Autonomous Program Structure
Second Year B. Tech. Fourth Semester
Computer Engineering
Academic Year: 2021-2022 Onwards

Course Code	Course Title	Teaching Scheme Hours /Week			Examination Scheme				Marks	Credits
		Lecture	Tutorial	Practical	In Semester	End Semester	Oral	Practical		
20BSCE 401	Calculus and Statistics	3	1	0	50	50`	0	0	100	4
20CE401	Theory of Computation	3	1	0	50	50	0	0	100	4
20CE402	Database Management Systems	3	0	0	50	50	0	0	100	3
20CE403	Operating Systems	3	0	0	50	50	0	0	100	3
20CE404	Machine Learning	3	0	0	50	50	0	0	100	3
20CE402L	Database Management Systems Laboratory	0	0	4	25	0	0	25	50	2
20CE403L	Operating Systems Laboratory	0	0	2	25	0	25	0	50	1
20CE404L	Machine Learning Laboratory	0	0	4	25	0	25	0	50	2
20AC401	Audit Course	0	0	2	0	0	0	0	0	No Credits
	Total	**15**	**2**	**12**	**325**	**250**	**50**	**25**	**650**	**22**
	Grand Total	**29**			**650**					**22**

APPROVED BY
Secretary Governing Body
MKSSS's Cummins College of Engineering
For Women, Pune-411052

APPROVED BY
Chairman Governing Body
MKSSS's Cummins College of Engineering
For Women, Pune-411052

M.K.S.S.S. Cummins college of Engg. for Women KARVENAGAR Pune -52

20BSCE 401 Calculus & Statistics

Teaching Scheme
Lectures: 3 Hrs. /week
Tutorials: 1Hr/week

Examination Scheme
In semester: 50 marks
End semester: 50 marks
Credits: 04

Prerequisite:

1. First order linear ordinary differential equations.
2. Basics of Vector Algebra
3. Integration – basic properties, standard results, Beta & Gamma Functions.
4. Partial Fractions.
5. Permutation & Combination. Basics of probability.

Course Objectives: Mathematics is a necessary path to scientific knowledge which opens new perspective of mental activity. Our aim is to provide sound knowledge of engineering mathematics to make the students think mathematically and strengthen their thinking power to analyze and solve engineering problems in their respective areas.

Course Outcome:
After completion of the course, students will be able to

1. Find the solution of higher order linear differential equation.
2. Apply the concepts of vector calculus to find vector differentiation and vector integration.
3. Find the mathematical transform of a given function, use transform technique to solve integral equation, difference equation.
4. Apply concepts of statistics to interpret the data, calculate probabilities of random events.

Unit 1: Higher Order Linear Differential equation and application (08)

Higher order Linear differential Equation with constant coefficients, Cauchy's and Legendre's Differential Equations, Simultaneous Differential Equations, Modelling of electrical circuits.

Unit 2: Vector Calculus (07)

Physical interpretation of vector differentiation, vector differential operator, Gradient, Divergence, Curl, Directional derivative, Solenoidal, Irrotational and Conservative fields, Scalar potential, Vector identities, Line integral.

Unit 3: Fourier Transform (06)

Complex exponential form of Fourier series, Fourier integral theorem, sine and cosine integrals, Fourier transform, Fourier Sine and Cosine transform, Inverse Fourier Transform, Introduction to Discrete Fourier Transform.

Unit 4: Z – Transform (06)

Definition, standard properties, Z- Transform of standard sequences, Inverse Z – Transform using standard results, Inversion integral method, solution of difference equation to solve Computer Engineering Problems.

Unit 5: Probability **(07)**

Theorem of total probability, Conditional probability, Baye's theorem, Random variables- discrete and continuous, Mathematical expectations, Distributions – Binomial, Poisson, Normal.

Unit 6: Statistics **(08)**

Types of data: Quantitative & qualitative data, cross-sectional and time-series data, discrete and continuous data, Measures of central tendency, standard deviation, coefficient of variation, Moments, Skewness, Kurtosis, Introduction to Bi-variate data, Correlation, Linear Regression.

Text Books:

1. B. S. Grewal, **'Higher Engineering Mathematics'**, *Khanna Publications.*
2. B. V. Ramana, **'Higher Engineering Mathematics'**, *Tata McGraw Hill Publications* (2007)
3. C.R.Wylie, L.C. Barrette, '**Advanced Engineering Mathematics**', *McGraw Hill Publications, New Delhi.(*6th edition)(2003)

References:

1. Peter V. O'Neil, **'Advanced Engineering Mathematics'**, *Thomson Brooks / Cole, Singapore (*5th edition) (2007).
2. Erwin Kreyszig ,**'Advanced Engineering Mathematics'** *Wiley Eastern Ltd.* (8th Student Edition), (2004).
3. S.C. Gupta, V. K. Kapoor, **'Fundamental of Mathematical Statistics**', *S. Chand & Sons*, (10th edition), (2002).

20CE401 THEORY OF COMPUTATION

Teaching Scheme
Lectures: 03 Hrs/Week
Tutorial : 01 Hrs/Week

Examination Scheme
In Semester: 50 Marks
End Semester: 50 Marks
Credit: 4

Prerequisites:

1. Data Structures and Algorithms (20CE302)
2. Discrete Mathematics (20CE303)

Course Objectives:
To facilitate the learners -

1. Recall and understand the basics of mathematical concepts, formal languages and machines.
2. Understand and design different computational models like finite automata, regular expression, push down automata, context free grammar, turing machine for a given language.
3. Apply inter conversion between equivalent representations of a language.
4. Design appropriate computational models

Course Outcomes:
By taking this course, the learner will be able to -

1. Apply the knowledge of basics of mathematics and logic for problem understanding, representation and solving.
2. Construct different computational models like Finite automata, Regular Expression and Context Free Grammar.
3. Evaluate capabilities of computational models by inter-conversions.
4. Design appropriate computational models to solve given problems using PushDown Automata and Turing Machine.

Unit 1: Introduction (06)

Finite and infinite set. Basic concepts of symbol, alphabet, Kleene Closure and positive Closure of Alphabet, Strings, Empty String, Substring of a string, Concatenation of strings, Formal Language Definition, Finite representation of languages. Concept of Basic Machine and Finite State Machine. Finite Automata (FA): (Deterministic FA, Non-deterministic FA, Є-NFA): Definition.

Unit 2: Finite Automata (07)

Construction of FA (DFA, NFA, Є-NFA) - Transition Function and language acceptance, Transition graph. Conversion of NFA with Є moves to NFA without Є moves, Conversion of NFA without Є moves to DFA, Direct Conversion of NFA with Є to DFA.

Unit 3: Regular Expression (07)

Regular Expression (RE): definition and operators, Primitive Regular Expressions, Algebraic Laws of Regular Expressions, Languages Defined by Regular Expressions, Building Regular Expressions, Closure Properties of Regular Languages, Regular expression examples. Inter-conversion of RE and FA, Construction of FA equivalent to RE (RE to Є-NFA, Є-NFA to DFA). Construction of RE equivalent to FA using Arden's Theorem. Pumping Lemma for Regular languages, Limitations of FA.

Unit 4: Context Free Grammar and Languages (07)

Grammar: Definition, representation of grammar. Context Free Grammar (CFG) - Definition, Derivation – Leftmost, Rightmost, sentential form, parse tree, ambiguous grammar and removing ambiguity from grammar, Simplification of CFG, Normal Forms - Chomsky normal form, Greibach normal form, Closure properties of Context Free Languages (CFL), Decision properties of CFL, Chomsky hierarchy. Regular grammar- Definition, left linear, right linear grammar, Applications of grammar.

Unit 5: Push Down Automata (07)

Push Down Automata (PDA): Definition, Notations, Transition Table form, Types of PDA (Deterministic PDA and Non Deterministic PDA), acceptance by final state, acceptance by empty stack, Construction of PDA (DPDA, NPDA), Instantaneous Description of PDA. Equivalence of PDA and CFG - Grammar to PDA conversion, Applications of PDA.

Unit 6: Turing Machine (08)

Turing machine (TMs): Formal Definition, TM Instantaneous Description, Transition Function, Languages of TM, Turing Machine and halting, Deterministic Turing Machines (DTM), Construction of DTM. Universal Turing Machine (UTM), Church-Turing hypothesis, Comparison between FA, PDA and TM. Turing Machine Halting Problem. TM's as acceptors, Recognizing Languages with TM's.

Text Books:

1. Hopcroft J., Motwani R., Ullman J., "Introduction to Automata Theory, Languages and Computations", Third edition, 2008, Pearson Education Asia. ISBN: 9788131720479

2. Michael Sipser, "Introduction to The Theory of Computation", Third edition, 2017 Thomson Course Technology, ISBN: 9781131525296

Reference Books:

1. Daniel Cohen., "Introduction to Computer Theory", Second edition, 2011, Wiley Publications (India) ISBN: 9788126513345

2. H.R. Lewis, C. H. Papadimitrou, "Elements of the Theory of Computation", Second edition, 2006, Prentice Hall Inc. ISBN: 8131703878

3. John C Martin. "Introduction to Language and Theory of Computation", Third edition, 2012, Tata McGraw- Hill, ISBN: 978007660489

4. Vivek Kulkarni, "Theory of Computation", Oxford university edition, 2013, ISBN 13:9780198084587

5. K. L. P Mishra, N. Chandrashekaran (2003), Theory of Computer Science-Automata Languages and Computation, 2nd edition, Prentice Hall of India, India.

Suggestive List of Tutorials:

1. Design of Finite state machine
2. Design Deterministic Finite Automata
3. NFA design and NFA to DFA conversion
4. Design of Regular Expression from Language
5. Converting RE to NFA with null moves and then NFA with null moves to NFA without null moves
6. Formal language and CFG interconversion
7. Simplification / standardization of CFG to Normal Forms
8. Design of Push down Automata
9. Design of Turing Machine

4. Vivek Kulkarni, "Theory of Computation", Oxford university edition, 2013, ISBN 13:9780198084587

5. K. L. P Mishra, N. Chandrashekaran (2003), Theory of Computer Science-Automata Languages and Computation, 2nd edition, Prentice Hall of India, India.

Online/Web/Other References:

1. 1. Automata Theory. Estimated 7 weeks course of 5 -10 hours/week offered by edx 2020 https://www.edx.org/course/automata-theory

2. www.nptel.ac.in . 12 weeks course offered by IIT B. nptel.ac.in/courses/106104028/theory of computation.

20CE 402 Database Management Systems

Teaching Scheme
Lecture: 3 Hrs/week

Examination Scheme
In semester: 50 marks
End semester: 50 marks
Credits: 3

Course Objectives
To facilitate the learners to-
1. Design database schema using an entity relationship diagram (ERD) and normalization.
2. Design queries using Structured Query Language (SQL) to retrieve the required data from the database.
3. Understand the storage systems and query processing and optimization concepts.
4. Understand Transaction management in a Database management System.
5. Understand NoSQL Databases to handle unstructured data.

Course Outcomes:
With successful completion of the course, the students will be able to–
1. Design the Entity Relationship diagram for the system / application considering its constraints and design issues.
2. Apply the knowledge of SQL to retrieve the required data from the database.
3. Understand the storage systems and query processing and optimization concepts.
4. Make use of various Transaction management concepts for scheduling concurrent transactions.
5. Apply the knowledge of NoSQL databases to handle unstructured data.

Unit 1: Database Design (10)
Introduction to database management systems, Advantages of a Database Management Systems over file processing systems. Data abstraction, Data Independence, DBMS Architecture.
Database Design - Entity Relationship Diagram (ERD), Converting Entity Relationship Diagram into tables, Extended Entity Relationship (EER) Diagram features, rules for converting EER diagram to tables, Primary key, Foreign key and other Integrity constraints.
Codd's Twelve Rules for Relational DBMS, Normalization.

Unit 2: Relational query languages (8)
Relational algebra, Introduction to Structured Query Language (SQL)
SQL - Data Definition Language (DDL): SQL Data Types, Null values and Literals, Creating, Modifying and Deleting tables. Views and Indexes.
SQL - Data Manipulation Language (DML): Insert, Update, Delete, Select, Set Operations, Joins, Tuple Variables, Nested sub-queries, Query Processing.
PL/SQL (Programming Language SQL): Stored Procedures and Functions, Cursors, Triggers.
SQL - Data Control Language (DCL): Grant and Revoke commands

Unit 3: Storage and Querying (8)
Storage and file systems: Storage and File structure, Files with Fixed / Variable Length Records, Hashed Files; Indexing: Indexed Files, Single Level and Multi Level Indexes, B+ Trees

Query Processing: Overview, measures of query cost, Selection and join operations, Evaluation of expressions, Introduction to query optimization, Estimation, Transformation of Relational expressions, Sort Operation, Impact of Indices on Query Performance;

Unit 4: Transaction management **(8)**

Transactions, ACID Properties of Transactions, Concept of Schedule, Serial Schedule, Serializability: Conflict serializability, View serializability, Cascaded Aborts, Recoverable and Non-recoverable Schedules. Concurrency Control: Need, Locking Methods, Deadlocks, Timestamping methods. Recovery methods: Shadow-Paging and Log-Based Recovery.

Unit 5: Advance topics in Databases **(8)**

NoSQL Databases

Introduction to NoSQL databases: Structured and unstructured data, NoSQL- Comparative study of SQL and NoSQL databases, Big data. BASE Properties, Types of NoSQL databases- Key-value store – JSON, Document Store – MongoDB: CRUD Operations, Indexing, Aggregation and MapReduce in MongoDB.

Special purpose databases :

Cloud, in memory, Spatial databases etc.

Introduction to data mining and machine learning

Text Books:

1. Abraham Silberschatz, Henry F. Korth, and S. Sudarshan, 'Database System Concepts', McGraw Hill, (6 th edition), (2013)
2. Jiawei Han, Micheline Kamber and Jian Pei, 'Data Mining – Concepts and Techniques', Morgan Kaufmann Publishers,(3 rd Edition), (2012)
3. Kristina Chodorow, Michael Dirolf, 'MongoDB: The Definitive Guide' , O'Reilly, (2 nd Edition), (2013)
4. Ramez Elmasri and Shamkant B. Navathe, 'Database Systems',Pearson, (6 th Edition), (2013)

References:

1. Raghu Ramakrishnan and Johannes Gehrke, 'Database Management Systems', McGraw Hill, (3 rd Edition), (2003)
2. C. J. Date, 'An Introduction to Database Systems', Pearson, (8 th Edition), (2006)
3. Thomas Connally, Carolyn Begg, 'Database Systems', Pearson, (4 th Edition), (2012)

20CE 403 Operating Systems

Teaching Scheme
Lectures: 3 Hrs/week

Examination Scheme
In Semester :50 Marks
End Semester: 50 Marks
Credits: 3

Prerequisites:

1. Fundamentals of Programming Languages – II (20ES05)
2. Digital Systems and Computer Organization (20CE 304)

Course Objectives:
To facilitate the learner -

1. To understand basic concepts of Operating Systems.
2. To understand process life-cycle, process control block and scheduling algorithms.
3. To apply memory management strategies.
4. To understand file System concepts, protection and security.
5. To learn operating system for managing resources such as I/O, CPU, memory etc.
6. To understand Inter-process Communication and deadlock concepts.

Course Outcomes:
By taking this course, the learner will be able -

1. To build the basic knowledge of operating system.
2. To identify the process management concepts along with CPU scheduling algorithms.
3. To choose the memory management strategies.
4. To build the file management concepts along with protection and security features on various types of file.
5. To make use of the knowledge of storage devices for disk management.
6. To apply the concepts of Inter-process Communication.

Unit 1: Introduction to Operating Systems (06)

Introduction to Operating System (OS), Evolution of OS, Functions of OS, Types of OS, OS Concepts: Process, Files, Shell and its types, System calls and its types , Kernel and its types, Virtual Machine, Bootstrapping and shutdown, Case Study of introduction to UNIX Operating System.

Unit 2: Process and CPU Scheduling (08)

Process Concept, Operations On Processes, Creation, Termination, States, Transition and Context Switching, Process control block, Scheduling Criteria, Scheduling Algorithm, First-Come First-Serve (FCFS), Shortest Job First (SJF), Round-Robin (RR), Introduction to Threads and Benefits, Comparison of Thread and Process, Process related system calls (in UNIX) , Case Study of UNIX Process Management.

Unit 3: Memory Management **(08)**

Contiguous and Non-Contiguous Memory, Swapping, Paging, Segmentation, Virtual Memory, Demand Paging, Page Replacement Algorithms- First-In First-Out (FIFO), Least Recently Used (LRU),Optimal, Allocation of Frames and Trashing, Case Study of UNIX memory management.

Unit 4: Introduction to the File System (06)

File Concepts, File Attributes, File Operations, File Types, File Sharing, File Structure, Directory Overview, Types of Directories, File System Implementation- inode block/table, super block, data block, boot block, Allocation Methods, Types of Users, Access Modes, Introduction to Protection and Security, Case Study of UNIX File Structure.

Unit 5: I/O Management and Disk Scheduling **(06)**

I/O Devices, Organization of I/O Functions, Operating System Design Issues Related to I/O, I/O Buffering, Disk Scheduling - First Come-First Serve (FCFS), SCAN, Circular SCAN (C- SCAN), Shortest Seek Time First (SSTF).

Unit 6: Inter-Process Communication (IPC) **(08)**

Critical Section Problem, Hardware Support for Mutual Exclusion, Semaphores, Classical Problems of Synchronization, Monitors, Deadlocks, Methods of Handling Deadlock, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection and Recovery from Deadlock.

Text Books:

1. Silberschatz, Galvin, Gagnes, "Operating System Concepts", John Wiley & Sons, (8/e), ISBN: 9971- 51-388-9.
2. William Stallings, "Operating System-Internals and Design Principles ", Prentice Hall India, (5/e) ISBN: 81-297-0 1 094-3.
3. Maurice J. Bach, “The Design of the Unix Operating System”, Pearson Education, ISBN: 81-7758- 770-6.

Reference Books:

1. Evi Nemeth, Garth Snyder, Tren Hein, Ben Whaley, “Unix and Linux System Administration Handbook”, (4/e), ISBN: 978-81-317-6177-9. (2011).
2. Milan Milenkovic, “Operating Systems”, TMH, (2/e), ISBN: 0-07-044700-4.
3. Andrew S. Tanenbaum, "Modern Operating Systems", Prentice Hall India, (4/e), ISBN: 81-203- 2063-8.

Web References:

1. NPTEL: Computer Science and Engineering: Introduction to Operating Systems: https://nptel.ac.in/courses/106/106/106106144/
2. NPTEL: Computer Science and Engineering: Operating System Fundamentals: https://nptel.ac.in/courses/106/105/106105214/

20CE 404 Machine Learning

Teaching Scheme

Lectures: 3 Hours / Week

Examination Scheme

In Semester: 50 Marks

End Semester: 50 Marks

Credits: 3

Course Objectives:

To facilitate the learner to

1. Introduce students to the basic concepts and techniques of Machine Learning.
2. Utilize data pre-processing techniques and dimensionality reduction techniques for given data.
3. Become familiar with supervised machine learning algorithms such as regression, classification
4. Become familiar with unsupervised machine learning algorithms such as clustering, association rule mining method
5. Become familiar with Artificial neural networks and its learning algorithms.
6. Evaluate the performance of the designed machine learning model.

Course Outcomes:

After completion of the course, students will be able to

1. Acquire fundamental knowledge of machine learning theory.
2. Make use of data pre-processing and dimensionality reduction technique for given data.
3. Apply supervised machine learning techniques such as classification and regression for problem solving and evaluate the designed technique using performance measures.
4. Solve the problems using various unsupervised machine learning techniques such as clustering and association rule mining
5. Apply the artificial neural network technique to solve the problem.

Unit I: Introduction to Machine learning (6)

Types of Learning: Rote Learning, Learning by General Problem Solving, Concept Learning, Learning by Analogy, learning problems and designing the learning systems, Machine

Learning: Types of Problems in Machine Learning, Aspects of Inputs to Training, Supervised, unsupervised, semi supervised, reinforcement learning, overfitting, underfitting, best practices in machine learning, Intelligent Agents.

Unit II: Data Pre-processing and Dimensionality Reduction (6)

Data cleaning, data integration, data reduction, data transformation and data discretization, curse of dimensionality, Principle Components Analysis, Bias/Variance trade-off.

Unit III: Supervised Learning (12)

Regression: Correlation and regression, line fitting by least square, outliers, linear and multiple regression
Classification: Logistic regression, Nearest Neighbour Classification: K-nn, Introduction to Decision tree and Bayesian Classification

Performance Measures: Confusion matrices, accuracy, sensitivity, specificity, kappa statistics, precision, recall, F-measure, Methods of cross-validation, Types of Errors: RMSE, MSE etc

Unit IV: Unsupervised Learning (10)

Introduction to Clustering methods, k-means clustering, Hierarchical clustering: agglomerative clustering method, decisive clustering method
Market Basket analysis, Apriori Algorithm, Association rule mining, Outlier analysis

Unit V: Introduction to Artificial Neural networks (8)

Supervised learning: McCulloch-Pitts model, Perceptron model, multi-layer perceptron, feed forward networks, Perceptron learning algorithm
Unsupervised learning: Self organizing maps

Text Books:

1. Peter Flach, "Machine Learning: The Art and Science of Algorithms that make sense of data", Cambridge University Press, 1st Edition, 2015, ISBN No.: 978-1-316-50611-0
2. Ethem Alpaydin, "Introduction to Machine Learning", PHI, 2nd edition, 2013, ISBN 978-0- 262-01243-0

3. Jiawei Han, Micheline Kamber, Jian Pei, "Data Mining: Concepts and Techniques", 3rd Edition, 2012, Morgan Kaufmann publishing, ISBN: 978-0-12-381479-1

4. V. Susheela Devi, M. Narasimha Murty, "Pattern Recognition: An introduction", University Press, 2011, ISBN 978-81-7371-725-3

Reference Books:

1. Rodolfo Bonnin, "Machine learning for developers" , Packt publication, 2017, ISBN 978-1-78646-987-8

2. Vinod Chandra S. S., Anand Hareendran S., 'Artificial Intelligence and machine learning', PHI, (2014), ISBN 978-81-

3. Tom M. Michell, 'Machine Learning', McGraw Hill Education, Indian edition 2013, ISBN 978-1-25-909695-2.

4. John Paul Mueller, Luca Massarom, "Machine learning (in python and R) dummies", Wiley publication, 2016, ISBN 978-81-265-63050

5. Manohar Swamynathan, "Mastering Machine learning with python in six steps", Apress publication, 2018, ISBN 978-1-484-24044-1

Online/Web/Other References:

1. Nptel/coursera courses on Machine Learning

20CE 402L Database Management Systems Laboratory

Teaching Scheme
Practical: 04 Hours/Week

Examination Scheme
In-semester: 25 Marks
Practical: 25 Marks
Credits: 2

Course Objectives:
To facilitate learners to-
1. Implement/Execute Structured Query Language (SQL) queries.
2. Implement/Execute PL/SQL stored procedures and functions.
3. Implement/Execute MongoDB queries.
4. Develop database applications.

Course Outcomes:
On completion of the course, student will be able to–
1. Apply the knowledge of Structured Query Language (SQL) clauses to query the relational database.
2. Apply the knowledge of PL/SQL to solve the given business problem.
3. Apply the knowledge of NoSQL databases to query semi structured documents.
4. Solve the given database problem using database programming skills.

The lab is designed in such a way that the student can apply the DBMS concepts and implement the SQL commands. Once they are thorough with the SQL commands, they can build a database application using front end and back end concepts. The students can handle the semi-structured data and query it by using MongoDB commands. Motivation here is that the students should get a good practice of the SQL, PL/SQL and MongoDB syntax. Faculty members are encouraged to use different database systems and design different queries such that all the SQL clauses are covered. Group A assignments are mandatory and four assignments from group B can be given to the students such that each assignment is implemented in some or the other batch. Group C consists of a mini project which aims at giving the students an experience of building a database application from the scratch.

Example List of Assignments for the Laboratory

GROUP A (Mandatory)

1. Design and Execute SQL Data Definition Language (DDL) commands to create tables and insert data into the tables.
2. Design and Execute SQL queries for suitable database application using SQL Data Manipulation Language (DML) commands: Insert, Select, Update and Delete.
3. Design and execute SQL queries for suitable database application using SQL DML statements: all types of Joins and Sub-Query. Implement the concept of index and View.
4. Build a 2-tier database application.
5. Create a MongoDB collection and implement the CRUD operations.

GROUP B (Any 4)

1. Consider a library management system. Create a PL/SQL stored procedure for returning a book. Make suitable assumptions wherever necessary.
2. Consider a payroll system of a company. Write a PL/SQL stored procedure for calculating the income tax of employees of the company. Make suitable assumptions wherever necessary.
3. Consider a student information system in which total marks scored by the student is inserted by the teacher. Write a PL/SQL stored procedure for populating the "Class" field in the 'Students' table which indicates the class secured by every student in the class. If marks>=75% then Class= Distinction. If marks>=60% and marks<75% then First Class. If marks>=50% and marks <60% the Higher Second Class. If marks>=40% and Marks<50% then Second class. If marks<40% then class= Fail
4. Consider a Library management system where there are two tables in which the Books data is stored. Write a PL/SQL block of code that will merge the data from the "old_Books" table to the "new_Books" table. If the data in the first table already exist in the second table then that data should be skipped. Make suitable assumptions wherever necessary.
5. Consider an Employee management system of a company. Write a database trigger which will ensure that when data is inserted in the EMPLOYEES table, the department name is always in Upper case.
6. Consider the accounts system of a company. Write a database trigger which will ensure that when data in the Accounts table is updated, the old copy is preserved in the "Transaction_Log" table along with the date and userID.
7. Consider an Employee management system of a company. Write a database trigger which will ensure that when data in the EMPLOYEE table is deleted, it is first copied in the Ex-employees table along with the date of deletion.
8. Consider a student management system of our college. Write a PL/SQL function to calculate the number of distinction holders, first class holders, second class holders in the class. Make suitable assumptions wherever necessary.
9. Implement aggregation and indexing in MongoDB.
10. Implement Map reduce operation in MongoDB.

GROUP C (Mandatory)

Mini Project

Mini project aims at giving students a hands-on experience of building a Database application from the scratch and includes following tasks and deliverables:

- Gather requirements for the system and generate the requirements document
- Design the database system for the above requirements using ER-Diagram
- Convert the ERD to tables and normalize the tables up to Third Normal Form.
- Create a 2tier/ 3tier database application using frontend and backend concepts.
- Demonstrate the mini project on completion.

20CE 404L Machine Learning Laboratory

Teaching Scheme
Practical: 4 Hours / Week

Examination Scheme
In Semester : 25 Marks
Oral : 25 Marks
Credits: 2

Prerequisite:
1) 20 CE 306 Programming Skills Development-I Laboratory

2) 20ES05 Fundamental of Programming Languages-II

Course Objectives:
To facilitate the learner to

1. Implement some pre-processing operations on given data.
2. Implement supervised machine learning algorithms such as regression, classification.
3. Implement unsupervised machine learning algorithms such as clustering, association rule mining method.
4. Implement artificial neural networks and its learning algorithms.
5. Implement a small machine learning application and evaluate the performance of the designed machine learning model.

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

1. Apply pre-processing operations on given data.
2. Apply the classification and regression machine learning techniques to solve the problem.
3. Apply various clustering and association rule mining techniques of machine learning to solve the problem.
4. Apply the artificial neural network technique to solve the problem.
5. Develop small machine learning applications using different techniques.

A large part of the lab would be for understanding the basic concepts of machine learning and implementation of some real-world simple applications. Assignment statements are in brief and should be implemented in JAVA/Python programming language. Motivation here is that students should be able to code the basic algorithm and also should be able to make use of built-in functions available in different libraries of Java/Python. Faculty members are encouraged to expand problem statements with variations. Assignments can be framed and expanded in such a way that it explores concepts, logic of solution and simple application. Students will be encouraged to solve open problems in different domains. Faculty will appropriately adopt assignments on similar lines as the examples shown here. Group A assignments are on pre-processing data, supervised learning methods such as classification and regression, and simple logic gates implementation using artificial neural network. Group B assignments are on unsupervised learning and Group C assignment is on case study implementation for

different application.

Suggestive List of Assignments

Group A : (Mandatory)

1. Explore language used for Machine Learning Python/Java and perform the following operations: Understand the basic functionality, visualization of data. Study the different file format, explore the available data sets and its usage using programming language.

2. Suppose that the data for analysis includes the attribute age: 13, 15, 16, 16, 19, 20, 20, 21, 22, 22, 25, 25, 25, 25, 30, 33, 33, 35, 35, 35, 35, 36, 40, 45, 46, 52, 70. (a) Plot an equal-width histogram of width 10. (b) Sketch examples of each of the following sampling techniques: SRSWOR, SRSWR, cluster sampling, and stratified sampling. Use samples of size 5 and the strata "youth," "middle-aged," and "senior." (c) scale and also normalize the data. Pre-process the given data as given here.

3. BMI and body fat of persons are given. Use this BMI to predict the body fat of a person. Implement predictive modelling using regression analysis using a programming language that you are familiar with such as Java/Python. Fit the model and predict the value for given problem.

4. Data given for the SPEED and AGILITY rating of 20 college athletes and whether they were drafted by professional team. Implement k-nn classification technique of Machine learning using a programming language that you are familiar with such as Java/Python. Compare the performance of classification by changing value of k for the given data.

5. Build the logic gates AND, OR, NOT, NOR, NAND gates using ANN assuming random initialization. Write a program to implement Perceptron learning in an artificial neural network using Java/Python.

Group B: (Any Two)

1. You have a list of shopping items purchased by many people. Find out what are the frequently purchased combination of 2 items. Implement Apriori, a Frequent Pattern Analysis algorithm using Java/Python.

2. One of the earliest and well-known applications of the SOM is the phonetic typewriter of Kohonen. It is set in the field of speech recognition, and the problem is to classify phonemes in real time so that they could be used to drive a typewriter from dictation. The real speech signals obviously needed pre-processing before being applied to the SOM. Simulate this application where 4-dimensional input space is mapped to 2 nodes. Write a program to implement Self Organising Map (SOM) using Java/Python.

3. A Hospital Care chain wants to open a series of Emergency-Care wards within a region. We assume that the hospital knows the location of all the maximum accident-prone areas in the region. They have to decide the number of the Emergency Units to be opened and the location of these Emergency Units, so that all the accident-prone areas are covered in the vicinity of these Emergency Units.

   The challenge here is to decide the location of these Emergency Units so that the whole region is covered. Implement a K-means clustering algorithm using a programming language that you are familiar with such as Java / Python. Compare the performance of your algorithm on the dataset by changing input parameter value such as K

Group C

1. Machine learning case study for readily available data sets using the techniques studied, and evaluate the designed and implemented model.

20CE 403L Operating Systems Laboratory

Teaching Scheme
Practical: 2 Hours / Week

Examination Scheme
In Semester: 25 Marks
Oral : 25 Marks
Credits: 1

Prerequisites:

1. Fundamentals of Programming Language Lab-II (20ES05L)
2. Data Structures and Algorithms (20CE 302)
3. Digital Systems and Computer Organization(20CE 304)

Course Objectives:
To facilitate the learner to

1. Understand the fundamentals of Operating Systems.
2. Understand shell scripting to automate operating system operations.
3. Apply the concepts of Operating System for Process and Memory management.
4. Understand the operations performed by the Operating System as a resource manager.
5. Understand the communication among the processes.

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

1. Choose UNIX/Linux Commands for Shell Programming.
2. Make use of different CPU scheduling algorithms.
3. Apply Memory Management algorithms.
4. Implement various disk scheduling algorithms.
5. Explore the Inter-Process Communication concepts.

Preamble:
Operating Systems Laboratory is designed to understand and implement the fundamental concepts of Operating System. Motivation here is that students should be able to code the basic shell script and make use of various Operating system services algorithms. Faculty members are encouraged to expand problem statements with variations. Assignments can be framed and expanded in such a way that it enhances the concepts and logic of the solution. Students will be encouraged to solve open problems in different operating systems. Faculty will appropriately adopt assignments on similar lines as the examples shown here. Group A assignments are on Unix/Linux commands, shell script and process creation using system calls. Group B assignments are on simulation of schedulers, memory and I/O management algorithms and Inter-process communication problems using Semaphores. Group C assignment is on case study for different Operating Systems and its services.

Suggestive List of Assignments

Group A : (Mandatory)

1. Demonstration of Installation of Linux Operating System.
2. Exploration of Unix/Linux Commands (File, Directory and Process commands).
3. Write a shell script for adding users / groups and modifying permissions of file / directory accordingly.
4. Write a program to implement operations on processes using fork and join system calls.

Group B: (Any Four)

5. Simulation of the scheduling algorithms. For example: First Come First Serve (FCFS), Shortest Remaining Time First (SRTF).
6. Simulation of scheduling algorithms. For example: Round-Robin (RR), Shortest Job First (SJF).
7. Simulation of memory allocation strategies. For example: First Fit, Best Fit and Worst Fit.
8. Simulation of Page replacement algorithms. For example: First-In-First-Out (FIFO), Least Recently Used (LRU), optimal page replacement.
9. Simulation of disk scheduling algorithms. For example: First Come First Serve (FCFS), SCAN, Circular – SCAN (C-SCAN), Shortest Seek Time First (SSTF).
10. Write a program to implement Banker's Algorithm for deadlock handling.
11. Write a program to implement Reader-Writer problem using semaphores.

Group C

Case study of various Operating systems services. (Example: Android, RTOS, Linux, IOS, Windows etc.)