, S. Chaudhary, 2nd edition, Oxford Univ. Press.
5. Java Programming and Object-oriented Application Development, R. A. Johnson, Cengage Learning.

19BS2215: DISCRETE MATHEMATICAL STRUCTURES

B.TECH II Year II Sem

L T P C

3 - - 3

Prerequisites: An understanding of Mathematics in general is sufficient.

Course Objectives:

- Translate statements from a natural language into its symbolic structures in logic.
- Define the syntax and semantics of propositional and predicate logic.
- To learn set theory, Relations, functions, ordering relations.
- To introduce generating functions and recurrence relations.
- To learn Graph Theory for solving problems.

Course Outcomes:

- Ability to understand and construct precise mathematical proofs.
- Ability to use Mathematical logic to formulate precise statements.
- Ability to perform operations on discrete structures such as sets, functions and relations.
- Ability to solve discrete mathematics problems that involve computing Permutations and combinations of a set.
- Ability to analyze and solve problems involving recurrence relations and generating functions.
- Ability to apply graph theory in solving computing problems.

UNIT – I:

Mathematical Logic and Proofs: Propositional Logic, Applications of Propositional Logic, Propositional Equivalence, Predicates and Quantifiers, Nested Quantifiers, Rules of Inference, Introduction to Proofs, Proof Methods and Strategy: Direct Proof, Indirect Proof, and Proof by Contradiction.

UNIT-II:

Sets and Relations: Sets, Functions, Cardinality of Sets, Relations and their Properties, Representing Relations, Matrix Representation of Relations, Closures of Relations, Equivalence Relations, Partial Ordering, Lattices.

Algebraic Structures: Introduction, Algebraic Systems, Semi groups and Monoids, Groups.

UNIT- III:

Elementary Combinatorics : Basics of Counting, Combinations and Permutations, Enumeration of Combinations and Permutations, Enumerating Combinations and Permutations with Repetitions, Enumerating Permutations with Constrained Repetitions, Binomial Coefficients, the Binomial and Multinomial Theorems.

UNIT IV:

Recurrence Relations: Sequences and Summations, Generating Functions, Calculating coefficients of Generating Functions, Recurrence Relations, Solving Linear Recurrence Relations by substitution method and Generating Functions, Inclusion-Exclusion, Applications of Inclusion-Exclusion.

UNIT - V

Graphs: 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, Chromatic number, Graph Coloring.

Trees: Introduction to Trees, Applications of Trees, Spanning Trees, Minimum Spanning Trees.

TEXT BOOKS:

1. Discrete Mathematics and its Applications with Combinatorics and Graph Theory – Kenneth H Rosen, 7th Edition, TMH.

REFERENCE BOOKS:

1. Discrete Mathematical Structures with Applications to Computer Science – J.P. Tremblay and R. Manohar, TMH.

19CY2251: COMPUTER NETWORKS LAB

B.Tech. II Year II Sem

L T P C

0 0 3 1.5

Course Objectives:

- To understand the working principle of various communication protocols.
- To understand the network simulator environment and visualize a network topology and observe its performance.
- To analyze the traffic flow and the contents of protocol frames.

Course Outcomes:

- Implement data link layer framing methods.
- Analyze error detection and error correction codes.
- Implement and analyze routing and congestion issues in network design.
- Implement Encoding and Decoding techniques used in presentation layer.
- To be able to work with different network tools.

List of Experiments

1. Implement the data link layer framing methods such as character, character-stuffing and bit stuffing.
2. Write a program to compute CRC code for the polynomials CRC-12, CRC-16 and CRC-CCIP
3. Develop a simple data link layer that performs the flow control using the sliding window protocol, and loss recovery using the Go-Back-N mechanism.
4. Implement Dijkstra's algorithm to compute the shortest path through a network
5. Take an example subnet of hosts and obtain a broadcast tree for the subnet.
6. Implement distance vector routing algorithm for obtaining routing tables at each node.
7. Implement data encryption and data decryption
8. Write a program for congestion control using Leaky bucket algorithm.
9. Write a program for frame sorting technique used in buffers.
10. Wireshark
 - i. Packet Capture Using Wireshark
 - ii. Starting Wireshark
 - iii. Viewing Captured Traffic
 - iv. Analysis and Statistics & Filters.
11. How to run Nmap scan
12. Operating System Detection using Nmap
13. Do the following using NS2 Simulator
 - i. NS2 Simulator-Introduction
 - ii. Simulate to Find the Number of Packets Dropped
 - iii. Simulate to Find the Number of Packets Dropped by TCP/UDP
 - iv. Simulate to Find the Number of Packets Dropped due to Congestion
 - v. Simulate to Compare Data Rate & Throughput.
 - vi. Simulate to Plot Congestion for Different Source/Destination

vii. Simulate to Determine the Performance with respect to Transmission of Packets

19CS2252: OPERATING SYSTEMS LAB

B.Tech. II Year II Sem.

L T P C

- - 2 1

Course Objectives

- To write programs in a Linux environment using system calls.
- To implement the scheduling algorithms.
- To implement page replacement algorithms
- To implement file allocation methods.
- To understand and implement IPC mechanisms using named and unnamed pipes.
- To develop solutions for synchronization problems using semaphores.

Course Outcomes

- Ability to develop application programs using system calls in Unix.
- Ability to implement inter process communication between two processes.
- Ability to design and solve synchronization problems.
- Ability to simulate and implement operating system concepts such as scheduling, deadlock management, file management, and memory management.

Use Linux operating system and GNU C compiler.

List of Programs:

1. a) Write a shell script that accepts a filename, starting and ending line numbers as arguments and displays all the lines between the given line numbers.
b) Write a shell script that deletes all lines containing a specific word in one or more files supplied as arguments to it.
c) Write a shell script that displays a list of all the files in the current directory to which the user has read, write and execute permissions.
2. a) Write a shell script that receives any number of filenames as arguments and checks if every argument supplied is a file or a directory, and reports accordingly. Whenever the argument is a file, the number of lines on it is also reported.
b) Write a shell script that accepts a list of filenames as its argument, counts and reports the accuracy of each word that is present in the first argument file on another argument file.
c) Write a shell script to list all of the directory files in a directory.
3. a) Write a shell script to find the factorial of a given integer.
b) Write an awk script to count the number of lines in a file that do not contain vowels.
c) Write an awk script to find the number of characters, words and lines in a file.
4. a) Write a C program that makes a copy of a file using standard I/O and system calls.
b) Implement the following LINUX commands in C using system call.
i) cat ii) mv
c) Write a C program to list files in a directory.
5. (a) Write a C program to copy the contents of one file to another file using system calls.

Mm

- (b) Write C programs to simulate the following CPU scheduling algorithms:
 - i) FCFS
 - ii) SJF
- 6. a) Write C programs to simulate the following CPU scheduling algorithms:
 - i) Round Robin
 - ii) Priority
- b) Write a C program to implement the ls| sort command (Use unnamed Pipe).
- 7. a) Write C programs to implement IPC between two unrelated processes using named pipe.
- b) Write a C program to solve the Dining- Philosopher problem using semaphores.
- 8. Write C programs to simulate the following techniques of memory management:
 - a) Paging
 - b) Segmentation
- 9. Write C programs to simulate the following page replacement algorithms:
 - a) FIFO
 - b) LRU
 - c) LFU
- 10. Write C programs to simulate the following File allocation methods:
 - a) Contiguous
 - b) Linked
 - c) Indexed
- 11. Write C programs to simulate the following File organization techniques:
 - a) Single level directory
 - b) Two level
 - c) Hierarchical
- 12. Write a C program to simulate Bankers Algorithm for Deadlock Avoidance.
- 13. Write a C program to simulate Bankers Algorithm for Deadlock Prevention.

TEXT BOOKS:

1. Operating System Principles - Abraham Silchatz, Peter B. Galvin, Greg Gagne 7th Edition, John Wiley.
2. Advanced Programming in the UNIX Environment by W. Richard Stevens Pearson Education.

REFERENCES:

1. An Introduction to Operating Systems, P.C.P Bhatt, 2nd edition, PHI.
2. UNIX System Programming Using C++, Terrence Chan, PHI/Pearson.
3. Modern Operating Systems, Andrew S Tanenbaum, 3rd Edition, PHI.

19DS2253: JAVA PROGRAMMING LAB

B.Tech. II Year IISem.

L T P C

0 0 3 1.5

Course Objectives:

- To write programs using abstract classes.
- To write programs for solving real world problems using java collection framework.
- To write multi threaded programs.
- To write GUI programs using swing controls in Java.
- To introduce java compiler and eclipse platform.
- To impart hands on experience with java programming.

Course Outcomes:

- Able to write programs for solving real world problems using java collection framework.
- Able to write programs using abstract classes.
- Able to write multi threaded programs.
- Able to write GUI programs using swing controls in Java.

Note:

1. Use LINUX and MySQL for the Lab Experiments. Though not mandatory, encourage the use of Eclipse platform.
2. The list suggests the minimum program set. Hence, the concerned staff is requested to add more problems to the list as needed.

List of Experiments:

1. Use Eclipse or Net bean platform and acquaint with the various menus. Create a test project, add a test class, and run it. See how you can use auto suggestions, auto fill. Try code formatter and code refactoring like renaming variables, methods, and classes. Try debug step by step with a small program of about 10 to 15 lines which contains at least one if else condition and a for loop.
2. Write a Java program that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +, -,*, % operations. Add a text field to display the result. Handle any possible exceptions like divided by zero.
3. a) Develop an applet in Java that displays a simple message.
a) Develop an applet in Java that receives an integer in one text field, and computes its factorial Value and returns it in another text field, when the button named "Compute" is clicked.

4. Write a Java program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num 2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a Number Format Exception. If Num2 were Zero, the program would throw an Arithmetic Exception. Display the exception in a message dialog box.
5. Write a Java program that implements a multi-thread application that has three threads. First thread generates random integer every 1 second and if the value is even, second thread computes the square of the number and prints. If the value is odd, the third thread will print the value of cube of the number.
6. Write a Java program for the following:
 - Create a doubly linked list of elements.
 - Delete a given element from the above list.
 - Display the contents of the list after deletion.
7. Write a Java program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green with radio buttons. On selecting a button, an appropriate message with “Stop” or “Ready” or “Go” should appear above the buttons in selected color. Initially, there is no message shown.
8. Write a Java program to create an abstract class named Shape that contains two integers and an empty method named print Area (). Provide three classes named Rectangle, Triangle, and Circle such that each one of the classes extends the class Shape. Each one of the classes contains only the method print Area () that prints the area of the given shape.
9. Suppose that a table named Table.txt is stored in a text file. The first line in the file is the header, and the remaining lines correspond to rows in the table. The elements are separated by commas. Write a java program to display the table using Labels in Grid Layout.
10. Write a Java program that handles all mouse events and shows the event name at the center of the window when a mouse event is fired (Use Adapter classes).
11. Write a Java program that loads names and phone numbers from a text file where the data is organized as one line per record and each field in a record are separated by a tab (\t). It takes a name or phone number as input and prints the corresponding other

value from the hash table (hint: use hash tables).

12. Write a Java program that correctly implements the producer – consumer problem using the concept of inter thread communication.
13. Write a Java program to list all the files in a directory including the files present in all its subdirectories.
14. Write a Java program that implements Quick sort algorithm for sorting a list of names in ascending order
15. Write a Java program that implements Bubble sort algorithm for sorting in descending order and also shows the number of interchanges occurred for the given set of integers.

REFERENCE BOOKS

1. Java for Programmers, P. J. Deitel and H. M. Deitel, 10th Edition *Pearson*education.
2. Thinking in Java, Bruce Eckel, *Pearson*Education.
3. Java Programming, D. S. Malik and P. S. Nair, *Cengage*Learning.
4. Core Java, Volume 1, 9th edition, Cay S. Horstmann and G Cornell,*Pearson*.

19MC0001-- GENDER SENSITIZATION

(An Activity-based Course)

B.Tech. II Year II Sem.

L T P C

2 - - 0

Course Objectives

- To develop students sensibility with regard to issues of gender in contemporary India.
- To provide a critical perspective on the socialization of men and women.
- To introduce students to information about some key biological aspects of genders.
- To expose the students to debates on the politics and economics of work.
- To help students reflect critically on gender violence.
- To expose students to more egalitarian interactions between men and women.

Course Outcomes

- Students will have developed a better understanding of important issues related to gender in contemporary India.
- Students will be sensitized to basic dimensions of the biological, sociological, psychological and legal aspects of gender. This will be achieved through discussion of materials derived from research, facts, everyday life, literature and film.
- Students will attain a finer grasp of how gender discrimination works in our society and how to counter it.
- Students will acquire insight into the gendered division of labour and its relation to politics and economics.
- Men and women students and professionals will be better equipped to work and live together as equals.
- Students will develop a sense of appreciation of women in all walks of life.
- Through providing accounts of studies and movements as well as the new laws that provide protection and relief to women, the textbook will empower students to understand and respond to gender violence.

UNIT - I

Understanding Gender: Introduction- Definition of Gender-Basic Gender Concepts and Terminology-Exploring Attitudes towards Gender-Construction of Gender-Socialization: Making Women, Making Men - Preparing for Womanhood. Growing up Male. First lessons in Caste.

UNIT - II

Gender Roles and Relations: Two or Many? -Struggles with Discrimination-Gender Roles and Relations-Types of Gender Roles-Gender Roles and Relationships Matrix-Missing Women-Sex Selection and Its Consequences-Declining Sex Ratio. Demographic Consequences-Gender Spectrum: Beyond the Binary.

UNIT - III

Gender and Labour: Division and Valuation of Labour-Housework: The Invisible Labor- “My Mother doesn’t Work.” “Share the Load.”-Work: Its Politics and Economics -Fact and Fiction. Unrecognized and unaccounted work-Gender Development Issues-Gender, Governance and Sustainable Development-Gender and Human Rights-Gender and Mainstreaming

UNIT - IV

Gender Based Violence: The Concept of Violence- Types of Gender- based Violence - Gender-based Violence from a Human Rights Perspective-Sexual Harassment: Say No! - Sexual Harassment, not Eve-teasing - Coping with Everyday Harassment - Further Reading: “*Chupulu*”.

Domestic Violence: Speaking Out Is Home a Safe Place? -When Women Unite [Film]. Rebuilding Lives. Thinking about Sexual Violence Blaming the Victim-“I Fought for my Life....”

UNIT - V

Gender and Culture: Gender and Film-Gender and Electronic Media-Gender and Advertisement-Gender and Popular Literature- Gender Development Issues-Gender Issues-Gender Sensitive Language-Gender and Popular Literature - Just Relationships: Being Together as Equals Mary Kom and Onler. Love and Acid just do not Mix. Love Letters. Mothers and Fathers. Rosa Parks-The Brave Heart.

TEXT BOOKS:

1. All the five Units in the Textbook, “Towards a World of Equals: A Bilingual Textbook on Gender” written by A.Suneetha, Uma Bhrugubanda, Duggirala Vasanta, Rama Melkote, Vasudha Nagaraj, Asma Rasheed, Gogu Shyamala, Deepa Sreenivas and Susie Tharu and published by Telugu Akademi, Hyderabad,Telangana State in the year 2015.

REFERENCES:

1. Menon, Nivedita. Seeing like a Feminist. New Delhi: Zubaan-Penguin Books, 2012
2. Abdulali Sohaila. “I Fought For My Life...and Won.”Available online at:
3. <http://www.thealternative.in/lifestyle/i-fought-for-my-lifeand-won-sohaila-abdul/>.