KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY

(AN AUTONOMOUS INSTITUTE)

Accredited by NBA & NAAC, Approved by AICTE, Affiliated to JNTUH, Hyderabad

B.Tech. II Year I & II Semester Course Structure for (KR21)
Common to CSE, IT, CSE (AI&ML) and CSE (DS)
Applicable from 2021-22 Admitted Batch

S. No.	Course Code	Course Title	L	T	P	Credits
1	21CS301BS	Mathematical Foundations of Computer Science	3	1	0	4
2	21CS302PC	Computer Organization and Architecture	3	0	0	3
3	21CS303PC	Data Structures Through C++	3	0	0	3
4	21CS304ES	Analog and Digital Electronics	3	0	0	3
5	21CS305PC	Introduction to Machine Learning	3	1	0	4
6	21CS306PC	Machine Learning Using Python Lab	0	0	2	1
7	21CS307PC	Data Structures through C++ Lab	0	0	4	2
8	21CS308ES	Analog and Digital Electronics Lab	0	0	2	1
	*21MC309HS	Gender Sensitization Lab	0	0	3	0
		Total Credits	**15**	**2**	**11**	**21**

S. No.	Course Code	Course Title	L	T	P	Credits
1	21MA406BS	Statistical Methods for Computer Science	3	0	0	3
2	21CS401PC	Java Programming	3	1	0	4
3	21CS402PC	Database Management Systems	3	0	0	3
4	21CS403PC	Operating Systems	3	0	0	3
5	21CS404PC	Automata Theory and Compiler Design	3	0	0	3
6	21CS405PC	Java Programming Lab	0	0	4	2
7	21CS406PC	Database Management Systems Lab	0	0	3	1.5
8	21CS407PC	Operating Systems Lab	0	0	3	1.5
	*21MC408HS	Constitution of India	3	0	0	0
		Total Credits	**18**	**1**	**10**	**21**

*Mandatory Course

Noted and Approved

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY

(AN AUTONOMOUS INSTITUTE)
Accredited by NBA & NAAC, Approved by AICTE, Affiliated to JNTU, Hyderabad

B. Tech. in CSE (DATA SCIENCE)
II Year I Semester Course Syllabus (KR21)

MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE (21CS301BS)

L	T	P	C
3	1	0	4

Prerequisite/Corequisite:

1. MA201BS–Linear Algebra and Differential Equations Course.

Course Objectives: The course will help to

1. Learn the concepts of mathematical logic.
2. Understand the operations associated with sets, functions, and relations.
3. Understand the appropriate set, function, or relation model, and interpret the associated operations and terminology in context.
4. Generate functions and recurrence relations.
5. Solve the problems using Graph Theory.

Course Outcomes: After learning the concepts of this course, the student is able to

1. Identify mathematical logic to solve problems.
2. Interpret sets, relations, functions, and discrete structures.
3. Apply logical notation to define and reason about fundamental mathematical concepts such as sets, relations, and and functions.
4. Formulate problems and solve recurrence relations.
5. Apply knowledge to design and solve real-world problems by using graphs.

UNIT - I

Mathematical logic: Introduction, Statements and Notation, Connectives, Normal Forms, Theory of Inference for the Statement Calculus, The Predicate Calculus, Inference Theory of the Predicate Calculus.

UNIT - II

Set theory: Introduction, Basic Concepts of Set Theory, Representation of Discrete Structures, Relations and Ordering, Functions.

Algebraic Structures: Introduction, Algebraic Systems, Semi groups and Monoids, Groups, Lattices as Partially Ordered Sets, Boolean algebra.

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, The Principle of Inclusion-Exclusion.

UNIT - IV

Recurrence Relations: Generating Functions of Sequences, Calculating Coefficients of generating functions, Recurrence relations, solving recurrence relations by substitution and Generating functions, The method of Characteristic roots, Solutions of Inhomogeneous Recurrence Relations, Master's Theorem and its applications.

UNIT - V

Graphs: Basic Concepts, Isomorphisms and Subgraphs, Planar Graphs, Euler's Formula, Multigraphs and Euler Circuits, Hamiltonian Graphs, Chromatic Numbers, The Four-Color Problem.

TEXT BOOKS:

1. Discrete Mathematical Structures with Applications to Computer Science - J.P. Tremblay, R. Manohar, McGraw Hill education (India) Private Limited, 2017.
2. Discrete Mathematics for Computer Scientists & Mathematicians - Joe L. Mott, Abraham Kandel, Theodore P. Baker,2nd Edition, Pearson,2015.

REFERENCE BOOKS:

1. Discrete Mathematics and its Applications - Kenneth H. Rosen,7thEdition, McGraw Hill education (India) Private Limited, 2017.
2. Discrete Mathematics - D.S. Malik & M.K. Sen, Revised Edition, Cengage Learning, 2012.
3. Elements of Discrete Mathematics - C. L. Liu and D. P. Mohapatra,4thEdition, McGraw Hill education (India) Private Limited,2017.
4. Discrete Mathematics with Applications - Thomas Koshy, Elsevier, 2004.
5. Discrete and Combinatorial Mathematics - 6R. P. Grimaldi, Pearson, 2006.

Noted and Approved

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY

(AN AUTONOMOUS INSTITUTE)

Accredited by NBA & NAAC, Approved by AICTE, Affiliated to JNTU, Hyderabad

B. Tech. in CSE (DATA SCIENCE)
II Year I Semester Course Syllabus (KR21)

COMPUTER ORGANIZATION AND ARCHITECTURE (21CS302PC)

	L	T	P	C
Prerequisite/ Corequisite:	**3**	**0**	**0**	**3**

1. 21CS304ES – Analog and Digital Electronics course.

Course Objectives: The course will help to

1. Understand the basic components of a computer.
2. Understand the instruction sets, instruction formats and various addressing modes.
3. Explore the Memory organization and I/O Organization.
4. Understand the concept of pipelining techniques.
5. Learn the architecture of CPU, GPU and the basic concepts of CUDA programming.

Course Outcomes: After learning the contents of this course the student is able to

1. Understand the basics of instructions sets and their impact on processor design.
2. Demonstrate an understanding of the design of the functional units of a digital computer system.
3. Evaluate cost performance and design trade-offs in designing and constructing a computer processor including memory.
4. Design a pipeline for consistent execution of instructions with minimum hazards.
5. Analyze high performance computing techniques and its efficiency.

Unit-I

Digital Computers: Introduction, block diagram of digital computer, definition of computer organization and architecture.

Basic Computer Organization and Design: Computer Registers Instruction codes, Computer instructions, Memory Reference Instructions, Register reference instructions, Input – Output instructions, Timing and Control, Instruction cycle, Interrupt cycle.

Unit-II

Instruction Set Architecture: General Register Organization, Instruction Formats, Addressing modes, Data Transfer and Manipulation, Program Control, Computer Arithmetic

Introduction to Computer Arithmetic: Addition and subtraction, multiplication Algorithms, Division Algorithms, Floating–point Arithmetic operations.

Unit-III

Memory Architecture: Memory Hierarchy-Main Memory, Auxiliary memory, Associate Memory, Cache Memory Mapping and Concept of Virtual Memory- Associate mapping, Direct mapping, Set Associate mapping.

Input-Output Organization: Input-Output Interface, Asynchronous data transfer, Modes of Transfer, Priority Interrupt, Direct memory Access.

Unit-IV

Overview of Pipelining Techniques: Parallel Processing, Pipelining, Arithmetic Pipeline, Instruction Pipeline, RISC Pipeline, Design Issues, Hazards: Structural Hazards, Data Hazards and Control Hazards, Static Branch Prediction, Dynamic Branch Prediction.

Introduction to Processor Architecture: CISC Characteristics, RISC Characteristics, Differences between CISC and RISC Characteristics, advantages and disadvantages of CISC over RISC.

Unit-V

Introduction to High Performance Computing:

Goals of Parallel Computing, Speed or Parallelism, CPU vs High performance computing, Architecture of a Modern GPU, Parallel Programming Languages and Models, GPU Computing, Data Parallelism.

Introduction to CUDA Programming: CUDA program structure, Vector Addition Kernel, Matrix Multiplication Kernel, Device Global Memory and Data Transfer, Kernel Functions and Threading, CUDA Thread Organization and Synchronization, Querying Device properties, CUDA Memories Types.

TEXT BOOKS:

1. M .Morris Mano, "Computer System Architecture", 3rd Edition,Pearson, 2017
2. Jason Sanders & Edward Kandrot, "CUDA by Example: Introduction to General-Purpose GPU Programming", 1st Edition, Addison-Wesley 2010.

REFERENCE BOOKS:

1. Carl Hamacher, Zvonko Vranesic, Safwat Zaky: "Computer Organization", 5th Edition, TMH, 2002.
2. Shane Cook, "CUDA Programming: A Developer's Guide to Parallel Computing with GPUs" 1st Edition, Morgan Kaufmann, 2012.

Noted and Approved

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY

(AN AUTONOMOUS INSTITUTE)
Accredited by NBA & NAAC, Approved by AICTE, Affiliated to JNTU, Hyderabad

B. Tech. in CSE (DATA SCIENCE)
II Year I Semester Course Syllabus (KR21)

DATA STRUCTURES THROUGH C++ (21CS303PC)

L	T	P	C
3	0	0	3

Prerequisite/Corequisite:

1. PP102ES - Programming for Problem Solving Course

Course Objectives: The course will help to

1. Learn the principles of data abstraction, inheritance, polymorphism virtual functions and exception handling
2. Understand the behavior of data structures such as trees, hash tables, search trees, Graphs and their representations.
3. Choose an appropriate data structure for a specified application.
4. Understand and analyze various searching and sorting algorithms.
5. Implement ADTs such as lists, stacks, queues, trees, graphs, search trees in C++ to solve problems.

Course Outcomes: After learning the concepts of this course the student is able to

1. Demonstrate object-oriented programming concepts like Encapsulation, Inheritance, polymorphism, and abstraction and utilize them to create basic program designs.
2. Use template classes in C++ to understand and Implement Exception handling mechanisms.
3. Construct programs using a variety of data structures such as stacks, queues, hash tables, binary trees, search trees, heaps, graphs, and B-trees.
4. Interpret appropriate data structures to represent data items in real world problems.
5. Design and implement String Matching Algorithms to apply them in real world.

UNIT- I

C++ Programming Concepts: Review of C, input and output in C++, functions in C++- value parameters, reference parameters, Parameter passing, function overloading, function templates, arrays, pointers, new and delete operators, class and object, access specifiers, friend functions, constructors and destructor, Operator overloading, class templates.

UNIT- II

Inheritance and Polymorphism, Exception Handling, throwing an exception, the try block, catching an exception, exception objects, exception specifications, catching all exceptions, Introduction to linked lists, stacks, queues and applications of stacks.

UNIT- III

Trees – definition, terminology, Binary trees-definition, Properties of Binary Trees, Binary Tree ADT, representation of Binary Trees-array and linked representations, Binary Tree traversals, threaded binary trees, Binary Search Tree ADT, Definition, Operations- Searching, Insertion, Height, Traversals and Deletion, Balanced search trees-AVL Trees-Definition and Examples.

UNIT- IV

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

Priority Queues –Definition and applications, Max Priority Queue ADT-implementation-Max Heap-Definition, Insertion into a Max Heap, Deletion from a Max Heap.

Sorting- Radix Sort, Counting Sort, Quick sort, Heap Sort, Merge sort, Comparison of Sorting methods.

UNIT- V

Graphs–Definitions, Terminology, Applications and more definitions, Properties, Graph ADT, Graph Representations- Adjacency matrix, Adjacency lists, Graph Search methods - DFS and BFS, Spanning Trees and Minimum Cost Spanning Trees.

String Matching- Introduction, String matching algorithms-Brute force, the Boyer–Moore algorithm, the Knuth- Morris- Pratt algorithm.

TEXT BOOKS:

1. The Complete Reference C++ - Herbert Schildt, 4th Edition, Tata Mc Graw Hill.
2. Data structures, Algorithms and Applications in C++ - Sartaj Sahni, 2nd Edition, Universities Press.
3. Data structures and Algorithms in C++ - Adam Drozdek, 4th Edition, Cengage learning.

REFERENCES BOOKS:

1. The C++ Programming Language, 3rd Edition, B. Stroutstrup, Pearson Education.
2. Data structures with C++, J. Hubbard, Schaum's outlines, TMH.
3. Data structures and Algorithm Analysis in C++ - M. A. Weiss, 3rd Edition, Pearson

Noted and Approved

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY

(AN AUTONOMOUS INSTITUTE)
Accredited by NBA & NAAC, Approved by AICTE, Affiliated to JNTU, Hyderabad

B. Tech. in CSE (DATA SCIENCE)
II Year I Semester Course Syllabus (KR21)

ANALOG AND DIGITAL ELCTRONICS (21CS304ES)

L	T	P	C
3	0	0	3

Prerequisite/ Corequisite:

1. BE105ES - Basic Electrical and Electronics Engineering Course

Course Objectives: The course will help to

1. Understand and learn the applications of components
2. Understand the basic design techniques of electronic components
3. Know the basic techniques of gate level minimization
4. Understand the concepts of combinational logical circuits
5. Introduce the concepts of sequential circuits

Course Outcomes: After learning he concepts of this course the student is able to

1. Know the characteristics of various components.
2. Understand the utilization of components.
3. Design and analyze small signal amplifier circuits.
4. Learn Postulates of Boolean algebra and to minimize combinational functions
5. Design and analyze combinational and sequential circuits

UNIT– I

Diodes and Applications: Diode resistance, Diffusion capacitance, Effect of Temperature, Diode Switching times, Tunnel diodes, photo diode.
Diodes Applications: clipping circuits, comparator, Half wave and full wave diodes.

UNIT – II

Transistors and Biasing: Operating point, Self-Bias, Thermal runaway and Stability, RC coupled Amplifier, Two cascaded CE Amplifier, frequency response of CE amplifier, Overview of FET, CS and CD amplifiers, Low frequency response of CS and CD amplifiers.
BJT'S: Transistor configurations (CB, CE, CC), Comparison of Configurations.

UNIT– III

Number Systems: Binary Numbers, Octal, Hexadecimal and other base numbers, complements, Boolean algebra and logic gates, Exclusive-OR function
Gate – Level Minimization: Canonical and standard forms, The K-Map Method, Two Variable Map, Three-Variable Map, Four-Variable Map, Five-Variable Map, sum of products, product of sums simplification, don't care conditions, NAND and NOR implementation.

UNIT– IV

Combinational Circuits (CC): Combinational circuit for different code converters and Binary Adder(Half adder & Full Adder) - Subtractor (Half Sub & Full Sub), Parallel Adder, Decimal Adder, Binary Multiplier, Magnitude Comparator, Decoders, Encoders, Multiplexers, and De-multiplexers.

UNIT– V

Sequential Circuits:(SC), Latches, Flip-flops (SR,D, JK, T), Master-Slave JK Configuration, Conversion from one type of Flip-Flop to another, Registers, Shift registers, Ripple counters, Synchronous counters, Mod-N counter.
IC Applications: IC 555 Timer, applications - Multi Vibrators, Trigger circuits.

TEXT BOOKS:

1. Integrated Electronics: Analog and Digital Circuits and Systems, 2/e, Jaccob Millman, Christos Halkias and Chethan D.Parikh,TataMcGraw-HillEducation,India,2010.
2. Analog and Digital Electronics - S Salivahanan & S Arivazhagan, Mcgraw hill, 2019.
3. Digital Design,5/e,Morris Mano and Michael D.Cilette, Pearson Education, 2011.
4. Op-Amps & Linear ICs, Ramakanth A. Gayakwad, PHI.

REFERENCE BOOKS:

1. Electronic Devices and Circuits, Jimmy J Cathey, Schaum' soutline series, 1988.
2. Digital Principles,3/e, RogerL. Tokheim, Schaum'soutline series, 1994.
3. Operational Amplifiers & Linear Integrated Circuits, R.F. Coughlin & Fredrick F. Driscoll, PHI.

Noted and Approved

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY

(AN AUTONOMOUS INSTITUTE)
Accredited by NBA & NAAC, Approved by AICTE, Affiliated to JNTU, Hyderabad

B. Tech. in CSE (DATA SCIENCE)
II Year I Semester Course Syllabus (KR21)

INTRODUCTION TO MACHINE LEARNING (21CS305PC)

L	T	P	C
3	1	0	4

Prerequisites/Corequisites:

1. PP204ES– Python Programming Course.

Course Objectives: The course will help to

1. Introduce basic concepts of Probability and Machine Learning.
2. Introduce Descriptive Statistics and data analysis along with visualization.
3. Gain knowledge on Regression analysis.
4. Gain knowledge on Classification Techniques.
5. Gain experience on non-Parametric machine learning algorithms and SVMs.

Course Outcomes: After learning the concepts of this course, the student is able to

1. Understand the basic concepts of Probability and Machine Learning
2. Develop the Statistics and data analysis along with visualization.
3. Implement the different types of regression models.
4. Implement the classification model for categorical data.
5. Analyze and develop the non-Parametric models.

UNIT –I

Introduction-Probability, Machine Learning and Data Science, Use Machine Learning.
Types of Machine Learning Systems: supervised, unsupervised, semi-supervised, Reinforcement, Batch and Online Learning, Main Challenges of Machine Learning.

UNIT – II

Descriptive Statistics: Data representation, types of data- nominal, ordinal, interval and continuous, central tendency- calculating mean mode median, mean vs. median, variability, variance, standard deviation, Mean Absolute Deviation using sample dataset, finding the percentile, inter quartile range, Box Plot, Outlier, whisker, calculating correlation, covariance, causation, Exploratory data analysis, Data preparation and preprocessing, visualization and its tools.

UNIT – III

Regression: Introduction to Regression analysis, measure of linear relationship, Regression with stats models, determining coefficient, meaning and significance of coefficients, coefficient calculation with least square method, Types of regression, Simple Linear Regression, Using Multiple features, Polynomial Regression, Metrics for Regression: MSE, RMSE,MAE.

UNIT – IV

Classification: Classification problem, Probability based approach, Logistic Regression- log-odd, sigmoid transformation, Metrics: Confusion Matrix, Accuracy, Error Rate, Precision, Recall, ROC curve, F1 score, and introduction to gradient descent.

UNIT – V

Nonparametric & SVM classification: About Non parametric classification, Decision Trees: Entropy, Gain ratio, Information Gain, Splitting criteria.
Ensemble Method: Introduction to Random Forest, Accuracy measure & performance.
Instance based learning- Introduction, KNN algorithm, Distance measures, model building, locally weighted regression, radial basis functions, SVM classifier, hyper-plane, slack variables, geometric transformation kernel trick, kernel transformation.

TEXT BOOKS:

1. Hands-On Machine Learning with Scikit - Learn and Tensor Flow –Aurélien Géron, O'Reilly Media, 2017.
2. Practical Python Data Visualization: A Fast Track Approach to Learning Data Visualization with Python, Ashwin Pajankar, A Press.
3. Python: End-to-end Data Analysis - Phuong Vo.T.H, Martin Czygan, Ivan Idris, Magnus Vilhelm Persson, Luiz Felipe Martins, Packet Pub.

REFERENCE BOOKS:

1. Machine Learning in Action, Peter Harrington, Manning Publications.
2. Python for Data Analysis–wes McKinny, 2nd Edition, O'REILLY Publications.

Noted and Approved

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY

(AN AUTONOMOUS INSTITUTE)
Accredited by NBA & NAAC, Approved by AICTE, Affiliated to JNTU, Hyderabad

B. Tech. in CSE (DATA SCIENCE)
II Year I Semester Course Syllabus (KR21)

MACHINE LEARNING USING PYTHON LAB (21CS306PC)

L	T	P	C
0	0	2	1

Prerequisite/Corequisite:

1. PP204ES -Python Programming Course
2. 21CS305PC -Introduction to Machine Learning Course

Course Objectives: The course will help to

1. Perform exploratory data analysis on the given data sets.
2. Provide hands-on experience with descriptive statistics, data analysis, and visualization.
3. Implement Regression models on given datasets.
4. Build Classification models.
5. Implement models on SVMs.

Course Outcomes: After learning the concepts of this course the student is able to

1. Execute the basic concepts of Probability and Machine Learning.
2. Apply the Statistics and data analysis along with visualization.
3. Implement the different types of regression models.
4. Explore the classification model for categorical data.
5. Design and develop the non-Parametric models.

LIST OF EXERCISES

1. Apply central tendency and variability on given dataset
2. Perform EDA on given dataset and prepare dataset to train and test ML model
3. Build a linear regression model using python on given data set by
    a. Prepare the data for ML model.
    b. Splitting Training data and Test data.
    c. Evaluate the model (intercept and slope).
    d. Visualize the training set and testing set using Matplotlib, Seaborn.
    e. Predicting the test set result.
    f. Compare actual output values with predicted values.
4. Implement regression model without using ML libraries.
5. Apply various regression models on given dataset and find a proper model for prediction with minimal errors.

6. Implement a logistic regression model on given dataset and check the accuracy for test dataset.
7. Build a decision tree model for given dataset to predict the target with best accuracy.
8. Implement KNN model to classify the target in given dataset.
9. Demonstrate regression and classification metrics using sample data.
10. Build SVM model with various kernels and select best kernel for given dataset.
11. Build Random Forest model and apply on given dataset. Evaluate the model with suitable metrics.

TEXT BOOKS:

1. Hands-On Machine Learning with Scikit-Learn and Tensor Flow,AurélienGéron - O'Reilly Media, 2017.
2. Practical Python Data Visualization: A Fast Track Approach To Learning Data Visualization With Python, Ashwin Pajankar, A Press
3. Python: End-to-end Data Analysis, Phuong Vo.T.H, Martin Czygan, Ivan Idris, Magnus VilhelmPersson, Luiz Felipe Martins, Packet Pub

REFERENCE BOOKS:

1. Peter Harrington, Machine Learning in Action, Manning Publications.
2. Python for Data Analysis by wes McKinny, 2nd Edition, O'REILLY Publications.

Noted and Approved

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY

(AN AUTONOMOUS INSTITUTE)
Accredited by NBA & NAAC, Approved by AICTE, Affiliated to JNTU, Hyderabad

B. Tech. in CSE (DATA SCIENCE)
II Year I Semester Course Syllabus (KR21)

DATA STRUCTURES THROUGH C++ LAB (21CS307PC)

L	T	P	C
0	0	4	2

Prerequisites/Corequisites:

1. PP102ES - Programming for Problem Solving Course
2. PP106ES - Programming for Problem Solving Lab Course
3. 21CS303PC - Data Structures Through C++Course

Course Objectives: The course will help to

1. Introduce various concepts of C++ programming language.
2. Understand data structures such as stacks and queues.
3. Explore trees and graphs.
4. Introduce searching and sorting algorithms.
5. Implement Data Structures in real world problems.

Course Outcomes: **After learning the concepts of this course,student is able to**

1. Design and implement C++ programs for computing real life applications.
2. Understand basic elements of control statements, arrays, functions, pointers and strings, and data structures like stacks, queues and linked lists.
3. Implement searching and sorting algorithms.
4. Select appropriate Data Structure to solve the problems.
5. Test and debug the application.

List of Programs:

1. Write a program to perform the following operations on singly linked list:
   a. Creation b. Insertion c. Deletion d. Traversal e. Length of the linked list
2. Write a program to perform the following operations on doubly linked list:
   a. Creation b. Insertion c. Deletion d. Traversal e. Print the list in reverse
3. Write a program to perform the following operations on circular linked list:
   a. Creation b. Insertion c. Deletion d. Traversal e. Count of nodes
4. **Stacks**
   a. Write a program that implement stack (its operations)using
      i) Arrays ii) linked lists
   b. Infix to Postfix Conversion, postfix evaluation.

c. Check for balanced parentheses in an expression

d. String Palindrome

5. **Queues**

a. Write a program that implement Queue, Circular Queue (its operations)using

i) Arrays ii) linked lists

b. Queue implementation using two stacks.

c. Check if a queue can be sorted into another queue using a stack

6. **Searching and Sorting**

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

i) Merge sort ii) Quick sort iii) Radix sort

b. Write a program to perform the following searching operations for a Key value in a given list of integers:

i) Linear search ii) Binary search

c. Write a program to implement Linear Probing (closed hashing technique).

7. **Binary Trees**

a. Write a program that implement Binary tree(it 's operations) using

i) array's ii) linked lists

b. Binary tree traversal methods

i) In order ii) Pre order iii) Post order iv) Level order

c. Second minimum node in a Binary tree.

d. Write a program to print the height of a Binary tree.

e. Write a program that checks a tree is a sub tree of another given tree.

f. Write a program to print the number of nodes in a given level k.

g. Write a program that checks a tree is a Symmetric Tree or not.

8. Binary Search Trees

a. Write a program to check if a binary tree is BST or not.

b. Write a program to perform the following operations on BST

i) Insertion ii) Deletion iii) Search

c. Write a program to print the Minimum and maximum element.

d. Write a program to print the nodes in a given range.

9. Graphs

a. Write a program that implements graph using

i) Arrays ii) Linked lists

b. Graph traversal methods

i) Depth first search ii) Breadth first Search

TEXT BOOKS:

1. The Complete Reference C++ - Herbert Schildt, 4th Edition, Tata Mc Graw Hill.
2. Data structures, Algorithms and Applications in C++ - Sartaj Sahni, 2nd Edition, Universities Press.
3. Data structures and Algorithms in C++ - Adam Drozdek, 4th Edition, Cengage learning.

REFERENCE BOOKS:

1. The C++ Programming Language, 3rd Edition, B. Stroutstrup, Pearson Education.
2. Data structures with C++, J. Hubbard, Schaum's outlines.

Noted and Approved

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY

(AN AUTONOMOUS INSTITUTE)
Accredited by NBA & NAAC, Approved by AICTE, Affiliated to JNTU, Hyderabad

B. Tech. in CSE (DATA SCIENCE)
II Year I Semester Course Syllabus (KR21)

ANALOG AND DIGITAL ELECTRONICS LAB (21CS308ES)

L	T	P	C
0	0	2	1

Prerequisite/Corequisite:

1. BE105ES - Basic Electrical and Electronics Engineering Course
2. EE109ES - Basic Electrical and Electronics Engineering Lab Course
3. 21CS304ES – Analog and Digital Electronics

Course Objectives: The course will help to

1. Introduce components such as diodes, BJTs and FETs.
2. Know the applications of components.
3. Give understanding of various types of amplifier circuits
4. Learn basic techniques for the design of digital circuits and fundamental concepts used in the design of digital systems.
5. Understand the concepts of combinational logic circuits and sequential circuits.

Course Outcomes: After learning the concepts of this course the student is able to

1. Know the characteristics of various components.
2. Understand the utilization of components.
3. Design and analyze small signal amplifier circuits.
4. Know about the logic families and realization of logic gates.
5. Postulates of Boolean algebra and to minimize combinational functions.
6. Design and analyze combinational and sequential circuits.

List of Experiments

1. P-N Junction Diode characteristics
2. Full Wave Rectifier with & without filters
3. Input and Output characteristics of CE configuration
4. Input and Output characteristics of CB configuration
5. Input and Output characteristics of CS configuration
6. Frequency response of Common Emitter amplifier
7. Frequency response of Common Source amplifier
8. Realization of logic gates using DTL, TTL, ECL, etc.
9. Realization of Boolean Expressions using Gates

10. Design and realization logic gates using universal gates
11. generation of clock using NAND / NOR gates
12. Design a 4 – bit Adder / Subtractor.
13. Design and realization a Synchronous and Asynchronous counter using flip-flops
14. Design of 4bit Shift register

TEXT BOOKS:

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

Noted and Approved

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY

(AN AUTONOMOUS INSTITUTE)
Accredited by NBA & NAAC, Approved by AICTE, Affiliated to JNTU, Hyderabad

B. Tech. in CSE (DATA SCIENCE)
II Year I Semester Course Syllabus (KR21)

GENDER SENSITIZATION LAB (*21MC309HS)

L	T	P	C
0	0	3	0

Prerequisite/Corequisite: Nil

Course Objectives: The course will help to

1. Develop students' sensibility with regard to issues of gender in contemporary India.
2. Provide a critical perspective on the socialization of men and women.
3. Introduce students to information about some key biological aspects of genders.
4. Expose the students to debates on the politics and economics of work.
5. Encourage pupils to think critically about gender violence.

Course Outcomes: After learning the concepts of this course the student is able to

1. Outline important issues related to gender in contemporary India.
2. Illustrate basic dimensions of the biological, sociological, psychological and legal aspects of gender.
3. Interpret how gender discrimination works in our society and how to counter it.
4. Demonstrate the gendered division of labor and its relation to politics and economics.
5. Develop a sense of appreciation of women in all walks of life.

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. Un recognized and Un accounted 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.

Note: Since it is Interdisciplinary Course, Resource Persons can be drawn from the fields of English Literature or Sociology or Political Science or any other qualified faculty who has expertise in this field from engineering departments. Classes will consist of a combination of activities: dialogue-based lectures, discussions, collaborative learning activities, group work and in-class assignments. Apart from the above prescribed book, Teachers can make use of any authentic materials related to the topics given in the syllabus on “Gender”.

TEXT BOOK:

1. The Textbook Towards a World of Equals:A Bilingual Textbook onGender -A.Suneetha, Uma Bhrugubanda, DuggiralaVasanta, Rama Melkote,VasudhaNagaraj, AsmaRasheed, GoguShyamala, DeepaSreenivas and Susie Tharu, TeluguAkademi, Telangana Government in 2015.

Noted and Approved

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY

(AN AUTONOMOUS INSTITUTE)
Accredited by NBA & NAAC, Approved by AICTE, Affiliated to JNTU, Hyderabad

B.Tech. in CSE (DATA SCIENCE)
II Year II Semester Course Syllabus (KR21)

STATISTICAL METHODS FOR COMPUTER SCIENCE (21MA406BS)

L	T	P	C
3	0	0	3

Prerequisite/Corequisite:

1. PP102ES -Programming for Problem Solving Course

Course Objectives: The course will help to

1. Learn Random variables for studying the importance of Probability distribution functions.
2. Study discrete, continuous, sampling distributions and estimate unknown parameters.
3. Draw inferences using testing of hypothesis.
4. Identify strength and relation between the variables by using correlation and regression
5. Introduce back propagation algorithm with gradient descent strategies.

Course Outcomes: After learning the concepts of this course student is able to

1. Distinguish between random variables pertaining to discrete/continuous distributions.
2. Illustrate the ideas of various distribution and Estimations.
3. Use hypothesis testing for drawing inferences.
4. Understand concept of ANOVA and determine the relation between variables using correlation and regression.
5. Estimate the parameters for regression using Back-propagation algorithm.

UNIT– I: Probability and Random Variables

Probability: Basics of Probability, addition theorem, Conditional Probability, multiplication theorem, Bayes theorem (without proof).
Random Variable: Definition, types, discrete random variable, probability mass function, discrete distribution function, Continuous random variable, probability density function, Continuous distribution function. Mean, Variance.

UNIT–II: Distributions and Estimations

Distributions: Binomial distribution, Poisson distribution, Normal distribution, Population, Sample, Types of Sampling, Parameters and Statistics, Sampling distribution of mean, Standard error of means, Central limit theorem, Student t- distribution, Chi-square distribution, F distribution.

Estimations: Types of estimation, properties of estimators, Confidence interval for means and proportions, Maximum error of estimates, Sample size, and Maximum likelihood estimations.

UNIT– III: Testing of Hypothesis

Null hypothesis, Alternate hypothesis, level of Significance, Type I & II error, Critical Region, Test concerning single mean, two means, test concerning single proportion, two proportions for large samples, Test concerning single mean, two means, test concerning single, two proportions for small samples, Chi-square d-test, F-test.

UNIT-IV: ANOVA, Correlation and Regression

ANOVA one-way classification, ANOVA two-way classification..

Definition of Correlation, Karl Pearson method for finding Correlation Coefficient, Definition of Regression, Simple linear regression, multiple linear regression & Logistic regression.

UNIT– V

Basics of Back Propagation: Introduction to Neural Networks – Back propagation - Setup and Initialization Issues - The Vanishing and Exploding Gradient Problems - Gradient-Descent Strategies - Batch Normalization - Practical Tricks for Acceleration and Compression.

TEXTBOOKS:

1. Probability and statistics for engineers and scientists – Sheldon MRoss, 5thEdition, Academic Press, 2014.
2. Fundamentals of Mathematical statistics – SC Gupta and VK Kapoor, Khanna publications.
3. Neural Networks and Deep Learning - Charu C. Aggarwal, Springer International Publishing AG.

REFERENCE BOOKS:

1. Fundamentals of Probability and Statistics for Engineers - T.T. Soong, John Wiley & Sons Ltd, 2004.
2. Probability & Statistics for Engineers & Scientists - Ronald E. Walpole, Raymond H. Myers, Sharon L. Myers, Keying Ye, 9th Ed. Pearson Publishers.
3. Deep Learning, Ian Good fellow, Yoshua Bengio, Aaron Courville, MIT Press.

Noted and Approved

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY

(AN AUTONOMOUS INSTITUTE)
Accredited by NBA & NAAC, Approved by AICTE, Affiliated to JNTU, Hyderabad

B.Tech. in CSE (DATA SCIENCE)
II Year II Semester Course Syllabus (KR21)

JAVA PROGRAMMING (21CS401PC)

L	T	P	C
3	1	0	4

Prerequisite/Corequisite:

1. PP102ES - Programming for Problem Solving Course

Course Objectives: The course will help to

1. Learn the object-oriented programming concepts.
2. Understand object-oriented programming concepts, and apply them in solving problems.
3. Introduce the principles of inheritance and polymorphism and demonstrate how they relate.
4. Understand the concepts of abstract classes, packages and interfaces.
5. Introduce the concepts of exception handling and multithreading.

Course Outcomes: After learning the concepts of this course the student is able to

1. Use concepts of OOPs such as data abstraction, inheritance, polymorphism, encapsulation and method overloading principles in structuring computer applications for solving problems.
2. Design and Implement packages.
3. Apply the concepts of I/O streams and exception handling in a given real time problem.
4. Analyze appropriate collections to solve programming problems.
5. Build java applications to utilize advanced mechanisms like multi-threading, database connectivity.

UNIT – I

Object-Oriented Thinking- A way of viewing the world –Methods, Responsibilities, Classes and Instances, Class Hierarchies- Inheritance, Method binding, Encapsulation, Abstraction. Summary of Object - Oriented concepts. Java buzzwords, An Overview of Java, Data types, Variables and operators, expressions, control statements, Introducing classes, Methods and Classes.

Arrays - One Dimensional Array, Second Dimensional Array, Jagged Arrays, String handling. String Builder, String Buffer, String Tokenizer(), StringAPI'slikelength(),substring(),charAt(),indexOf(),replace(),toCharArray()

Inheritance– Inheritance concept, Inheritance basics, Member access, Constructors, Types of constructors, Creating Multilevel hierarchy, super uses, this uses, static uses, static and instance blocks, using final with Inheritance, Polymorphism Method overriding method overloading, abstract classes, Object class, forms of inheritance-specialization, benefits of inheritance, costs of inheritance.

UNIT - II

Introduction to 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, Reading and Writing Objects to a file, Serialization, De-Serialization, transient keyword, transient and static variables.

UNIT - III

Exception handling - Fundamentals of exception handling, Exception types, Termination or resumptive models, Uncaught exceptions, using try and catch, multiple catch clauses, nested try statements, throw, throws and finally, built- in exceptions, creating own exception subclasses.

Multithreading- Differences between thread-based multitasking and process-based multitasking, Java thread model, thread life cycle, different ways of creating threads, thread priorities, synchronized keyword and synchronized block, inter thread communication, Producer Consumer Problem. Thread class APIs.

UNIT - IV

The Collections Framework (java.util)- Collections overview, Collection Interfaces List, Map and Set, The Collection classes- Array List, Linked List, Hash Set, Tree Set, Priority Queue, Array Deque. Accessing a Collection via an Iterator, Using an Iterator, The for-Each alternative, Map Interfaces and Classes, Comparators, Arrays, Stack, Vector, Enumerations, Auto boxing, Scanner class.

UNIT - V

An overview of Advanced JAVA: Introduction to JDBC, Types of JDBC Drivers, Connectivity with Oracle/ MySQL, Driver Manager API, Connection API, Statement API- Prepared Statement, invoking stored procedure using Callable Statement, Result Set, Properties of Result Set. Transaction Management using JDBC API, auto commit, save point and rollback methods, Exception Handling in JDBC.

TEXT BOOKS:

1. Java The Complete reference- Herbert Schildt, 9th Edition, McGraw Hill Education (India)Pvt. Ltd,2014.
2. Java Database Best Practices- George Reese, O'Reilly Media, 2003.

REFERENCE BOOKS:

1. An Introduction to programming and OO design using Java- J.Nino and FA.Hosch, John Wiley &sons, 2008.
2. Introduction to Java programming - Y. Daniel Liang, Pearson Education, 2012.

Noted and Approved

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY

(AN AUTONOMOUS INSTITUTE)
Accredited by NBA & NAAC, Approved by AICTE, Affiliated to JNTU, Hyderabad

B.Tech. in CSE (DATA SCIENCE)
II Year II Semester Course Syllabus (KR21)

DATABASE MANAGEMENT SYSTEMS (21CS402PC)

L	T	P	C
3	0	0	3

Prerequisites/Corequisites:

1. PP102ES - Programming for Problem Solving Course
2. 21CS303PC - Data Structures Through C++ Course

Course Objectives: The course will help to

1. Understand the concepts of database management systems and ER Diagrams.
2. Gain the knowledge on relational model and integrity constraints.
3. Understand SQL queries and Normalization of relations.
4. Understand the transaction management and concurrency control techniques.
5. Understand the concepts of Triggers, Stored Procedures and DCL Commands.

Course Outcomes: After learning the concepts of this course the student is able to

1. Recognize the basic concepts and the applications of database systems.
2. Design ER-models to represent simple database application scenarios.
3. Demonstrate SQL queries and apply Normalization techniques.
4. Summarize the usage of different concurrency control protocols.
5. Demonstrate the role of DBA using DCL commands.

UNIT - I

Database System Applications: File Systems versus a DBMS**,** the Data Model, Levels of Abstraction in a DBMS, Data Independence, Structure of a DBMS, Advantages of DBMS

Introduction to Database Design: Database Design and ER Diagrams, Entities, Attributes, and Entity Sets, Relationships and Relationship Sets, Additional Features of the ER Model, Conceptual Design With the ER Model.

UNIT – II

Introduction to the Relational Model: Data Definition Language, Integrity constraint over relations, Types of Integrity Constraints Domain Constraint-String, character, Integer, date, Entity Integrity Constraint-, Primary Key, Referential Integrity Constraint-Foreign Key, Other Key Constraint – NULL,NOT NULL,CHECK and etc, querying relational data, logical database design, introduction to views, destroying/altering tables and views.

UNIT - III

Introduction to SQL: Select Queries, Constraints: Data Manipulation Language –Insert, Delete, Update, form of basic SQL query, UNION, INTERSECT, and EXCEPT, Nested Queries, Co-related Queries aggregation operators, NULL values, complex integrity constraints in SQL.

Concept of Joins: Join, Outer Join, Left Outer Join, Right Outer Join, Self-Join

Schema Refinement: Problems caused by redundancy, decompositions, problems related to decomposition, reasoning about functional dependencies, FIRST, SECOND, THIRD normal forms, BCNF, lossless join decomposition, multi-valued dependencies, FOURTH normal form, FIFTH normal form.

UNIT - IV

Introduction to Transactions: Transaction Concept, Transaction State, Implementation of Atomicity and Durability, Concurrent Executions, Serializability, Recoverability, Implementation of Isolation, testing for serializability, Lock Based Protocols, Timestamp Based Protocols, Validation- Based Protocols, Multiple Granularity, Recovery and Atomicity, Log–Based Recovery, Recovery with Concurrent Transactions. TCL Commands – Save point Commit and Rollback.

UNIT – V

Overview of Triggers, Stored Procedures: triggers-Row level table level and active databases, Stored Procedures IN, OUT parameters, Execution of Stored Procedure from Java.

DBA – Introduction to DBA, Creating Users, Grant/Revoke Permissions on tables using DCL Commands.

TEXT BOOKS:

1. Database Management Systems – Raghurama Krishnan, Johannes Gehrke, Tata McGraw Hill, 3th Edition, 2008.
2. Database System Concepts - Silber Schatz, Korth, McGraw Hill, Vedition, 2005.

REFERENCE BOOKS:

1. Database Systems design, Implementation and Management- Peter Rob & Carlos Coronel, 7th Edition, Cengage Learning, 2006.
2. Fundamentals of Database Systems - Elmasri Navrate, Pearson Education, 2016.
3. Introduction to Database Systems - C.J .Date, Pearson Education, 2004.

Noted and Approved

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY
(AN AUTONOMOUS INSTITUTE)
Accredited by NBA & NAAC, Approved by AICTE, Affiliated to JNTU, Hyderabad

B.Tech. in CSE (DATA SCIENCE)
II Year II Semester Course Syllabus (KR21)

OPERATING SYSTEMS (21CS403PC)

L	T	P	C
3	0	0	3

Prerequisites/Corequisites:

1. PP102ES - Programming for Problem Solving Course
2. 21CS303PC - Data Structures Through C++ Course

Course Objectives: The course will help to

1. Understand the design and the services provided by an operating system.
2. Learn different process scheduling algorithms and deadlocks.
3. Facilitate students in understanding Inter process communication.
4. Categorize the operating systems' resource management techniques, file and memory management techniques.
5. Impart fundamentals of Protection and Sockets.

Course Outcomes: After learning the concepts of this course, the student is able to

1. Understand the basic concepts of Operating Systems.
2. Illustrate the different process scheduling algorithms and deadlocks.
3. Demonstrate Inter process communication.
4. Explore memory management techniques and file management concepts.
5. Analyze Sockets and its communication.

UNIT-I

Introduction to Operating System –Operating system objectives, User view, System view, Operating system definition, Computer System Organization, Computer System Architecture, OS Operations, Operating System services, OS Structures- Simple, Layered Architecture, Micro Kernel, Modular, Hybrid structure.

Introduction to Linux and Linux Utilities: Architecture of LINUX, features of LINUX, Introduction to vi editor.

System components and Linux commands: System components, Introduction to various LINUX commands such as PATH, man, echo, printf, script, passwd, uname, who, date, pwd, cd, ls, cd, touch, mv, rm, man, mkdir, rmdir, tar, gzip, cat.

UNIT-II

Process and CPU Scheduling - Process concepts and scheduling, Operations on processes, Threads, Scheduling Criteria, Scheduling Algorithms.

System call interface for process management-Introduction to various system calls such as fork, vfork, exit, wait, wait pid, exec.

Deadlocks-System Model, Deadlocks Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection and Recovery from Deadlock.

UNIT-III

Process Management and Synchronization -The Critical Section Problem, Synchronization Hardware, Semaphores and Classical Problems of Synchronization, Critical Regions, Monitors.

Inter Process Communication Mechanisms: IPC between processes on different systems, using pipes: FIFOs, Semaphores: semget, semop, semctl, message queues: msgget, msgsnd, msgrcv, msgctl, shared memory: shmget, shmat, shmdt, shmctl, ipc status commands.

UNIT-IV

Memory Management and Virtual Memory - Logical versus Physical Address Space, Swapping, Contiguous Allocation, Paging, Segmentation, Demand Paging, Page Replacement, Page Replacement Algorithms.

File System Interface and Operations – Access methods, Directory Structure, Protection, File System Structure, Allocation methods, Free-space Management.

Case Study: Introduction to UNIX file system, inode (Index Node), file descriptors, system calls and device drivers. File Management: File Structures, System Calls for File Management – create, open, close, read, write, lseek, link, symlink, unlink, stat, fstat, lstat, chmod, chown, Directory API – opendir, readdir, closedir, mkdir, rmdir,umask.

UNIT-V

Protection – System Protection, Goals of Protection, Principles of Protection, Domain of Protection.

Introduction to Sockets- Socket, socket connections - socket attributes, socket addresses, socket, connect, bind, listen, accept, socket communications.

TEXT BOOKS:

1. Operating System Principles - Abraham Silberchatz, Peter B. Galvin, Greg Gagne, John Wiley, 9th Edition, 2005.
2. Advanced Programming in the UNIX Environment - W. Richard. Stevens, 3rd Edition, Pearson Education, New Delhi, India. 2005.

REFERENCE BOOKS:

1. Internals and Design Principles - William Stallings, Operating Systems, 5th Edition, Pearson Education/PHI, 2005.
2. UNIX and shell Programming - Behrouz A. Forouzan, Richard F.Gilberg Thomson.
3. UNIX Programming Environment - Kernighan and Pike, PHI/ Pearson Education.2015.

Noted and Approved

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY

(AN AUTONOMOUS INSTITUTE)
Accredited by NBA & NAAC, Approved by AICTE, Affiliated to JNTU, Hyderabad

B.Tech. in CSE (DATA SCIENCE)
II Year II Semester Course Syllabus (KR21)

AUTOMATA THEORY AND COMPILER DESIGN (21CS404PC)

L	T	P	C
3	0	0	3

Prerequisites/ Corequisites:

1. 21CS301PC - Mathematical Foundations of Computer Science Course
2. 21CS303PC - Data Structures Through C++ Course

Course Objectives: The course will help to

1. Provide information to some of the central ideas of theoretical computer science from perspective of formal languages.
2. Introduce fundamental concepts of Formal languages, grammars, and types of automata.
3. Introduce the major concepts of language translation and compiler design.
4. Impart the knowledge of practical skills for constructing a compiler.
5. Introduce various phases of compiler with example.

Course Outcomes: After learning the concepts of this course, the student is able to

1. Identify and understand the concept of abstract machines and their power to recognize the languages.
2. Explore context free grammar for formal languages.
3. Design a compiler if given a set of language features.
4. Implement LL and LR parsers.
5. Design and develop machine code for three address code.

UNIT-I

Introduction to finite automata: The central concepts of automata theory, Structural representation of FA, Types of FA, Conversion of NFA to DFA, NFA with epsilon to NFA without epsilon conversion.
Regular Expression: Introduction to Regular language, Algebraic laws for regular expressions, Conversion of FA to RE, Conversion of RE to FA, Pumping lemma for Regular language.

UNIT-II

Grammar: Definition of Grammar, Types of grammars, Derivation, Types of derivations, Derivation tree, Ambiguity, Left recursion elimination.
Push Down Automata: Definition of PDA, Structural representation of PDA, Construction of PDA, conversion of PDA – CFG, conversion of CFG to PDA

UNIT-III

Turing Machine: Definition of TM, Structural representation of TM, Construction of TM.

Compiler: Definition of Compiler, Phases of Compiler, Lexical Analysis, Input Buffering.

UNIT-IV

Syntax Analysis: Types of Parsing, Top-down Parsing: Recursive Decent parsing, Predictive parsing, Bottom-up Parsing : SLR, CLR, LALR.

Semantic Analysis: Introduction to Syntax Directed Definition, Syntax Directed Translation, Attributes, Types of Attributes, Bottom-up evaluation of attributes.

Intermediate code generation: Types of Intermediate codes, Types of Three address codes, Evaluation of three address code.

UNIT-V

Runtime Environment: Storages organization, Storage allocation strategies: Static, Stack, Heap allocations, Activation Record.

Code Optimization: Introduction, Principal sources of optimization, flowgraph, techniques in local loop and global optimizations.

Code generation: Issues in code generation, DAG, Simple code generator.

TEXT BOOK:

1. Compilers: Principles, Techniques and Tools – Alfred V.Aho, Monica S.Lam, Ravi Sethi, Jeffry D. Ullman, 2nd Edition, Pearson.
2. Compiler Construction – Principles and Practice, Kenneth CLouden, Cengage Learning.

REFERENCE BOOKS:

1. Modern compiler implementation in C - Andrew W Appel, Revised Edition, Cambridge University Press.
2. The Theory and Practice of Compiler writing - J. P.Tremblay and P.G. Sorenson, TMH.
3. Writing compilers and interpreters - R.Mak, 3rdEdition,Wiley student edition.

Noted and Approved

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY

(AN AUTONOMOUS INSTITUTE)
Accredited by NBA & NAAC, Approved by AICTE, Affiliated to JNTU, Hyderabad

B.Tech. in CSE (DATA SCIENCE)
II Year II Semester Course Syllabus (KR21)

JAVA PROGRAMMING LAB (21CS405PC)

L	T	P	C
0	0	4	2

Prerequisites/Corequisites:

1. PP102ES - Programming for Problem Solving Course.
2. 21CS401PC - Java Programming Course.

Course Objectives: The course will help to

1. Write programs using abstract classes.
2. Write programs for solving real world problems using java collection frame work.
3. Write multi threaded programs.
4. Introduce java compiler and eclipse platform.
5. Impart hands on experience with java programming.

Course Outcomes: After learning the concepts of this course the student is able to

1. Outline the concepts of OOPs in structuring computer applications for solving problems.
2. Choose appropriate collections to solve programming problems.
3. Design programs using abstract classes.
4. Utilize the concepts of I/O streams and exception handling in a given real time problem.
5. Develop java applications by using JDBC.

List of Exercises:

1.

a. Given a number, check if the number (N) can be written as the form (k+1)*k. Write a java program to print those numbers in the given range.

b. Write a java program to check whether the given number is gapful or not.

A number is gapful if it is at least 3 digits long and is divisible by the number formed by stringing the first and last numbers together. The smallest number that fits this description is 100. First digit is 1, last digit is 0, forming 10, which is a factor of 100. Therefore, 100 is gapful.

c. Cricketer's Pension Continuing our journey in mastering the conditional statements & our interest with cricket. Let us help the Indian cricket's governing body (BCCI) to automate its plan of allotting pensions to former players. The rules are given below:
If a player has played more than 10 test matches and 100 ODI's he receives Rs.50,000. If a player has played more than 10 test matches, he receives Rs.25,000.

If a player has played more than 100 ODI's he receives Rs.15,000. If a player has played for India he receives Rs.10000. The amount is incremented by 1/4th for every 'man of the match' award.

If a player has not played for India but played IPL he receives an amount of Rs.8000. If a player has not played for India nor IPL he receives an amount of Rs.7000.

2. A resistor is a circuit device designed to provide a specific resistance between its two ends. Resistance is expressed in ohms (Ω) or kilo-ohms (kΩ). Resistors are usually marked with colored bands that encode their resistance, as shown in figure-1 below. The first two bands represent digits and the third is a power-of-ten multiplier.

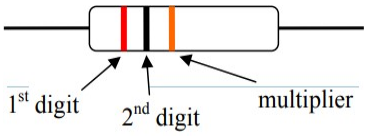


Figure 1: Colour Bands of a resistor

The table below shows the number value of each band color. For example, if the first digit is red (represents 2), the second is digit is black (represents 0), and the third digit is orange represented as multiplier (represents 3), the resistance is 20×10^3 Ω or 20 kΩ.

Color	Number Value
Black	0
Brown	1
Red	2
Orange	3
Yellow	4
Green	5
Blue	6
Violet	7
Grey	8
White	9

Write a Resistor class containing the parameterized constructor, which takes in three strings representing the three band colors. Write the methods to calculate and set the resistance for the resistor.

3.

a. Given an array **arr[]** of **N** integers, the task is to find the maximum difference between any two elements of the array.

b. Write a Java program to fill the below pattern into a square matrix:

The matrix has to be filled with numbers starting from 1. It has to start fill first row last column, last row (reverse), first column (reverse) and so on. Please refer the following example input 5

output =

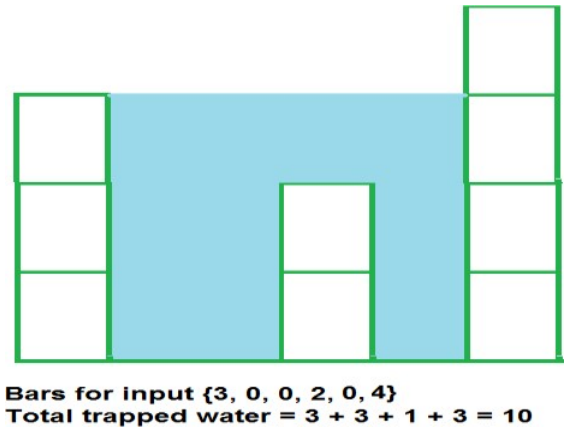
1	2	3	4	5
16	17	18	19	6
15	24	25	20	7
14	23	22	21	8
13	12	11	10	9

c. Given n non-negative integers representing an elevation map where the width of each bar is 1, compute how much water it is able to trap after raining.

Examples:

Input: arr[] = {3, 0, 0, 2, 0, 4}

Output: 10



We can trap "3*2 units" of water between 3 an 2, "1 unit" on top of bar 2 and "3 units" between 2 and 4.

4.

a) Given two strings text1 and text2, return the length of their longest common subsequence. A subsequence of a string is a new string generated from the original string with some characters (can be none) deleted without changing the relative order of the remaining characters. (eg, "ace" is a subsequence of "abcde" while "aec" is not). A common subsequence of two strings is a subsequence that is common to both strings. If there is no common subsequence, return0.

b) Given two strings s1 and s2, your task is to merge those strings to form a new merged string. A merge operation on two strings is described as follows:

Append alternating characters from s1 and s2, respectively, to merge String. Once all of the characters in one of the strings have been merged, append the remaining characters in the other string to merged String.

c) You are given a string S of alphabet characters and the task is to find its matching decimal representation as on the shown keypad. Output the decimal representation corresponding to the string.

For ex: if you are given—Amazon then its corresponding decimal representation will be 262966.

5. Define a MyRectangle class with four public data members representing the x- and y-coordinates of the bottom-left vertex and top-right vertex of a rectangle, whose sides are parallel to the x- or y-axis.

For example, the statement new MyRectangle (20, 80, 30, 90) creates a rectangle with bottom-left vertex at position (20, 80), and top-right vertex at (30, 90).

a. Write an area() method, which computes the area of a rectangle.

b. Write the overlap (My Rectanglerect) method. This method returns a rectangle which is the overlapped region of two rectangles. In the event that there is no overlap, it should return a rectangle with both bottom-left vertex and top-right vertex at position (0, 0).

c. Using the overlap(MyRectanglerect) method written above, write the overlapAll(MyRectangle [] rectangles) method which returns the overlapped region of all the rectangles in the array. You may assume that there is at least one element in the array. Your method should be efficient in that the moment it finds that the overlapped region is empty, it should return a rectangle with both vertices at (0,0)immediately.

d. Write MySqaure.java, MySquareextendsMyRectangle.A square is defined by its bottom-left vertex and the length of its side. Complete the super (. . .) statement in the constructor.

e. Below is output of MySquare.java program when the user enters: 10 305.

f. Override toString () method in MyRectangle in order to get such output.

6.

a. Write a java program that loads names and phone numbers from the text file into Hash Table where data is organized as one line per record and each field in record are separated by a tab(\t). It takes a name or phone number as input and prints the corresponding other value from hash table.

b. You have created a web-based survey of favorite programming languages and are capturing the results into a text file named — log file|. The structure of the text file is:

Total # Entries
Vote for Entry 1
IP Address for Entry 1
Timestamp in seconds for Entry 1
Vote for Entry 2
IP Address for Entry 2
Timestamp in seconds for Entry 2
….

For example, here is a sample log file of six entries:

6
PHP
137.229.156.12
1000002
C#
137.229.156.18
1000005
PHP
137.229.156.12
1000006
Prolog
156.213.38.31
1000010
PHP
128.120.56.214
1000020
PHP
137.229.156.12
1000022

The log file is ordered by increasing timestamp. You are concerned that some people are voting multiple times for the same item. To somewhat address this problem, throw out any new votes for the same item that come from the same IP address within 20seconds.

In the above example, the second and last votes for PHP would be thrown out because they are for the same item from the same IP address and occur within 20 seconds of other PHP votes from the same IP address. However, the PHP vote from 128.120.56.214 would be retained since there is not another PHP vote from this IP address.

Write a Java program to count the votes from the log file, throwing out duplicate votes using the rules above. Display the votes in a table, as shown below for the example:

PHP 2

C# 1

Prolog 1

7.

a. Write a Java Class to implement a method Addition () that returns a new Array where each array element at the index k corresponds to the sum of elements of the array (src) starting at index 0 and including element at the index _k ‘. For example, for array [2, 3, 5], the method will return array [2, 5, 10]. For an array of size ‘0‘ora null parameter, the method will throw exception Illegal Argument Exception With the message ―Invalid Argument.
b. Write a Java Code to implement a multithreaded version of FizzBuzz with four threads.

If the number is divisible by 3, output "fizz".
If the number is divisible by 5, output "buzz".
If the number is divisible by both 3 and 5, output “fizzbuzz".
If the number is not divisible by both 3 and 5 print the number.

For instance if n is 15, we will have the output as — 1, 2, fizz, 4, buzz, fizz, 7, 8, fizz, buzz, 11, fizz, 13, 14, fizzbuzz

1. Thread A will call fizz () to check for divisibility of 3 and outputs fizz.
2. Thread B will call buzz () to check for divisibility of 5 and outputs buzz.
3. Thread C will call fizzbuzz () to check for divisibility of 3 and 5 and outputs fizzbuzz.
4. Thread D will call number () which should only output the numbers.

8.

a. Write a java program to store the employee details in an ArrayList and display the employee details in ascending order of their experience. Create 'Employee' class with two instance variables Employee name and Employee experience (no. ofyears).

b. Write a program to find the most common words in the list of words given in sorted order based on occurrence from largest to smallest. If any of words are having same occurrence then consider the smallest character order comesfirst.

Input format: First line contains the list of words and next line contains a number (k) which represent the top most words to display.Output format: display the k top most words.

9.

a. Write a java Program to write a method fCount which takes a string as a parameter. The Method fCount should return the Map which has the frequency count of the given word. For example if the string passed is—"hello", the map should return {h- 1, e-1, l-2, o-1}. The order of the characters should be same as in the string.

b. When working with HashMaps, sometimes cases arise where we wish to determine if two HashMaps have any key-value pairs in common. For example, we might have the following two Hashmaps (named hashmap1 and hashmap2, respectively) that map from String to String (i.e., their type is HashMap) and we want to count how many key-value pairs they have in common.

Hashmap1

Key	Value
Alice	Healthy
Mary	Ecstatic
Bob	Happy
Chuck	Fine
Felix	Sick

Hashmap2

Key	Value
Mary	Ecstatic
Felix	Healthy
Ricardo	Superb
Tam	Fine
Bob	Happy

In the example above, these two Hash-Maps have two key-value pairs in common, namely: "Mary"-"Ecstatic" and "Bob"-"Happy". Note that although the key "Felix" is in both HashMaps, the associated value with this key is different in the two maps (hence this does not count as a key-value pair that is common to both HashMaps). Similarly, just having the same value without the same key (such as the value "Fine" which is mapped to by different keys in the two different HashMaps) would also not count as a common key-value pair between the two HashMaps.

Your job is to write a method:

```
Public int common key value price (HashMap<string,string> map1,
                                   HashMap<string,string> map2 )
```

That is passed two objects of type HashMap<string, string> and returns the number of common key/value pairs between the two HashMaps.

10

a. Demonstrate a JDBC Program on Employee Schema given below

Read a department number from the user and display those employee names in ascending order who are working in the department.

Note - Display records based on employee names in ascending order The DB Credentials Name of the DB - test;
Name of the table- emp;

JDBC_DRIVER = "com.mysql.jdbc.Driver"; DB_URL = "jdbc:mysql://localhost/test"; Username-student Password-student

emp schema;

Field	Type	Null	Key	Default	Extra
empno	int (4)	NO	PRI	NULL	
ename	varchar (50)	NO		NULL	
job	varchar (50)	NO		NULL	
mgr	int (4)	YES		NULL	
hiredate	date	YES		NULL	
sal	decimal (10,2)	YES		NULL	
comm	decimal (10,2)	YES		NULL	
deptno	int (2)	YES	MUL	NULL	

b. Demonstrate a JDBC program to display all the employee names in ascending order who are working in "Dallas" location using createStatement (). Schema Given below emp schema;

Field	Type	Null	Key	Default	Extra
empno	int (4)	NO	PRI	NULL	
ename	varchar (50)	NO		NULL	
job	varchar (50)	NO		NULL	
mgr	int (4)	YES		NULL	
hiredate	date	YES		NULL	
sal	decimal (10,2)	YES		NULL	
comm	decimal (10,2)	YES		NULL	
deptno	int (2)	YES	MUL	NULL	

c. Demonstrate JDBC program to read three values (dno, dname, dloc) from the user and insert those records into the dept table using Prepared Statement

The Reading of input should be first deptno followed by dept name followed by dept location Dept Schema

Field	Type	Null	Key	Default	Extra
deptno	int (2)	NO	PRI	NULL	
dname	varchar (50)	NO		NULL	
location	varchar (50)	NO		NULL	

TEXT BOOKS:

1. Java The complete reference-Herbert Schildt - 9th Edition, McGraw Hill Education (India) Pvt. Ltd,2014.
2. Java Database Best Practices- George Reese, O'Reilly Media, 2003.

REFERENCE BOOKS:

1. An Introduction to programming and OO design using Java-J.Ninoand F.A. Hosch, John Wiley & sons, 2008.
2. Introduction to Java programming - Y. Daniel Liang, Pearson Education, 2012.

Noted and Approved

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY

(AN AUTONOMOUS INSTITUTE)
Accredited by NBA & NAAC, Approved by AICTE, Affiliated to JNTU, Hyderabad

B.Tech. in CSE (DATA SCIENCE)
II Year II Semester Course Syllabus (KR21)

DATABASE MANAGEMENT SYSTEMS LAB (21CS406PC)

L	T	P	C
0	0	3	1.5

Prerequisites/Corequisites:

1. PP102ES - Programming for Problem Solving Course
2. 21CS303PC - Data Structures Through C++ Course
3. 21CS401PC - Java Programming Course
4. 21CS402PC - Database Management Systems Course

Course Objectives: The course will help to

1. Introduce ER data model, database design and normalization.
2. Learn SQL basics for data definition and data manipulation.
3. Design data base schema for a given application and apply normalization.
4. Acquire skills in using SQL commands for data definition and data manipulation.
5. Develop solutions for database applications using procedures, cursors and triggers.

Course Outcomes: After learning the concepts of this course, the student is able to

1. Design database schema for a given application and apply normalization.
2. Summarize SQL commands for data definition and data manipulation.
3. Develop solutions for database applications using procedures, cursors and triggers.
4. Construct queries using SQL and Demonstrate creation and usage of Triggers, Views and Stored Procedures using SQL.
5. Apply and relate various advanced SQL queries related to Transaction Processing & Locking using the concept of Concurrency control.

List of Exercises

Scenario 1:

Product-Sales database: **South Wind**

South wind database is a sample database used by Organization. The database contains the sales data for South Wind Traders, it is foods export-import company. Using this schema to demonstrate how customers can choose and order products, how orders are placed and how those products get delivered to the customer.

Products: This Entity will have all the products details where suppliers will supply products based on customers demand.

Supplies: This Entity will supply the products demanded by the customers. Shippers: This Entity will take the orders from suppliers and deliver to customers. Employees: Employees will monitor the orders placed by customers.
Invoices: This Entity will take care of the billing process based on customer order. Etc..identify some more entities and find out the relationship between them.

A product - sales the above process involves many steps like

1. Analyzing the problem and identifying the Entities and Relationships,
2. E-R Model
3. Relational Model
4. Normalization
5. Creating the database
6. Querying.

Exercise 1: E-R Model
Analyze and come up with the entities in it. Identify what data has to be persisted in the database. This contains the entities, attributes etc. Identify the primary keys for all the entities. Identify the other keys like Foreign Key and constraints like NULL, NOT NULL, CHECK etc.
Example to create for **products**, **customers**, **suppliers**, **orders**, employees**, order details, categories**, among others.
Students should submit E-R diagrams using the above tables.

Exercise 2: Installation & DDL

Installation of Mysql and practicing DDL commands.

Creating databases, how to create tables, altering the database or tables, dropping tables and databases if not required. You will also try truncate, rename commands etc.
Data Definition Language (DDL): create, alter, drop.

Exercise 3: DML
Data Manipulation Language Commands (DML) commands are used to for managing data within schema objects. Exercising the commands using **DML:** insert, delete, update on the following tables : products, customers, suppliers, orders, , employees, order details, categories.

- INSERT – insert data into a table.
- UPDATE – updates existing data within a table.
- DELETE – deletes single or all records from a table. Data Query Language –Select.

Populate all the tables designed in experiment: 2 with appropriate data.

Exercise 4: Querying

Practice queries on **Aggregate functions** like count, max, min, avg, sum Practice queries like nested queries/co-related queries using ANY, ALL, IN, Exists, NOT EXISTS, UNION, INTERSECT, group-by, having etc. **Joins:** Join, Left Outer Join, Right Outer Join, Self-Join

Exercise 5: Querying (continued…)
Some examples to practice the queries:

1. Display all the order details of given a customer.3
2. Display all the products.
3. Get the highest sold product from given supplier ID

4. List all products grouped by category
5. List the products, whose products unit price is greater than all the products of average.
6. List Details of order and customer of each order
7. List the products which were sold in year 1997
8. Display the total amount for each order
9. Display Order Details for given an order ID

Order Details: product name and unit price for given order ID
Exercising Simple to complex Queries using joins, nested and co-related queries.

Exercise 6: Stored Procedures:

1. Create a stored procedure, Alter and Drop a procedure, IN, OUT, IN & OUT parameters.
2. Create a Procedure to display order details of given customer ID like ordered, order Date, Required Date, Shipped Date.
3. Create a procedure to accept a customer ID and display the customer order history (product name and how much quantity ordered for that particular product).Ex: product name, Total quantity he/she ordered.
4. Create a procedure to display Ten Most Expensive Products Columns should be displayed Product name & Unit price.

Exercise 7: Views

1. Create a view to display the current product list which is available (not discontinued).
2. Create a view to display the products by category.
3. Display product name, quantity Per Unit, units In Stock, Discontinued.
4. Create a view ―Invoices to display all the information from order, customer, shipper for each Order Details.

Exercise 8: Triggers

Demonstrate Create Trigger, Alter Trigger, Drop Trigger, Row Level, Table Level triggers, Before Insert, After Insert, Before Update, After Update, Before Delete, After Delete.

Exercise 9:

Demonstrate the role of DBA using DCL commands.

TEXT BOOKS:

1. Database Management Systems- Raghurama K rishnan, Johannes Gehrke, Tata Mc GrawHill, 3rd Edition, 2008.
2. Database System Concepts-Silber schatz, Korth, McGraw Hill, 5th Edition, 2005.

REFERENCE BOOKS:

1. Database Systems design, Implementation and Management – Peter Rob& Carlos Coronel, 7th Edition, 2006.
2. Fundamentals of Database Systems – Elmasri Navrate, Pearson Education, 2016.
3. Introduction to Database Systems - J.Date, Pearson Education, 2004.

Noted and Approved

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY
(AN AUTONOMOUS INSTITUTE)
Accredited by NBA & NAAC, Approved by AICTE, Affiliated to JNTU, Hyderabad

B.Tech. in CSE (DATA SCIENCE)
II Year II Semester Course Syllabus (KR21)

OPERATING SYSTEMS LAB (21CS407PC)

L	T	P	C
0	0	3	1.5

Prerequisites/Corequisites:

1. PP102ES - Programming for Problem Solving Course
2. 21CS303PC - Data Structures Through C++ Course
3. 21CS403PC - Operating Systems Course

Course Objectives: The course will help to

1. Provide an understanding of the design aspects of operating system concepts through simulation.
2. Introduce basic UNIX commands, system call interface for process management; inter process communication and I/O in UNIX.
3. Demonstrate the knowledge of the components of computer and the irrespective roles in computing.
4. Recognize and resolve user problems with standard operating environments.
5. Gain practical knowledge of how programming languages, operating systems, and architectures interact and how to use each effectively.

Course Outcomes: After learning the concepts of this course the student is able to

1. Illustrate the use of Linux OS, by means of a command line shell.
2. Implement operating system concepts such as scheduling, deadlock management, file management and memory management.
3. Implement C programs using UNIX system calls.
4. Demonstrate synchronization and various components of a typical operating system.
5. Analyze various system calls for managing processes, memory and the file system.

List of Exercises

Exercise 1:

Identify the peripherals of a computer, components in a CPU and its functions. Draw the block diagram ofthe CPU along with the configuration of each peripheral.

a. Disassemble and assemble the PC back to working condition.
b. Install MS windows on the personal computer.
c. Install Linux on the computer. This computer should have windows installed. The system should be configured as dual boot with both windows and Linux.

Exercise 2:

a. Implement in c language the following Unix commands using system calls i) cat ii) lsiii) mv

b. Write a C program that takes one or more file/directory names as command line input and reports following information

i) File Type ii) Number of Links iii) Time of last Access iv) Read, write and execute permissions.

Exercise 3:

Assume you have the following jobs to execute with one processor, with the jobs arriving in the order listed here:

i	T(pi)
0	80
1	20
2	10
3	20
4	50

With the following values write a program to get the required output which is listed below

a. Suppose a system uses FCFS scheduling. Create a Gantt chart illustrating the execution of these processes?
b. What is the turnaround time for process p3?
c. What is the average wait time for the processes?

Exercise 4:

a. Write a C program that illustrate communication between two unrelated process using named pipes
b. Write a C program that receives a message from message queue and display them
c. Write a C program to allow cooperating process to lock a resource for exclusive use (using semaphore)
d. Write a C program that illustrate the suspending and resuming process using signal
e. Write a C program that implements producer-Consumer system with two process using semaphore

Exercise 5:

Consider the following snapshot of a system. P0, P1, P2, P3, P4 are the processes and A, B, C, D are the resource types. The values in the table indicates the number of instances of a specific resource (for example: 3 3 2 1 under the last column indicates that there are 3 A-type, 3 B-type, 2 C-type and 1 D-type resources are available after allocating the resources to all five processes). The numbers under allocation-column indicate that those numbers of resources are allocated to various processes mentioned in the first column. The numbers under Max-column indicate the maximum number of resources required by the processes. For example: in 1st row under allocation-column 2 0 0 1 indicate there are 2 A-type, 0 B-type, 0 C-type and 1 D- type resources are allocated to process P0. Whereas 4 2 1 2 under Max-column indicate that process P0‘s maximum requirement is 4 A-type, 2 B-type, 1 C-type and 2 D-type resources.

Process	Allocation A B C D	Max A B C D	Available A B C D
P0	2 0 0 1	4 2 1 2	3 3 2 1
P1	3 1 2 1	5 2 5 2	
P2	2 1 0 3	2 3 1 6	
P3	1 3 12	1 4 2 4	
P4	1 4 3 2	3 6 6 5	

Answer the following questions using banker’s algorithm by providing all intermediate steps

a. How many instances of resources are present in the system under each type of are source?
b. Compute the Need matrix for the given snapshot of a system.
c. Verify whether the snapshot of the present system is in a safe state by demonstrating an order in which the processes may complete. If a request from process P1 arrives for (1,1,0,0), can the request be granted immediately?
d. If a request from process P4 arrives for (0,0,2,0), can the request be granted immediately?

Exercise 6:

a. Write a C program to list every file in directory, its inode number and filename
b. Write a C program to create child process and allow parent process to display "parent" and the child to display "child" on the screen

Exercise 7:

a. Write client server programs using c for interaction between server and client process using Unix Domain sockets.

TEXT BOOKS:

1. Operating System Principles - Abraham Silberchatz, Peter B. Galvin, Greg Gagne, 9th Edition, JohnWiley, 2005.
2. Advanced Programming in the UNIX Environment -W. Richard. Stevens, 3rdEdition, Pearson Education, New Delhi, India.2005

REFERENCE BOOKS:

1. Internals and Design Principles - William Stallings, Operating Systems, 5thEdition, Pearson Education/PHI, 2005.
2. UNIX and shell Programming - Behrouz A. Forouzan, Richard F.Gilberg, Thomson.
3. UNIX Programming Environment, Kernighan and Pike, PHI/ Pearson Education, 2015.

Noted and Approved

KESHAV MEMORIAL INSTITUTE OF TECHNOLOGY

(AN AUTONOMOUS INSTITUTE)
Accredited by NBA & NAAC, Approved by AICTE, Affiliated to JNTU, Hyderabad

B.Tech. in CSE (DATA SCIENCE)
II Year II Semester Course Syllabus (KR21)

CONSTITUTION OF INDIA (*21MC408HS)

L	T	P	C
3	0	0	0

Prerequisites/Corequisites: Nil

Course Objectives: The course will help to

1. Create awareness among students about the Indian Constitution.
2. Acquaint the working conditions of union, state, local levels, their powers and functions.
3. Create consciousness in the students on democratic values and principles articulated in the constitution.
4. Expose the students on the relations between federal and provincial units.
5. Divulge the students about the statutory institutions.

Course Outcomes: After learning the concepts of this course the student is able to

1. Outline important issues related to gender in contemporary India.
2. Illustrate the working conditions at union, state and local levels.
3. Apply democratic values and principles articulated in the constitution.
4. Categorize relations between federal and provincial units.
5. Compare and contrast various statutory institutions.

UNIT-I
Evolution of the Indian Constitution: 1909 Act, 1919 Act and 1935 Act. Constituent Assembly: Composition and Functions; Fundamental features of the Indian Constitution.

UNIT-II
Union Government: Executive-President, Prime Minister, Council of Minister
State Government: Executive: Governor, Chief Minister, Council of Minister
Local Government: Panchayat Raj Institutions, Urban Government.

UNIT-III
Rights and Duties: Fundamental Rights, Directive principles, Fundamental Duties

UNIT-IV
Relation between Federal and Provincial units: Union-State relations, Administrative, legislative and Financial, Inter State council, NITI Ayog, Finance Commission of India

UNIT-V
Statutory Institutions: Elections-Election Commission of India, National Human Rights Commission, National Commission for Women

TEXTBOOKS:

1. D.D. Basu, Introduction to the constitution of India, Lexis Nexis, New Delhi.
2. Subhash Kashyap, Our Parliament, National Book Trust, New Delhi.
3. Peu Ghosh, Indian Government &Politics, Prentice Hall of India, New Delhi.
4. B.Z. Fadia & Kuldeep Fadia, Indian Government &Politics, Lexis Nexis, New Delhi.

Noted and Approved