### Autonomous Programme Structure of
Third Year B Tech Information Technology
Academic Year 2018-2019

**Third Year B. Tech. Information Technology Semester – 5**

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
<th>Teaching Scheme</th>
<th>Examination Scheme</th>
<th>Marks</th>
<th>Credit</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Hours /Week</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Lecture</td>
<td>Tutorial</td>
<td>Practical</td>
<td>In Semester</td>
</tr>
<tr>
<td>IT 3101</td>
<td>Database Management Systems</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>50</td>
</tr>
<tr>
<td>IT 3102</td>
<td>Theory of Computations</td>
<td>3</td>
<td>1</td>
<td>0</td>
<td>50</td>
</tr>
<tr>
<td>IT 3103</td>
<td>Machine Learning</td>
<td>3</td>
<td>1</td>
<td>0</td>
<td>50</td>
</tr>
<tr>
<td>OEHS 3101</td>
<td>Open Elective- I</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>50</td>
</tr>
<tr>
<td>PEIT 3101</td>
<td>Program Elective- I</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>50</td>
</tr>
<tr>
<td>IT 3104</td>
<td>Database Management Systems Laboratory</td>
<td>0</td>
<td>0</td>
<td>4</td>
<td>25</td>
</tr>
<tr>
<td>IT 3105</td>
<td>Machine Learning Laboratory</td>
<td>0</td>
<td>0</td>
<td>4</td>
<td>25</td>
</tr>
<tr>
<td>PEIT 3106</td>
<td>Program Elective I Laboratory</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>AC 3101</td>
<td>Audit Course</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td><strong>Total</strong></td>
<td><strong>15</strong></td>
<td><strong>2</strong></td>
<td><strong>12</strong></td>
<td><strong>300</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Grand Total</strong></td>
<td><strong>29</strong></td>
<td></td>
<td><strong>625</strong></td>
<td></td>
</tr>
</tbody>
</table>

**OEHS 3101 Elective 1**
1. Entrepreneurship Development
2. Introduction to Digital Marketing
3. Intellectual Property Rights
4. Project Management

**PEIT 3101 Program Elective-I**
1. Computer Graphics and Animation
2. Artificial Intelligence
3. Business Intelligence
4. Object Oriented Modeling and Design

AC 3101: Audit Course: Employability Skills Development
IT 3101 Database Management Systems

Teaching Scheme:
Lectures: 3 hrs/week
Tutorial: NIL

Examination Scheme:
In-Semester: 50 marks
End-Semester: 50 marks
Credits: 3

Prerequisites: Data structures

Course Objectives:
Familiarize students with
1. Concepts and applications of database management.
2. Different models and normalization used for database design.
3. Query languages in databases.
4. Basic issues of database design and utilization.

Course Outcomes:
Students should be able to
1. Identify basic purpose and functions of database management system.
2. Build appropriate database schema for the given application.
3. Make use of query commands and concurrency control protocols.
4. Analyze database for given problem domain.

Unit – I: Introduction to DBMS


Unit – II: Database Design and SQL

Database Design, Functional Dependency, Purpose of Normalization, Data Redundancy, Anomalies. Normal forms 1NF, 2NF, 3NF, BCNF, 4NF and 5NF. Introduction to SQL, SQL Data Types, DDL, DML and DCL queries, Views, Indexes, Null handling, Nested Queries. PLSQL. Query optimization

Unit – III: Database Transactions

Basic concept of a Transaction, Transaction Management, Properties of Transactions, Concept of Schedule, Serial Schedule, Serializability, Conflict and View, Cascaded Aborts, Recoverable and Non recoverable Schedules.
Unit – IV: Concurrency control and Advanced Database Architectures (07)

Concurrency Control, Locking Methods, Deadlocks, Protocols, Recovery Methods, Database Architectures, Centralized and Client-Server Architectures, 2 Tier and 3 Tier Architecture, Indexing and hashing, Parallel Databases, Distributed Databases.

Unit – V: No SQL and semi structured Data Management (07)

Introduction to Big Data, No SQL Databases, MongoDB, Map reduce. XML Databases, DTD, XML Schemas, XQuery, XPath. JSON

Unit – VI: Data Warehousing and Data Mining (07)

Data Warehousing, Architecture and features of Data Warehouse, ETL Process, OLAP. Data Mining, Knowledge Discovery, Data Mining techniques, Applications of data mining.

Text Books:


Reference Books:

IT 3102 Theory of Computations

Teaching Scheme:  
Lectures: 3 hrs/week  
Tutorial: 1 hr/week

Examination Scheme:  
In-Semester: 50 marks  
End-Semester: 50 marks  
Credits: 4

Prerequisites: Discrete structures

Course Objectives:  
Familiarize students with  
1. Abstract computing models.  
2. Types and applications of formal grammars  
3. Application of Theory of Computer Science in System Programming

Course Outcomes:  
Students should be able to  
1. Construct abstract computing models  
2. Apply the concepts of formal grammars  
3. Analyze Decidable Languages and Reducibility  
4. Evaluate computing models.

Unit – I Fundamentals (07)  
Strings, Alphabet, Language, Operations, Finite state machine, definitions, finite automaton model, acceptance of strings, and languages, deterministic finite automaton and non-deterministic finite automaton, transition diagrams and Language recognizers. NFA to DFA conversion

Unit – II Finite Automata with application (07)  
NFA with e transitions - Significance, acceptance of languages, Equivalence between NFA with and without e transitions, minimization of FSM, equivalence between two FSM’s, Finite Automata with output- Moore and Mealy machines.

Lexical analyzer as an application of Finite Automaton. Introduction to Lex tool

Unit – III Regular Expression and Grammar Formalism (07)  
Regular expressions: Identity rules, Constructing finite Automata for a given regular expressions, Conversion of Finite Automata to Regular expressions. Pumping lemma of regular sets, closure properties of regular sets

Introduction to Grammar: derivation trees, sentential forms. Right most and leftmost derivation of strings, Chomsky hierarchy
Unit – IV  Regular Grammar with application (07)
Regular grammars-right linear and left linear grammars, equivalence between regular grammar and FA, inter conversion, Parsing techniques, Top-down parsing, Bottom-up parsing

Recursive descent parser as an application of Regular Grammar. Introduction to YACC tool

Unit – V  Context free grammars and Push down automata (07)
Context Free Grammars-Ambiguity in context free grammars. Minimization of Context Free Grammars. Normal Forms Chomsky Normal Form, Greibach Normal Form, conversion to CNF and GNF

Push down automata- definition, model, acceptance of CFL, Acceptance by final state and acceptance by empty state and its equivalence. Equivalence of CFL and PDA, inter conversion,

Unit – VI  Turing Machine (07)
Turing Machine, definition, model, design of TM, Computable functions, recursively enumerable languages. Church’s hypothesis, counter machine, types of Turing machines, Universal Turing Machine, decidability/un decidability of problems, Halting problem Correspondence problem, Turing reducibility

Modularized programming concept as an application of Turing machines

Text Books

Reference Books
2. Hopcroft Ulman, “Introduction To Automata Theory, Languages And Computations”, Pearson Education Asia, 2nd Edition
IT 3103 Machine Learning

Teaching Scheme:
Lectures: 3 hrs/week
Tutorial: 1 hr/week

Examination Scheme:
In-Semester: 50 marks
End-Semester: 50 marks
Credits: 4

Prerequisites: Linear Algebra and Calculus, Probability Basics

Course Objectives:
Familiarize students with
1. Basic learning algorithms and techniques
2. Applications of machine learning
3. Usage of large data sets

Course Outcomes:
Students should be able to
1. Identify different tasks in machine learning
2. Explain wide variety of learning algorithms and techniques
3. Apply proper learning algorithm to data depending on the task
4. Perform evaluation of learning algorithms

Unit – I  Introduction to Machine Learning (07)
Introduction: What is Machine Learning, Examples of Machine Learning applications, Training versus Testing, Positive and Negative Class, Cross validation
Types of Learning: Supervised, Unsupervised and Semi-Supervised Learning
Features: Types of features-Continuous, Discrete, Nominal, Ordinal. Extraction of Features from text document and image

Unit – II  Classification (07)
Binary and Multiclass Classification: Assessing Classification Performance, Performance of multi-class classification
Linear and Non-linear Models: Perceptron, Support Vector Machines (SVM), Soft Margin SVM, Kernel methods for non-linearity

Unit – III  Regression and Generalization (07)
Regression: Assessing performance of Regression – Error measures, Overfitting and Underfitting, Catalysts for Overfitting
Linear Models: Least Square method, Univariate Regression
Theory of Generalization: Bias and Variance Dilemma, Training and Testing Curves
Case Study of Polynomial Curve Fitting

Unit – IV  Distance Based Models (07)
Neighbors and Examples, Distance Measures: Euclidian, Manhattan, Minkowski, Hamming. Nearest Neighbor Classification (kNN), Distance based clustering algorithms - K-means, K-medoid, DBScan, Hierarchical Clustering: Single, Complete, Average and Centroid Linkage
Unit – V Rule and Tree based Models

Rule Based Models: Frequent Itemsets, Association rules mining – Apriori Algorithm, Confidence and Support parameters


Unit – VI Probabilistic Models


Text Books


Reference Books

PEIT 3101 Computer Graphics and Animation

Teaching Scheme: Lectures: 3 hrs/week

Examination Scheme:
In-Semester: 50 marks
End-Semester: 50 marks
Credits: 3

Prerequisites: Basics of Programming, Essential mathematics in geometry and trigonometry, Vectors and Matrices

Course Objectives:
Familiarize students with
1. Basic concepts of computer graphics
2. Basic primitives and objects in computer graphics
3. Various methods and techniques used in computer graphics
4. Applications of computer graphics in animation and gaming

Course Outcomes:
Students should be able to
1. Identify geometrical formulas and algorithms to draw computer graphics primitives
2. Use mathematics to transform computer graphics objects
3. Apply various techniques to achieve desired image manipulation.
4. Design algorithmic logic to solve complex problem like gaming

Unit – I Basic Concepts (07)
Introduction: Basics of graphics systems, Raster scan & Random scan displays, basic display processor. Display Files: display file structure, algorithms and display file interpreter. Primitive operations.
Plotting Primitives: Scan conversions, line segments, vectors, pixels and frame buffers, vector generation. Introduction to OpenGL: Basic OpenGL syntax, display-window management using GLUT, functions.

Unit – II Drawing and Filling Graphics Primitives (07)
Line and Circle drawing Algorithms: DDA, Bresenham’s, Midpoint.
Character Generation: Stroke Principle, Starburst Principle, Bit map method, aliasing and anti-aliasing
Polygon: Polygon and its types, inside test, polygon filling methods: Seed fill, Scan Line, Flood fill and Boundary fill

Unit – III Geometric Transformations (07)
2D Geometric Transformations: Translation, scaling, rotation, reflection, shearing, matrix representation and homogeneous coordinate system, Composite transformations
3D Geometric Transformations: Translation, scaling, rotation, rotation about X, Y, Z and arbitrary axis reflection about XY, YZ, XZ and arbitrary plane.
Unit – IV  Segments, Windowing and Clipping  
   (07)
   Segment:  Introduction, Segment table, Segment creation, closing, deleting and renaming, Visibility
   Windowing:  Concept of window and viewport, viewing transformations
   Line Clipping:  Cohen Sutherland Method, Midpoint subdivision method
   Polygon Clipping:  Sutherland Hodgman method for clipping convex and concave polygon

Unit – V  Shading and Animation  
   (07)
   Shading:  Halftoning, Gouraud and Phong Shading
   Computer Animation:  Design of Animation sequences, General Computer Animation functions, Computer Animation Languages, Key-frame Systems, Motion Specifications.

Unit – VI  Gaming  
   (07)
   Gaming platforms:  Graphics Memory Pipeline, Block diagram of NVIDIA workstation and i860
   Introduction to OpenGL ES
   Interactive Graphics & usage of the tools of computer graphics:  3D Studio and Maya
   2D games :Snake game

Text Books


Reference Books

PEIT 3101 Artificial Intelligence

Teaching Scheme:
Lectures: 3 hrs/week

Examination Scheme:
In-Semester: 50 marks
End-Semester: 50 marks
Credits: 3

Prerequisites: Discrete mathematics, basic probability theory and statistics
Knowledge of any programming language and data structures

Course Objectives:
Familiarize students with
1. The basic principles and applications of Artificial Intelligence.
2. Concepts of problem solving and knowledge representation
3. Concepts of planning and learning

Course Outcomes:
Students will be able to:
1. Assess underlying AI concepts and their usage.
2. Implement classical Artificial Intelligence techniques, such as search algorithms, minimax algorithm, and neural networks.
3. Represent knowledge using logic and infer new facts from it.
4. Apply Artificial Intelligence techniques for problem solving.

Unit – I Artificial Intelligence (07)

Unit – II Problem Solving (07)

Unit – III Knowledge Representation (07)
Logical Agents: Knowledge-Based Agents, Propositional logic, First-order Logic, Knowledge Representation: Ontological Engineering, Categories and Objects, Events, Reasoning Systems for Categories, Reasoning with Default Information

Unit – IV Planning (07)

Unit – V Reasoning and Learning (07)
Unit – VI  Artificial Neural Network


Text Books


Reference Books

2. Charniack and D. Mcdermott, Artificial Intelligence, Addison Wesley
6. Toby Segaran, Programming Collective Intelligence, O’Reilly
PEIT 3101 Business Intelligence

Teaching Scheme:
Lectures: 3 hrs/week

Examination Scheme:
In-Semester: 50 marks
End-Semester: 50 marks
Credits: 3

Prerequisites: Linear algebra, probability basics

Course Objectives:
Familiarize students with
1. The role of Business Intelligence in various business applications
2. Methods of data processing and modeling
3. Importance of visualization and reporting in business
4. Decision making process using Business Intelligence

Course Outcomes:
Students will be able to:
1. Identify business problems to provide BI solutions
2. Use data transformation and modeling concepts for building data warehouse
3. Analyze and visualize dimensional models for reporting.
4. Comprehend different BI trends and their applications.

Unit – I Introduction (07)
Concepts of Data, Information, and Knowledge, Design and implementation aspect of OLTP and OLAP/Data Warehouse, Business Intelligence(BI) Concepts and definitions, BI architectural models (Top-down and bottom-Up), Business Applications of BI, Role of Data warehouse in BI, BI system components

Unit – II Dimensional Modeling And Data Warehouse Design (07)
Star schema, Snow flake schema, and Fact Constellation schema, Grain of dimensional model, transactions, Recurring Snapshots, Accumulating Snapshots, Dimensions (SCD types, conformed dimensions), Facts (additive, semi-additive, non-additive), Junk dimensions, conformed dimensions, Bridge tables

Unit – III ETL (07)
Data Quality, Data profiling, Data enrichment, data duplication, Data cleaning, ETL Architecture and what is ETL, Extraction concept and Change data capture, Transformation concept, Loading concept, Initial and Incremental loading, Full loading, late arriving facts, data staging, Data marts

Unit – IV Reporting (07)
Metadata Layer, Presentation Layer, Data Layer, Use of different layers and overall Reporting architecture, Various report elements such as Charts, Tables, Materialized views, Query rewrite, Ad-hoc reports, Security: report level, data level (row, column),Scheduling.

Unit – V Analytics And Data Visualization (07)
Analytics: Application of Analytics concepts in Business Intelligence, Clustering with K-Means, Classification with Decision tree, In-Memory Analytics and In-DB Analytics
**Data visualization:** Types of data visualization, Techniques for visual data representations, data Visualization tools- Tableau, Dashboards, **Case study:** Credit card fraud detection, click stream analysis

**Unit – VI Recent Trends**
Introduction to Big Data, DW appliances, Types of BI: Real time BI, Operational BI, Embedded BI, Agile BI, Smart change data capture using log based techniques
**Case Study:** BI for sales force management, Social BI systems

**Text Books**
3. Ralph Kimball, Joe Caserta, “The Data warehouse ETL Toolkit”, Publisher: Wiley
4. Jiawei Han, Micheline Kamber, Jian Pei “Data Mining: concepts and techniques”, 2nd Edition, Publisher: Elsevier/Morgan Kaufmann.

**Reference Books**
1. Ralph Kimball, Margy Ross, “The Data Warehouse Toolkit”, 3rd edition, Publisher: Wiley
2. Reema Thareja, “Data Warehouse”, Publisher: Oxford University Press.
PEIT 3101 Object Oriented Modeling and Design

Teaching Scheme:
Lectures: 3 hrs/week

Examination Scheme:
In-Semester: 50 marks
End-Semester: 50 marks
Credits: 3

Prerequisites: Object Oriented Programming

Course Objectives:
Familiarize students with
1. Need of Object oriented Modeling and design
2. Unified Modeling language diagrams for representing the modeling
3. Purpose of every UML diagram for showing an aspect of modeling
4. Applying the analysis and design methodology to a moderate complex system

Course Outcomes:
Students should be able to
1. Identify the classes, attributes for moderately complex systems.
2. Co-relate the classes with has-a or is-a relationship.
3. Refine the relationships to remove the redundancy.
4. Represent the design using UML diagrams.

Unit – I Software Complexity Understanding the challenges OOAD can address (07)
Software Complexity, Object Model, Classes and Objects, Identification approaches using OOAD

Unit – II Use Case model and Activity Model (07)
Identifying use cases from requirements, Representing business flow as an activity diagram.
Relationships in use cases as extends, generalization, includes.
Notations of Activity diagrams an activity, join, fork, decision node, merge node, swim lane

Unit – III Domain Analysis Model (07)
Relationships among classes as association, aggregation, composite, generalization, dependency, realization, association class , binary association, ternary association, multiple inheritance

Unit – IV Interaction Model (07)
Realization of use cases, use case specifications, representing use case and a small class diagram,
showing the interaction among classes as sequence diagram

Unit – V SMART design Principles Architecture design (07)
Overview of SMART design principles, design pattern and software architecture design

Unit – VI Organization and deployment of Software (07)
Purpose of package diagram, component diagram and deployment diagram
Text Books
2. Russ Miles, “Learning UML 2.0“, O’Reilly

Reference Books
1. Grady Booch, “UML 2.0 use Guide” OMG Group
Group A: Introduction to Databases (Study assignment- Any one)

1. Study and design a database with suitable example using following database systems:
   a. Relational: SQL / PostgreSQL / MySQL
   b. Columnar: Hbase
   c. Document: MongoDB / CouchDB
   d. Graph: Neo4J
   Compare the different database systems based on points like efficiency, scalability, characteristics and performance.

2. Study the SQLite database and its uses. Also elaborate on building and installing of SQLite

Group B: SQL and PL/SQL (Minimum 6)

1. Design any database with at least 3 entities and relationships between them. Apply DCL and DDL commands. Draw suitable ER/EER diagram for the system.
2. Execute DDL statements which demonstrate the use of views. Try to update the base table using its corresponding view. Also consider restrictions on updatable views and perform view creation from multiple tables.
3. Design and implement a database and apply at least 10 different DML queries for the following task. For a given input string display only those records which match the given pattern or a phrase in the search string.
4. Execute the aggregate functions like count, sum, avg etc. on the suitable database. Use group by and having clauses. Retrieve the data from the database based on time and date functions.
5. Implement nested sub queries. Perform a test for set membership (in, not in), set comparison (=some, >=some, <all etc.) and set cardinality (unique, not unique).
6. Write and execute suitable database triggers.
7. Write and execute PL/SQL stored procedure and cursor to perform a suitable task on the database.

**Group C: NoSQL and Semi structured Databases (Minimum 3)**

1. Create a database with suitable example using MongoDB and implement
   - Inserting and saving document
   - Removing document
   - Updating document
2. Execute at least 15 different queries on any suitable MongoDB database that demonstrates following querying techniques:
   - find and findOne
   - Query criteria
   - Type-specific queries
   - $ where queries
   - Create and drop different types of indexes
3. Implement Map reduce example using Mongo DB.

**Group D: Mini Project / Database Application Development**

Student group preferably of size 4 students should decide the statement and scope of the project which will be refined and validated by the faculty.

Choose database as per the requirement of the mini project. Draw and normalize the design up to at ER Diagram with normalization in case of back end as RDBMS. Design front end using any open source technology and perform connectivity to the database. Implement suitable database operations along with business logic, validations, reports etc.
IT 3105 Machine Learning Laboratory

Teaching Scheme:
Practical: 4 hrs/week

Examination Scheme:
In-Semester: 25 Marks
Practical: 25 marks
Credits: 2

Prerequisites: Linear Algebra and Calculus, Probability Basics

Course Objectives:
Familiarize students with
1. Various tasks in Machine Learning
2. Different Machine Learning algorithms
3. Applications of machine learning algorithms for accomplishing given tasks

Course Outcomes:
Students should be able to
1. Apply proper learning algorithm to data depending on the task
2. Compare different learning algorithms performing similar tasks
3. Use large data sets
4. Evaluate the models

Implementation of programs to be done in Python
1. Classify data using Linear Support Vector Machine algorithm
2. Predict values using Linear Regression with one independent variable and one dependent variable
3. Cluster data using k-means algorithm for clustering
4. Identify frequent item item-sets using Apriori algorithm
5. Classify data using Naïve Bayes Classification algorithm

Text Books
1. Andreas Muller and Sarah Guido: Introduction to Machine Learning with Python, O’Reilly, 2017
2. Michael Bowles: Machine Learning in Python, Wiley, 2018

Reference Books
2. Jiawei Han, Micheline Kamber: Data Mining: Concepts and Techniques, Morgan Kaufmann Publishers, 3rd Edition, July 2011.
PEIT 3106 Computer Graphics Laboratory

Teaching Scheme:  
Practical: 2 hrs/week

Examination Scheme:  
Oral: 25 marks

Credits: 1

Prerequisites: Basics of Programming, Data Structures, Algorithms, Geometry, Trigonometry, Vectors and Matrices

Course Objectives:
Familiarize students with
1. Various methods and techniques used in computer graphics
2. Applications of computer graphics in animation and gaming

Course Outcomes:
Students should be able to
1. Apply mathematics and algorithms to draw computer graphics primitives.
2. Apply graphics data manipulation in an application.
3. Implement programs using different computer graphics algorithm.
4. Make use of OPEN GL to implement programs

List of Assignments

1. Get Familiar with basic OpenGL environment, display-window management using GLUT, OpenGL functions.
2. Write a function in OpenGL on Linux Platform to draw a Line using DDA/ Bresenham’s Line Drawing Algorithm. Call the Function to draw any pattern consisting at least 10 function calls.
3. Write a function in OpenGL on Linux Platform to draw a circle using Midpoint Circle Drawing Algorithm. Call this function at least 6 times to draw any pattern. User should only give centre coordinates and radius. Rest should be drawn automatically
4. Write a program in OpenGL on Linux Platform to draw chess board using any Line drawing algorithm and fill alternate blocks using flood fill algorithm
5. Write a program in OpenGL on Linux Platform to draw a flag using any Line drawing algorithm and fill it using scanline polygon filling algorithm.
6. Write a program in OpenGL on Linux Platform to for drawing a polygon and perform following 2DTransformations on Triangle. Translation, Scaling, Rotation
7. Write a program in OpenGL on Linux Platform to clip a Line using Cohen Sutherland Outcode Method.
8. Write a program in OpenGL on Linux Platform to clip a Polygon using SutherlandHodgman Polygon Clipping.
9. Write a program in OpenGL on Linux Platform to animate a scene for “Moving Car”.
10. Write a program to design a game using computer graphics basic techniques and OpenGL
PEIT 3106 Artificial Intelligence Laboratory

Teaching Scheme:
Practical: 2 hrs/week

Examination Scheme:
Oral: 25 marks
Credits: 1

Course Objectives:
Familiarize students with
1. Basic implementation of AI concepts.
2. Current Trends in AI.

Course Outcomes:
Students will be able to
1. Implement AI core concepts using AI algorithms.
2. Identify appropriate AI techniques for development of applications.
3. Apply basic principles of AI towards problem solving, knowledge representation and learning.
4. Gain basic understanding of various AI applications in intelligent and expert systems, artificial neural networks and other machine learning techniques.

List of Assignments (Any 5)
1. Implement A* algorithm for any of the following problems: a) 8 puzzle b) Missionaries and Cannibals c) Blocks World Problem
2. Solve 8-queens problem using backtracking.
3. Implement a program to solve constraint satisfaction problem using any searching technique.
5. Implement the code for decision tree learning.
7. Implement Neural network to understand backpropagation.

Group Assignment
1. Develop application such as but not limited to
   a) Chatbot
   b) Interactive Sudoku solver
   c) Stock market predictor (offline past data)
   d) Face Recognition
   e) Captcha breakers
   f) Auto tagging of friends on social media
   g) Pac-Man
PEIT 3106 Business Intelligence Laboratory

Teaching Scheme:
Practical: 2 hrs/week

Examination Scheme:
Oral: 25 marks
Credits: 1

Prerequisites: Linear Algebra and Calculus, Probability Basics, database concepts

Course Objectives:
Familiarize students with
1. BI tools and technologies
2. Data transformation techniques and modeling
3. Implementation aspects of business analytics and reporting

Course Outcomes:
Students should be able to
1. Identify the business problem and design BI solution
2. Analyze the model
3. Visualize large datasets
4. Implement BI application

Suggested list of laboratory assignments:
Given a Business Problem as Case Study design and build BI solution using BI concepts:
1. Perform dimension modeling and Execute ETL process for building data warehouse
2. Implement OLAP operations on given data set.
3. Visualize data using various charts in Tableau
4. Develop any one application

Text Books
2. Jiawei Han, Micheline Kamber, Jian Pei “Data Mining: concepts and techniques”, 2nd Edition, Publisher: Elsevier/Morgan Kaufmann

Reference Books
1. Ralph Kimball, Joe Caserta, “The Data warehouse ETL Toolkit”, Publisher: Wiley
PEIT 3106 Object Oriented Modeling and Design Laboratory

Teaching Scheme:  
Practical: 2 hrs/week

Examination Scheme:  
Oral: 25 marks
Credits: 1

Prerequisites: Object Oriented Programming

Course Objectives:  
Familiarize students with
1. Modeling a complex system from various views
2. UML diagrams of conveying the model which in programming language independent

Course Outcomes:  
Students should be able to
1. Identify the classes, attributes for a moderately complex systems
2. Co-relate the classes with has-a or is-a relationship
3. Refine the relationships to remove the redundancy
4. Represent the design using UML diagrams

List of Laboratory Assignments

Based on a system description given to a group of four students, they will be systematically analyzing and designing the system and drawing UML diagram to show the thought process via following assignments
1. Use Case Identification and Use case diagram
2. Activity diagram to show the business process
3. Class Diagram with advanced features
4. Sequence diagrams of each use case
5. State diagram
6. Package diagram and component diagram
7. Deployment diagram