UNIVERSITY OF ENGINEERING & MANAGEMENT, JAIPUR

Course Description

Title of Course: Database Management System
Course Code: BCA401
L-T Scheme: 3-1 **Course Credits: 3**

Introduction

Database Management Systems (DBMS) consists of a set of interrelated data and a set of programs to access that data. They underpin any computer system and are therefore fundamental to any program of study in computer science. An understanding of DBMS is crucial in order to appreciate the limitations of data storage and application behavior and to identify why performance problems arise.
Students who complete this course are expected to develop the ability to design, implement and manipulate databases. Students will apply and build databases for various day to day real life scenarios and real life applications. The course will by and large be structured but will introduce open-ended data base problems.

Course Objectives:

- Ability to build normalized databases.
- Knowledge of Entity Relationship Modeling.
- Familiarity with SQL, embedded SQL and PLSQL.
- Familiarity with query processing and query optimization techniques.
- Understanding of transaction processing.
- Ability to handle recovery and concurrency issues.
- Familiarity with ODBC, JDBC.

Outcomes:

- Develop the ability to design, implement and manipulate databases.
- Introduce students to build database management systems.
- Apply DBMS concepts to various examples and real life applications.

Expected Student Background (Preconditions)

- Introduction to any programming language (Preferably, C)
- Data Structures

Course Contents:

Unit 1: Introduction to DBMS- Concept & overview of DBMS, Data Models & database Language, Database Administrator, Database Users, architecture of DBMS, Three levels of abstraction.

Unit 2: Entity Relationship Model – Basic concepts, Design Issues,Entity-RelationshipDiagram,Weak Entity Sets, Extended E-R features.

Relational Model-
Structure of relational Databases, Relational Algebra,Relational Algebra Operations,Views, Modifications of the Database.

Unit 3: SQL and Integrity Constraints:Concept of DDL, DML, DCL,Basic Structure, Set operations, Aggregate Functions,Null Values, Domain Constraints, Referential Integrity Constraints,assertions, views, Nested Sub queries.

Course Description

Unit 4: Relational Database Design: -Functional Dependency, Different anomalies in designing a Database,Normalization using functional dependencies,Decomposition, Boyce-Codd Normal Form, 3NF,Normalization using multi-valued dependencies, 4NF, 5NF.

Unit 5: Transaction: -Transaction concept, transaction model,serializability,transaction isolation level,Transaction atomicity and durability, transaction isolation and atomicity.

Concurrency control and recovery system:
Lock based protocol, dead lock handling, time stamp based and validation based protocol,failure classification, storage, recovery algorithm, recovery and atomicity,backup.

Unit 6: Internals of RDBMS:-Physical data structures, Query optimization: join algorithm,Statistics and cost based optimization.

Unit 7: File Organization & Index Structures:-File & Record Concept, Placing file records on Disk, Fixed and Variable sized Records,Types of Single-Level Index (primary secondary, clustering), Multilevel Indexes,Dynamic Multilevel Indexes using B tree and B+ tree.

Text Books:

1. Silberschatz, Korth and Sudarshan, "Database System Concepts", 6th Edition,McGraw Hill, 2010

2. Elmasri and Navathe, "Fundamentals of Database Systems", 6th Edition, Pearson, Addison-Wesley, 2010

References:

1. C.J. Date, "An Introduction to Database Systems", 8th Edition, Addison-Wesley, 2003

2. Ramakrishnan&Gherke, Database Management Systems, 2nd Edn., McGraw

3. Connolly and Begg, "Database Systems", 4th Edn., Addison-Wesley, 2005

4. Toby, Lightstone and Jagadish, "Database Modeling and Design", 5th Edn, Elsevier, 2011

5. Coronel and Rob, "Database Systems", 9th Edn.,Cengage, 2011
6. IEEE / ACM Transactions on Database Systems (TODS).
7. DBMS related Journals.

UNIVERSITY OF ENGINEERING AND MANAGEMENT, JAIPUR

Course Description

Title of Course: Principles of Computer Programming-(C++)
Course Code: BCA402
L-T-P Scheme: 3-1-0 **Course Credit: 4**

Introduction:
This course provides a comprehensive introduction to understand the underlying principles, techniques and approaches which constitute a coherent body of knowledge in C++.

Objectives:
The course presents basics of C++ programming including: Basics of C++ environment, Data representation, Control structures, Functions, Arrays, Pointers, Strings, and Classes that aims to:

- Understand object oriented programming and able to explain the difference between object oriented programming and procedural programming.
- Be able to program using C++ features such as composition of objects, operator overloads, dynamic memory allocation, inheritance and polymorphism, file I/O, exception handling, etc.
- Be able to build C++ classes using appropriate encapsulation and design principles.

Learning Outcomes:

- Be able to develop, design and implement simple computer programs.
- Understand functions and parameter passing. know how to achieve polymorphism at compile and run time with the concept of function overloading, operator overloading, virtual function
- Understand object-oriented design and programming and also be familiar with the concept of constructor destructor, inheritance
- Understand dynamic memory allocation and pointers.

Course Contents:

Module-I: Introduction to Object-oriented Programming concept
Procedure-oriented Programming, Object-oriented Programming Paradigm; Basic concepts of Object-oriented programming, Benefit of OOPs.

Module-II: Beginning with C++
What is C++? Application of C++, A simple C++ program, An example with class; Structure of C++ program, tokens, keywords, identifiers and constants, data types, reference variables, scope resolution operator.

Module-III: Functions in C++
Main function, function prototyping, call by reference, return by reference, Inline functions and friend functions, virtual function, Concept of Function overloading

Module-IV: Classes and Objects:
Specifying a class, defining member functions; A C++ program with class; Making an Outside Function inline; Static data members; static member functions; arrays of objects; Objects as function arguments.

Module V: Constructors and Destructors:
Constructors, default Constructors; Multiple constructors in a class; parameterized constructor; copy constructor; Destructor

UNIVERSITY OF ENGINEERING AND MANAGEMENT, JAIPUR

Course Description

Module VI: Inheritance:
Defining Derived classes, single inheritance; multilevel inheritance, multiple inheritance; hierarchical, hybrid inheritance; virtual base classes, abstract classes; constructor in derived classes; Making a private member inheritable.

Module VII: Operator overloading:
Defining Operator overloading, rules for overloading operators; Overloading unary operators using member function; Overloading of unary operator with friend function; Overloading Binary operators using member function; Overloading Binary operators using friends, Examples; Type conversion.

Module VIII: Polymorphism:
Concept of polymorphism, runtime polymorphism, compile time polymorphism; Pointers, Pointers to objects; this pointer; Function overloading with an example(Program); Function overriding with a proper example; Virtual function; Pure Virtual function; Abstract class

Module IX: Exception Handling:
Introduction, Basics of Exception Handling; Exception Handling mechanism; Throwing and catching mechanism; Rethrowing an Exception.

Text Books:

- E.Balagurusamy; Object Oriented programming with C++; Tata McGraw Hill Education Pvt. Ltd.
- Schildt, H., the Complete Reference C++, Tata McGraw Hill Education Pvt. Ltd.

References:

- Debasish Jana, C++ object oriented programming paradigm, PHI.
- D. Ravichandran, Programming with C++, Tata McGraw Hill Education Pvt. Ltd.

UNIVERSITY OF ENGINEERING & MANAGEMENT, JAIPUR

Course Description

Title of Course: Software Project Management & Quality
Course Code: BCA403
L-T Scheme: 3-1 **Course Credits: 3**

Introduction:

The dictionary definitions put a clear emphasis on the project being a planned activity. The definition of a project as being planned assumes that to a large extent we can determine how we are going to carry out a task before we start. There may be some projects of an exploratory nature where this might be quite difficult. Planning is in essence thinking carefully about something before you do it – and even in the case of uncertain projects this is worth doing as long as it is accepted that the resulting plans will have provisional and speculative elements.

Objectives:

To develop an understanding of software engineering, software crisis, SDLC. Understanding the concept of software project planning – feasibility analysis, requirement analysis, SRS documents. Come to know the software designing strategies – structured analysis, structured design, DFD, structure chart. Understand concept of Project Management along with software testing, maintenance, back-up..

Learning Outcomes:

Knowledge:

1. Evaluate and analyze the SDLC and basic architecture SRS documents.
2. Help to understand the software design and coding techniques.
3. Understand the software testing principles.
4. Understand the concept project management.
5. Identify various concepts of Advanced UML techniques.

Course Contents:

Unit 1: **Project Management Concept**

Description of software project & software project management, 4P Management Spectrum-Scope & composition.

Unit 2: System Development Life Cycle

Process model, Waterfall Model, Iterative Waterfall Model, Prototyping Model, Evolutionary Model, Spiral Model

Unit 3: Project Scheduling- PERT, CPM. Gantt

Feasibility study, Work Breakdown Structure, Gantt Chart, Critical Path Method, Program Evaluation & Review Technique-Detail study with problem solving

Unit 4: Project Plan

Structure of Project Plan, Project organization, Managerial process, Technical Process

Unit 5: Formal Technical Review

FTR- Software review, role of people, Formal & informal review, classification of software review

Unit 6: Cost estimation and COCOMO Model

Software cost estimation, Measuring software, Function point Metric, Basic-Intermediate-Advanced COCOMO Model, COCOMO II

Unit 6: Software Testing Methodology

Course Description

Testing-Introduction, Testing types- Black box, testing type: White box & its classification, Unit Testing, Testing Type: Integration testing, System testing & its classification, Testing Type: Acceptance, Regression.

Unit 7: Quality Management

Total Quality Management, Quality Assurance, McCall's Quality Factor, Quality standards- ISO 9000, ISO 9001, ISO 90003, ISO 27001, ISO 10002, CMM, Six Sigma

Unit 8: Risk Management

Introduction to Project Risk, Risk Management Process, Risk Assessment, Risk Control

Unit 9: Configuration Management

Software versions, Why configuration management, Configuration Identification, Configuration Control, Configuration Accounting.

Unit 10: Project Management Software

Introduction to Project Management software, tasks, categories, issues, Comparisons of Project Management software's, working with MS Office EPM

Text Books

1. Software Project Management- Bob Hughes & Mike Cotterell
2. Software Project Management and Quality Assurance- Abhishek Bhattacharya, Tanusree Chatterjee

References

1. Software Project Management,Kelkar,PHI
2. Information System Project Mgmt., Schwable, VIKAS

UNIVERSITY OF ENGINEERING & MANAGEMENT,JAIPUR

Course Description

Title of Course: Statistics Numerical Method & Algorithm
Course Code:BM401
L-T Scheme: 3-1 **Course Credits: 4**

Introduction:

The goal of this course is to provide a very common simple intuition enables one to make right decisions and especially show how mathematics is applied to solve basic fundamental problems. The Topics to be covered (tentatively) include:

Approximation in numerical computation, Truncation and rounding errors, Interpolation: Lagrange's interpolation, Newton forward and backward differences interpolation, Newton divided difference. Numerical Integration: Trapezoidal rule, Simpson 1/3 rule, Weddle's rule. Numerical solution of a system of linear equation: Gausse limination method, Matrix inversion, LU factorization method, Gauss-Jacobi method, Gauss Seidel method. Algebraic Equation: Bisection method, Secant method, Regula Falsi method, Newton Raphson method, Method of Iteration. Numerical solution of ordinary differential equation: Taylor's series method, Euler'smethod, Runga–kutta method, predictor–correction method.

Objectives:

The primary goal is to provide engineering majors with a basic knowledge of numerical methods including: root finding, elementary numerical linear algebra, integration, interpolation, solving systems of linear equations, curve fitting, and numerical solution to ordinary differential equations.'C' language and SCILAB is the software environment used for implementation and application of these numerical methods. The numerical techniques learned in this course enable students to work with mathematical models of technology and systems.

Learning Outcomes:

Knowledge:

1. Students would be able to assess the approximation techniques to formulate and apply appropriate strategy to solve real world problems.
2. Be aware of the use of numerical methods in modern scientific computing.
3. Be familiar with finite precision computation.
4. Be familiar with numerical solution of integration, linear equations, ordinary differential equations, interpolations.

Application:

1. An ability to apply knowledge of mathematics, science, and engineering
2. An ability to design and conduct experiments, as well as to analyze and interpret data
3. An ability to design a system, component, or process to meet desired needs within realistic constraints
4. such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability
5. An ability to function on multidisciplinary teams

Books:

1. NumericalMathematicalAnalysis, Sastry,PHI
2. NumericalMathematicalAnalysis (ByJ.B.Scarborough)
3. NumericalAnalysis&Algorithms,PradeepNiyogi,TMH

4. NumericalMathematicalAnalysis, Mathews,PHI
5.Clanguageand NumericalMethods(ByC.Xacier)
6. NumericalAnalysis(ByS.AliMollah)
7. IntroductoryNumericalAnalysis (ByDutta&Jana)
8. NumericalMethods(Problems and Solution) (ByJain,Iyengar&Jain), NewAgeInternational
9. ComputerOriented NumericalMethods,N.Dutta,VIKAS
10. NumericalMethods,Arumugam,Scitech
11. NumericalMethodsin ComputerApplications,P.U.Wayse.EPH.

UNIVERSITY OF ENGINEERING & MANAGEMENT, JAIPUR

Course Description

Title of Course: Basic Environmental Engineering & Ecology
Course Code: HU401
L-T Scheme: 2-1 **Course Credits:2**

Introduction:

This course introduces the basic principles behind the environmental phenomena and how anthropogenic activities are affecting those environmental processes. The different administrative measures taken to safeguard our environment are also discussed in this course. The Topics to be covered (tentatively) include:

- Ecology
- Air pollution and control
- Water Pollution and Control
- Land Pollution
- Noise Pollution
- Environmental Management

Objectives:

In this course we will study about the pattern of growing human population and its effect on the planet. We will be familiarizing with the consequences of anthropogenic activities and measures to mitigate their harmful effects. We will learn about the mechanism behind the global issues like global warming, acid rain, water pollution, etc.

Learning Outcomes:

Knowledge:

1. To introduce the patterns of population growth and associated problems.
2. To familiarize with the cause, effect and control measures of various human made degrading processes.
3. To enable the students to know the mechanism behind the devices to control pollution.
4. To familiarize with administrative laws to mitigate various environmental problems.

Application:

1. To understand the problems associated with pollution
2. To familiarize with the global environmental issues.
3. To understand the principles behind various control devices.
4. To understand and comply with the various government environmental laws.

Course Contents:

Unit 1: Introduction, Ecology, Air pollution and control

Unit 2: Water Pollution and Control

Unit 3: Land Pollution, Noise Pollution

Unit 4: Environmental Management

Text Books

1. Gourkrishna Damohapatra, Basic Environmental Engineering and Elementary Biology, Vikas publishing.

References

1. A.K. De, Environmental Chemistry, New Age International.

.

UNIVERSITY OF ENGINEERING & MANAGEMENT, JAIPUR

Course Description

Title of Course: DatabaseLab(Oracle)
Course Code: BCA491
L-T-P Scheme: 0-0-3 **Course Credits: 2**

Objective:

At the end of the semester, the students should have clearly understood and implemented the following:

1. Stating a database design problem.
2. Preparing ER diagram
3. Finding the data fields to be used in the database.
4. Selecting fields for keys.
5. Normalizing the database including analysis of functional dependencies.
6. Installing and configuring the database server and the front end tools.
7. Designing database and writing applications for manipulation of data for a stand alone and shared database including concepts like concurrency control, transaction roll back, logging, report generation etc.
8. Get acquainted with SQL. In order to achieve the above objectives, it is expected that each students will chose one problem. The implementation shall being with the statement of the objectives to be achieved, preparing ER diagram, designing of database, normalization and finally manipulation of the database including generation of reports, views etc. The problem may first be implemented for a standalone system to be used by a single user. All the above steps may then be followed for development of a database application to be used by multiple users in a client server environment with access control. The application shall NOT use web techniques. One exercise may be assigned on creation of table, manipulation of data and report generation using SQL.

Learning Outcomes:

- Ability to build normalized databases.
- Knowledge of Entity Relationship Modelling.
- Familiarity with SQL, embedded SQL and PLSQL.
- Familiarity with query processing and query optimization techniques.
- Understanding of transaction processing.
- Ability to handle recovery and concurrency issues.
- Familiarity with ODBC, JDBC.

Course Contents:

Exercises that must be done in this course are listed below:

Exercise No.1:ER Model: An entity-relationship model (ERM) is an abstract and conceptual representation of data. Entity-relationship modeling is a database modeling method, used to produce a type of conceptual schema or semantic data model of a system

Exercise No. 2:EER Model: In computer science, the enhanced entity-relationship (EER) model is a high-level or conceptual data model incorporating extensions to the original entity-relationship (ER) model, used in the design of databases. It was developed by a need to reflect more precisely properties and constraints that are found in more complex databases.

Exercise No. 3: Relational Model: The relational model for database management is a database model based on first-order 4predicate logic, first formulated and proposed in 1969 by E.F. Codd. The model uses the concept of a mathematical relation, which looks somewhat like a table of values - as its basic building block, and has its theoretical basis in set theory and first-order predicate logic.

Exercise No. 4:1 NF: First normal form (1NF or Minimal Form) is a normal form used in database normalization. A relational database table that adheres to 1NF is one that meets a certain minimum set

of criteria. These criteria are basically concerned with ensuring that the table is a faithful representation of a relation and that it is free of repeating groups.

Exercise No. 5:2 NF: Second normal form (2NF) is a normal form used in database normalization. 2NF was originally defined by E.F. Codd in 1971. A table that is in first normal form (1NF) must

Exercise No. 6:3 NF: The Third normal form (3NF) is an important form of database normalization. 3NF is said to hold if and only if both of the following conditions hold:

• The relation R (table) is in second normal form (2NF)

• Every non-prime attribute of R is non-transitively dependent (i.e. directly dependent) on every candidate key of R.

Exercise No. 7:BCNF: A relation R is in Boyce-Codd normal form (BCNF) if and only if every determinant is a candidate key. 4The definition of BCNF addresses certain (rather unlikely) situations which 3NF does not handle.

Exercise No. 8:SQL-1: In this lab., we discuss basic SQL operations like creating a table, deleting a table, changing the schema of the table, primary key and foreign key constraints on a table and creating indexes on tables.

Exercise No. 9:SQL-2: Its scope includes efficient data insert, query, update and delete, schema creation and modification, and data access control. In this lab., we discuss SQL operations for populating the tables like inserting into a table, deleting values from a table, and updating the content of the tables.

References

1. “Database Systems: A Practical Approach to design, Implementation and Management”. Thomas Connolly, Carolyn Begg; Third Edition, Pearson Education.
2. "Fundamentals of Database Systems" Elmasri, Navathe, Pearson Education.
3. Bipin C Desai, ?An Introduction to Database Systems?, Galgotia. Publications Pvt Limited, 2001
4. “An Introduction to Database Systems”, C.J.Date, Pearson Education.
5. “A first course in Database Systems”, Jeffrey D. Ullman, Jennifer Windon, Pearson, Education.
6. “Data Management: databases and organization”, Richard T. Watson, Wiley.
7. “DataModeling Essentials”, Graeme C. Simxion, Dreamtech.
8. Introduction to Data Base Management, Naveen Prakash, Tata McGraw Hill
9. “Oracle 10g manuals”.

UNIVERSITY OF ENGINEERING & MANAGEMENT, JAIPUR

Course Description

Title of Course: Computing Lab
Course Code: BM491
L-T-P scheme: 0-0-3 **Course Credit: 2**

Objectives:
The dictionary definitions put a clear emphasis on the project being a planned activity. The definition of a project as being planned assumes that to a large extent we can determine how we are going to carry out a task before we start. There may be some projects of an exploratory nature where this might be quite difficult. Planning is in essence thinking carefully about something before you do it – and even in the case of uncertain projects this is worth doing as long as it is accepted that the resulting plans will have provisional and speculative elements.

Learning Outcomes:
1. Evaluate and analyze the SDLC and basic architecture SRS documents.
2. Help to understand the software design and coding techniques.
3. Understand the software testing principles.
4. Understand the concept project management.
5. Identify various concepts of Advanced UML techniques

Course Contents:
Exercises that must be done in this course are listed below:
Exercise No.1: WAP a program to implement Lagrange interpolation method.
Exercise No. 2: WAP a program to implement Newton's forward interpolation.
Exercise No. 3: WAP a program to implement Runge Kutta method.
Exercise No. 4: WAP a program to implement Euler's method.
Exercise No. 5: WAP a program to implement Taylor series method.
Exercise No. 6: WAP a program to implement Gauss Elimination method.
Exercise No. 7: WAP a program to implement SIMPSON'S 1/3 RULE.
Exercise No. 8: WAP a program to implement Newton's Backward interpolation.
Exercise No. 9: WAP a program to implement Waddle's Rule method .
Exercise No. 10: WAP a program to implement Bisection method.
Exercise No. 11: WAP a program to implement Newtons rapson method.

Text Book:
1. Jain, Mahinder Kumar, Iyengar, S R K , Jain, R K" **Numerical Methods: Problems and Solutions** "

Recommended Systems/Software Requirements:
1. Intel based desktop PC with minimum of 166 MHZ or faster processor with at least 64 MB RAM and 100 MB free disk space.
2. Turbo C or TC3 complier in Windows XP or Linux Operating System.