15CAP101 COMPUTER ORGANIZATION AND ARCHITECTURE 4H - 4C

Instruction Hours / week: L: 3 T: 1 P: 0 C : 4 Marks: Internal: 40 External: 60 Total: 100

End Semester Exam: 3Hours

Scope: The scope of this course is to teach the basic number system with its conversions, digital logic gates, boolean operations and computer architectural concepts with consideration of performance, usability, reliability, and power management.

Objectives: To make students
- understand the fundamental number system with its conversions
- gain knowledge about different codes, digital logic gates and Boolean operations
- acquire basic understanding about the Instruction Set Architecture
- understand the architecture of the basic functional units of the computer such as the input output system, memory systems and secondary storage systems.

UNIT I

UNIT II

UNIT III
Basic Structure of Computers: Functional units - basic operational concepts - bus structures. Machine instructions and Programs: Memory locations and addresses – Memory operations – Instructions and Instruction Sequencing – Addressing modes.

UNIT IV
UNIT V
Basic Processing Unit: Some fundamentals concepts – Execution of a complete Instruction – Multiple-Bus organization – Hardwired control – Microprogrammed control.

TEXT BOOKS

REFERENCES

WEB SITES
1. http://williamstallings.com
3. http://www.vocw.edu.vn/content/m10708/latest
**15CAP102 SOFTWARE ENGINEERING**

**Instruction Hours / week:** L: 3 T: 1 P: 0 C : 4  **Marks:** Internal: 40  External: 60  Total: 100

**End Semester Exam:** 3Hours

**Scope:** The scope of this course is to teach students software theory, principles, tools and processes and to develop and maintain scalable software.

**Objective:** To enhance students with the knowledge of
- the behavior and flow of the software
- planning, designing, developing, validating, and evaluating of software using contemporary practices.
- Software analyzing, coding, and Testing.

**UNIT I**

**UNIT II**

**UNIT III**

**UNIT IV**

**UNIT V**
Software testing techniques: software testing fundamentals – White box testing – basis path testing – control structure testing – Black box testing. Software testing strategies:- Unit testing – Validation testing.
TEXT BOOK

REFERENCES


WEB SITES
Scope: The scope of this course is to teach students the capabilities and limitations of computer operating systems, process management, processor scheduling, deadlocks, memory management, secondary memory management, file management and I/O systems.

Objective: To provide students with
- extensive knowledge of principles and modules of operating systems
- fundamental operating system abstractions such as processes, threads, files, semaphores, IPC abstractions, shared memory regions, etc.
- the principles of concurrency and synchronization, and apply them to write correct concurrent programs/software
- basic resource management techniques (scheduling or time management, space management) and how they can be implemented. These also include issues of performance and fairness objectives, avoiding deadlocks, as well as security and protection.

UNIT I

UNIT II

UNIT III

UNIT IV
UNIT V
Real Time Operating System and Micro Kernels: Characteristics of Real Time Systems-
Micro kernel’s and RTOS-Scheduling for real-time Systems.

TEXT BOOKS
   Prentice Hall of India, New Delhi. [ Unit 1 to 4]
   Prentice Hall of India, New Delhi. [ Unit 5]

REFERENCES
   Prentice Hall India (pvt) Ltd, New Delhi

   India, New Delhi

   Delhi.

WEB SITES
1. www.cs.columbia.edu/~nieh/teaching/e6118_s00/
2. www.clarkson.edu/~jnm/cs644
3. pages.cs.wisc.edu/~remzi/Classes/736/Fall2002/
SCOPE: The scope of this course is to lay down a strong foundation on the C programming including some of the more challenging aspects like pointers, structures, and dynamic memory allocation.

OBJECTIVES: To arm the students with sufficient knowledge of the concepts of C programming language and to make them write efficient C programs.

UNIT I
Introduction to Programming Languages- Introduction to C- Advantage of C- Identifier-Constants-Variable- Data Types- Variable declaration- Assignment operation- Expressions-Operators-Evaluation of Expression- Library functions- Mathematical- Character functions; Formatted and Unformatted Input and Output statements: Scanf(), Printf()- Character based functions.

UNIT II

UNIT III
Arrays: Single, Two and Multidirectional arrays- Strings and related library functions; Functions: Need of function-Types of functions-User-defined functions-Return values and their types-Calling a function-Category of functions-Recursion-functions with arrays-Storage classes.

UNIT IV
Pointers: Pointer Arithmetic- Declaring and initializing pointers- Pointers and arrays-Pointers and Strings- Pointers and functions- Pointers and Multidimensional Arrays; Structures: Declaration- Array of structures- Nested structures- Structures and Functions-Unions-Typedef statement- Enumeration.

UNIT V
Files: Types of Files- Functions related for FILE- Read/Write operations- Character, Word, Line and Structure based Read/Write operations- Random Files. Random access to files-Command line arguments-the preprocessor;

Advanced Topics: Interrupts-BIOS/DOS function handling- Video RAM- TSR Programme.
TEXT BOOKS

REFERENCES

WEB SITES
2. http://www2.its.strath.ac.uk/courses/c/
**Scope:** On successful completion of this course should gain knowledge about the Formal languages Automata Theory, logic and Graph Theory.

**Objective:**
- Understand the fundamental concepts of logic.
- Analyze the four types of normal forms.
- Understand the fundamental concepts of formal languages and graph theory.

**UNIT I**

**UNIT II**

**UNIT III**
Relations and functions: Relations – Properties of relations – Equivalence relations - composition of relations, Closure of relations, Binary relations – Recurrence relations – Order relations – Partial order relations – Partitions – Functions - one-to-one, onto, one-to-one-onto functions – composition of functions, Inverse functions.

**UNIT IV**
Formal languages and Automata: grammars- phrase – structure grammar, context-sensitive grammar, context-free grammar, regular grammar. Finite state automata- Deterministic finite automata and Non deterministic finite automata-conversion of non deterministic finite automata to deterministic finite automata.

**UNIT-V**
**Graph Theory:** Directed and undirected graphs, connected graph, path, reachability, circuits. Matrix representation- adjacency matrix, incidence matrix, path matrix. Trees-binary tree Theorems - statement only (No Proof).

**TEXT BOOK**

**REFERENCE**
The lab sessions will have experiments on the following:

1. **CASE tools**
   Use of diagramming tools for system analysis, such as Turbo analyst, for preparing Data Flow diagrams and E-R diagrams. Use of tools for relational database design such as relational Designer.

2. **Application Development Tools:**
   Use of tools such as Power Builder, Delphi, Magic etc. in developing application software including interactive data-entry screens, transaction processing, report generations, etc.

3. **Management Tools:**
   Use of tools for managing the process of software development such as Source Code Control System (SCCS), Revision Control System (RCS), Make etc.

4. **Program Using Selenium Tool: [web: http://docs.seleniumhq.org]**
   i. Using seleniumhq tool perform single webpage test case
   ii. Using seleniumhq tool perform testing for multiple webpage
1. To write a Linux program to display process deadlock state.
2. To write a program to implement signal handling.
3. Write a Shell program to handle student database with options given below:
   a) Create database. b) View Database. c) Insert a record d) Delete a record.
   e) Modify a record. f) Result of a particular student. g) Exit.
4. To write a simple Linux program using thread.
5. Deadlock avoidance using Banker's Algorithm.
6. To write a program to display the date & time using TCP Sockets.
7. To write a program to display the date & time using UDP Sockets.
8. Simulation of following CPU scheduling algorithms:
   A. FCFS
   B. SJF (preemptive and non-preemptive)
   C. Priority scheduling (preemptive and non-preemptive)
   D. Round Robin Scheduling
9. To write a Linux program to create a lock file.
10. To write a program to display the user information
1. SIN and COS Series
2. Array Operations ( Insert, Delete and Display )
4. Stack
5. Queue.
6. Implement String functions
7. Pointers and Arrays, Pointers and function
8. Recursive function
9. Dynamic Memory Allocation
10. Matrix Operations ( Addition, Subtraction and Multiplication )
11. Linked List Operations.
12. Mark sheet preparation using array of structures
13. Electricity Bill Preparation using Files (Use structures)
14. Implement TWO Dos commands using Command line arguments
15. Design an application using VRAM
16. Display a message every 5 minutes using TSR programming
SCOPE: This course introduces the concepts of Object Oriented Programming language.

OBJECTIVES:
- To prepare object-oriented design for small/medium scale problems
- To demonstrate the differences between traditional imperative design and object-oriented design
- To explain class structures as fundamental, modular building blocks
- To understand the role of inheritance, polymorphism, dynamic binding and generic structures in building reusable code

UNIT I

Control Structures: Pseudocode- if, if/else selection structures – while, for, do while repetition structure- Assignment, Increment, Decrement operators – switch Multiple selection structure – break and continue statements – Logical, Equality, Assignment operators.


UNIT II
Arrays: Declaring Arrays – Passing Arrays to Functions – Sorting Arrays – Linear Search and Binary Search.


Classes and Data Abstraction : Structure Definitions- Class Scope and Accessing Class Members – Access Functions and Utility Functions- Constructors- Destructors.

UNIT III
Classes : Part II : Introduction- friend functions and friend classes- this pointer- new, delete operator- static class members- Data abstraction and Information hiding- Container Classes and Iterators.

Operator Overloading : Fundamentals and restriction of operator overloading –Overloading Stream Insertion, Stream Extraction, Unary, Binary, ++ and – operators. Inheritance : Base Classes and derived Classes – Protected Members- casting base class pointers to derived class pointers- Using member Functions- public, protected, private inheritance – Direct Base classes and Indirect Base classes- Using constructors and Destructors in Derived Classes.
UNIT IV

Templates: Function, Class templates – Overloading template functions- Templates and Inheritance, friends, static members.

UNIT V
Exception Handling: Basics of C++ Exception handling: try, throw, and catch – Throwing, catching, and rethrowing an exception, Exception specifications.

File Processing: Data Hierarchy- files and streams- Creating, Reading, Updating sequential Access files and Random Access files.

The Preprocessor: #include, # define, #error, #pragma preprocessor directives.

Standard C++ Language Additions: Boolean data type- static_cast, const_cast, reinterpret_cast operator, namespaces.

TEXT BOOK

REFERENCES


WEB SITES
2. www.cplusplus.com/
3. www.cppreference.com/
Scope: This course aims to provide students with comprehensive and in-depth knowledge of architecture and functioning of database management systems. To design and build a database system and use computer and database management skills to implement a solution to a business case using database management systems.

Objective: To help students to
- Understand the role and nature of relational database management systems (RDBMS) in IT environment.
- Translate written business requirements into conceptual entity-relationship data models.
- Convert conceptual data models into relational database schemas using the SQL Data Definition Language (DDL).
- Query and manipulate databases using the SQL Data Manipulation Language (DML).

UNIT I

UNIT II
Relational Model: Integrity Constraints over relations- Enforcing integrity constraints- Querying relational data- Logical Database Design- ER to relation- Introduction to views- Destroying & Altering Tables & Views; Relational Algebra Calculus: relational algebra- relational calculus.

UNIT III
SQL Queries Programming: The form of Basic SQL Query- UNION, INTERSECT and EXCEPT- Nested queries- Aggregate operations- Null values Complex integrity constraints in SQL- Triggers & Active databases; Transaction Management Overview: The ACID Properties- Transactions & Schedules- Concurrent Execution of Transactions- Lock Based Concurrency Control- Performance of Locking Transaction support in SQL.

UNIT IV
An Introduction to PL/SQL: PL/SQL Overview- Declaration section- Executable Commands Section – Exception Handling Section; Triggers: Types of Triggers – Trigger Syntax – Enabling & Disabling Triggers – Replacing & Dropping Triggers; Procedures, Functions, and Packages- Required System &
Table privileges – Procedures vs. Functions- Procedures vs. Packages – Create Function syntax – Create Package syntax - Compiling. Replacing. Dropping Procedures, Functions and packages.

UNIT V

TEXT BOOKS

3. Kevin Lonewy. 2014. Oracle Database 11g, the Complete Reference, Tata Mc Graw-Hill, New Delhi. (Unit IV)

REFERENCES


WEB SITES
**15CAP203 VISUAL PROGRAMMING**

<table>
<thead>
<tr>
<th>Instruction Hours / week: L: 4 T: 0 P: 0 C:4</th>
<th>Marks: Internal: 40 External: 60 Total: 100</th>
</tr>
</thead>
<tbody>
<tr>
<td>End Semester Exam: 3 Hours</td>
<td></td>
</tr>
</tbody>
</table>

**Scope:** Scope of this course is for developing applications using Visual Basic and VC++. Students will also be introduced to use the VB IDE, .NET and class libraries to develop Windows desktop applications.

**Objective:** To help students to
- Use forms and controls to create a user interface.
- Create and use Sub and Function procedures, including predefined functions.
- Implement decision structures and loops by using conditional expressions.
- Apply object-oriented programming techniques to create classes, add methods, and add properties.
- Enhance the user interface by adding menus, status bars, and toolbars.
- Access and manipulate data in a Microsoft Access or Microsoft SQL Server database by using Microsoft ADO.
- Build, package, and deploy an application.

**UNIT I**
Introduction-Visual Basic Environment-Element of Visual Basic-Statements and Expressions-Functions-String Functions-Literals-Constants-Variables-Operators-Arrays-Control Arrays-Subroutines and Functions-Looping and decision control structure-If/Then/Else, Select For Next, Do/Loop, While/Wend Structures.

**UNIT II**
Intrinsic Controls: Pointer, Label, Frame, Checkbox, combo Box, Hscroll Bar, Timer, DirListBox, Shape, Image, OLE, Picture Box, List Box, Text Box, Command Button., Option Button, Vscroll Bar, Drive List Box, Line control usage and properties. Microsoft common control: Tree view control-List view control-Active X Control-Adding Control to form-SDI Forms-MDI Forms-Using Forms as Objects-Building class and collections.

**UNIT III**
DDE-Methods, Properties, Events-Database programming: Data Tools-DAO, ADO, OLEDB and RDO Database Connections-Active X Commands.

**UNIT IV**

**UNIT V**
Database: Serialization-Accessing File-ODBC-DAO-MFE Database Classes-Record sets Querying - Internet Application: Using personal web server-Web browser-ISAPI Application-Active X Controls-ATL.
TEXT BOOKS

2. Ivor Horton's, Beginning Visual C++, 1st 2014, Wiley India Pvt. Ltd. (Wrox Beginning Guides), New Delhi, India [Unit 4 & 5]

REFERENCE
Scope: This course is to introduce the fundamentals of computer graphics and a comprehensive introduction of basic input and output peripherals used commonly in computer graphics. This course also intends to teach different algorithms for performing clipping process, scan conversions and hidden surface elimination.

Objective:
- Know fundamentals of computer graphics with different drawing techniques
- Relate 2D & 3D Geometric transformations, Matrices and vectors.
- Understand Scan Conversions, Hidden Surface Elimination
- Discusses about Curves and Surfaces
- Analyze Shading, Colour, Anti-aliasing, Texture Mapping

UNIT I

UNIT II

UNIT III
UNIT IV


UNIT V


TEXT BOOK


REFERENCES


WEB SITES

1. www.microsoft.com/computer graphics/
2. www.en.wikipedia.org/wiki/computer graphics
3. www.w3schools.com/ngws/default.asp
### Scope:
Operations Research is the study of Optimization techniques. It is techniques. It is applied decision theory. This course will give students a comprehensive introduction of Linear Programming Problem. This course also intends to teach Transportation Problems and Assignment Problems. And also about the Inventory control, Replacement models, Queuing models and Network Scheduling.

### Objective:
- Form and solve linear programming problems
- To solve transportation and assignment problems.
- Understand the meaning of queueing and inventory problems.
- Constructing network for the projects.

### UNIT - I
Linear Programming: Formulation of LPP – Graphical solutions to LPP – Simplex method – Big M method - Duality in Linear programming Problem.

### UNIT - II
Transportation model: Mathematical formulation of the problem- Initial Basic Feasible solution - Optimum solution for nondegeneracy and degeneracy model - Unbalanced Transportation problems and Maximization case in Transportation problem

### UNIT - III
Inventory Control: Introduction – Costs involved in inventory, Deterministic models.EOQ models without and with shortage. Buffer stocks and Reorder level – Price Breaks models

### UNIT - IV
Replacement model: Introduction – Replacement of items that deteriorates gradually. Value of money does not change with time –value of money changes with time –Replacement of item that fails suddenly – Individual Replacement – Group replacement
Queuing theory: Introduction – Characteristics of queueing system-Markovian queuing models – Problems of single server with finite / infinite population model.
UNIT - V

TEXT BOOK


REFERENCES


1. Create a class Date whose data members are Day, Month, and Year. Write necessary member functions and perform the following operations using overload operators:
   - Increment a date by a day
2. Compares two dates
3. Create a class String that has a character array as a data member and perform the following operations using overloaded operators:
   - ‘+’ To add two strings
   - ‘+=’ To concatenate one string with the other
   - ‘==’ To compare two strings
4. Create a class Computer and derive two classes Client and Server from it. Have the data members of the classes as follows.
   In the main() program, get the data about n clients and servers and print it back in a neat format.
5. Create four classes with the relationship and data members as shown in the diagram.
   In the main() program, have the facility to:
   - Store the details of n inpatients and outpatients
   - Display the details in a neat format
6. Create a class staff that contains the name, designation, and years of experience of a staff member of a college. Using containership, create two more classes Teaching Staff and NonTeachingStaff according to the following specifications.
   In addition to the properties of the staff class, the Teaching Staff class should contain the highest qualification the staff member possesses and the departments he belongs to. The NonTeachingStaff class needs to contain the properties of Staff only. In the main(), get data about some of the teaching and NonTeachingStaff members of your college and print the details in neat format.
7. Create a class Address as whose data members are Name, Street, City, Pincode and Phone Number of a person. In the main() program, using array of pointers, get addresses of n persons, sort it in alphabetical order of names and display it back in a neat format.
8. Create a class Shape that contains two data members of type double to hold the two dimensions of the shape.
   Derive 3 more classes’ circle, rectangle, and Triangle from the class Shape. Using appropriate member functions, get the values, calculate and print the area of different shapes using dynamic binding.

   Hint:– write 2 member functions in all the derived classes: one to set the data and the other to calculate and display the area.
9. Create two classes British and Metric to store the measurements of distance in the British (feet and inches) and Metric (meters and centimeters) systems respectively.
   In the main () program, perform the following:
   - Get two measurements: one in British and the other in Metric
   - Ask the user in which system (British or Metric) (s) he wants the output.
   - Add two input measurements and print the result according to the user’s choice.

10. Create a class that copies the content of a text file into another file. Write the program in such a way that the program accepts command line arguments and make the program to execute in a way exactly the copy command in Dos works.

11. Create a class Student that could the name, register number and marks in the subjects of the semester. Have the program perform the following operations:
    - Store the data about n students in a data file.
    - Print the mark list of each student whenever requested.
    - Add details about some more students at any time.

12. Sort an integer and a floating –point array using function template.

13. Create a class template stack that accepts a generic data type as a parameter and performs the basic operations of a stack. Using the above class in the main () program, implement stacks for integer and floating point data types separately and perform the operations on the stack.
List of Practical Programs

Some sample applications which may be programmed, are given below

I Create following tables with appropriate constraints.

1. Write SQL statement for following queries.
   a. Age of employees.
   b. Employees whose take home salary is in the range Rs10,000 and 15,000
   c. Employees whose has put 10 years of service.
   d. Employees working under the department head RAGURAMAN
   e. Senior and youngest employee in each department.
   f. Employees who retires after one year.

2. Write cursors
   a. To raise the basic salaries by 25 percent
   b. To retain male employee in the existing employee table and to move female employees to a new table.

3. Create following views
   a. For male employees of age more than 45 years.
   b. For operator with the CHECK OPTION.
   c. For Female employees working under the department head RAJARAMAN.
   d. For male employees with the CHECK OPTION.

4. Create a trigger which fires when one tries
   a. To update records in employee table on Sundays.
   b. To insert records in salary table with basic >30000

5. Generate a pay slip where 40 percent, 20 percent and 15 percent of basic is given as HRA, TA and DA fro the employees.

II Create following tables with appropriate constraints.

1. Take a stock report and list of books in circulation.

2. Write SQL statements for following queries.
   a. Authors of C and C++ books.
   b. List of books issued for UG student
   c. Title of books in computer science department.
   d. Titles of book by the author ‘DIETEL’.
   e. Publishers of C and C++ books.
   f. Number of book issued to each student
   g. Publisher details of Commerce Department.
III Write a cursor
   a. To move books costing more than Rs.1500 into a separate table, Reference table.
   b. To move damaged books into a separate table damaged table.

Some sample PL/SQL Programs

1. Write a recursive program for finding the factorial of a given number.
2. Write a recursive program for finding the first n Fibonacci number.
3. Write a PL/SQL program for multiplication tables 3, 4, 5 and 6.
4. Write a recursive program for finding the reverse of a given number.
5. Write SQL queries to illustrate the string functions a mathematical functions.
6. Write a program for finding the reverse of a given string.
List of Practical Programs

1. Develop a Visual program to implement calculator.
2. Write a Visual basic Application for Banking using Data control
3. Write a visual basic application to implement Employee Pay Roll using ADO Data Control. Use MS Flex grid to display the Employee Details
4. Develop a Visual Basic Application to maintain the books in the library using ADO Data control. Uses Data Report to generate a report for Books availability in the library.
5. Write a Visual basic application to create a ActiveX Control for Calendar.
6. Write a Visual Basic Program to create a Screen saver application for windows,
7. Design a VC++ program to Create a window
8. Implement a Mouse Events
9. Write a VC++ program to create window with menu
10. Write a VC++ program to Store a records in the database for students.
11. Write a VC++ program to implement a mouse event using SDK
12. Write a VC++ program to transfer the file from source to destination using MFC.
SEMESTER-III

15CAP301 PROGRAMMING IN JAVA 4H -4C

Instruction Hours / week: L: 4 T: 0 P: 0 C:4

Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3Hours

**SCOPE:** The scope of this course is to provide an introduction to the object-oriented programming paradigm in Java.

**OBJECTIVES:** To expose the students to the best object oriented programming paradigm, java and strengthen their OOP’s fundamental knowledge.

**UNIT I**

**UNIT II**

**UNIT III**
UNIT IV

UNIT V

TEXT BOOK

REFERENCES


WEBSITES
1. java.sun.com/docs/books/tutorial/
2. www.en.wikipedia.org/wiki/Java
3. www.java.net/
SEMESTER-III

15CAP302 COMPUTER NETWORKS 4H - 4C
Instruction Hours / week: L: 3 T: 1 P: 0 C:4 Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3Hours

Scope: The scope of this course is to provide students the fundamentals of data communications networks, working knowledge of data transmission concepts, the various operation of all seven layers of OSI Model and the protocols used in each layer.

Objective: To make students to
- become familiar with layered communication architectures (OSI and TCP/IP).
- understand the client/server model and key application layer protocols.
- learn sockets programming and how to implement client/server programs.
- Understand the concepts of reliable data transfer and how TCP implements these concepts.
- know the principles of congestion control and trade-offs in fairness and efficiency.
- learn the principles of routing and the semantics and syntax of IP.
- understand the basics of error detection including parity, checksums, and CRC.

UNIT I

UNIT II

UNIT III
Network Layer: Network Layer Design Issue-Routing Algorithms-Congestion control Algorithms-Quality of service-Inter networkinging-Network Layer in Internet

UNIT IV
Transport Layer: Transport service-Elements of transport protocols-Internet Transport protocols: UDP and TCP-Performance Issues

UNIT V
TEXT BOOK

REFERENCE BOOKS


WEB SITES
2. www.amazon.com/Computer-Networks-Andrew-S-Tanenbaum/dp/0133499456
S Scope: To equip students with adequate skills to conceptualize, design and develop well engineered web applications.

O bjective: To make students to

- Understand the fundamental features of web applications.
- Understand the objects and components needed for a web designing.
- To identity which technologies can be used.
- Create a webpage

U nit I

U nit II
JavaScript: Introduction to javascript – Programming fundamentals – Functions and objects – Navigator object model

U nit III
JavaScript: Form and form elements – Scripting frames and multiple windows – Event object – Functions and custom objects.

U nit IV

U nit V
TEXT BOOKS

2. Jeremy McPeak, 2015, Beginning Javascript, 5th Edition, Wiley India Pvt. Ltd. (Wrox Beginning Guides), New Delhi, India (Unit II)


REFERENCES


WEB SITES
- www.w3schools.com/
- www.javascriptkit.com
- www.aspfree.com
- www.aspnettutorials.com
Scope:
The scope of this course is to introduce the concepts, techniques and applications of software agents. By the end of the course the students are expected to evaluate current software agent systems.

Objective: To help students to
- Understand the basic concepts, techniques and applications of software agents.
- Understand how agent communications between them.
- Know the applications of agents.
- Understand software agents design tools.

Unit I

Unit II
Deductive Reasoning Agents – Agents as Theorem Provers – Agent-Oriented Programming – Concurrent Mutate.
Reactive and Hybrid Agents- Brooks and the Subsumption Architecture – The Limitations of Reactive Agents – Hybrid Agents

Unit III
Reaching Agreements – Mechanism Design – Auctions – Negotiation – Argumentation Communication – Speech Acts – Agent Communication Languages – Ontologies for Agent Communication – Coordination Languages

Unit IV
Cooperative Distributed Problem Solving – Task Sharing – Combining Task and Result Sharing – Handling Inconsistency – Coordination – Multiagent planning and Synchronization

Unit V
Methodologies – Agent-Oriented Analysis and Design Techniques – Pitfalls of Agent Development – Mobile Agents-Applications of Agents
TEXT BOOK

REFERENCES

Scope:

This is an intensive course that explores Client-Server system architecture. In this course, students will learn how the systems work, how they break, and how to rapidly test them as thoroughly as possible.

Objectives: To helps students

- To Understand basics client/server architecture
- To Establish and use Database connectivity
- To gain the knowledge of CORBA

Unit I


Unit II

Clients, servers and operating systems- NOS- Creating the single system Image- RPC, messaging and peer-to-peer. Client/server systems development- Software- Client/server systems development- Hardware.

Unit III

SQL database servers- SQL middleware and federated databases- Data warehouse- OLAP and multidimensional data- Client/server transaction processing- The magic of transaction- TP monitors.

Unit IV

Database connectivity solutions: ODBC – The need for Database Connectivity- Design overview of ODBC- Architecture- Components- Applications- Driver Manager- Drivers- Data sources- ODBC 2.5 and ODBC 3.0.

Unit V


TEXT BOOK

REFERENCE
Scope:
Scope is to introduce software architectures and train students to perform the analyses necessary to formulate effective software architectures. They able to analyze software.

Objective: To make students

- The Software perspective architectural and how it differs from lower-level design.
- Understand the need for a Software Architecture.
- To exploit the software architecture environment.

Unit I

Unit II
Case studies - Key word is Context – Instrumentation Software – Mobile Robotics – Cruise Control – Three Vignettes in Mixed Style

Unit III

Unit IV

Unit V

TEXT BOOKS

REFERENCE:
SEMESTER-III

15CAP304W             MULTIMEDIA SYSTEMS             4H - 4C

Instruction Hours / week: L: 4 T: 0 P: 0 C: 4 Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3Hours

Scope: The scope of the course is to teach students interface design, content design and creation with graphics, animation, audio and video materials.

Objective: To make students
  - Basic multimedia concepts.
  - Basic knowledge of 2D animation
  - The knowledge of audio and video capture and editing tools

UNIT I

UNIT II

UNIT III
Storage Media & Access – Data Compression Techniques: Some basic compression techniques, JPEG, MPEG, DVI

UNIT IV
Multimedia Database- Content-Based Storage and Retrieval- Designing a Basic Multimedia Database- Image Color Features- Image Texture Features- Image-Shape Features- Classification of Data- Artificial Neural Networks- Semantics in Multimedia Data

UNIT V
TEXT BOOKS:


REFERENCE BOOKS:


WEB SITES
1. en.wikipedia.org/wiki/Multimedia
2. www.arenamultimedia.com/
3. www.nextwavemultimedia.com/
Scope: The scope of the course is to teach students gain knowledge on mobile communication systems and depth knowledge of mobile operating systems.

Objective:
- Understand basic communication systems.
- Acquire basic knowledge MAC Layer
- Wide knowledge about linux based operating systems
- Design different application on mobile based operating systems.

UNIT I

UNIT II

UNIT III
Linux Based Operating Systems: Access Linux-Android-DSLinunx- IPod Linux-Limo (Linux Mobile)-MeeGo-Mobilinux-OpenMoko-SHR-QT Extended- Ubuntu

UNIT IV

UNIT V
Symbian programming-Installing the Qt Software -Create a Simple Mobile Application. Windows Phone 7 Programming-Installing Visual Studio 2010- Installing Windows Phone 7 SDK- Create a Simple Mobile Application. Android Prrogramming: Installing JDK -NetBeans Installation-Create a Simple Mobile Application
TEXT BOOKS

REFERENCE BOOKS

WEB SITES:
- en.wikipedia.org/wiki/Mobile_operating_system
- www.shoutmeloud.com/top-mobile-os-overview.html
Scope: To provide a theoretical framework for considering corporate finance problems and issues and to apply these concepts in practice.

Objective:
- To give everybody a base level of finance knowledge that an MPA from a top business school should possess.
- To give everybody the ability and confidence to tackle common financial problems in practice.
- To provide adequate preparation for future managers should organize their financial transactions effectively and with integrity.

UNIT I

UNIT II
Ratio Analysis: Meaning - Advantages - Limitations; Classification of Ratio: Profitability, Turnover and Solvency Ratios.

UNIT III

UNIT IV
Budget and Budgetary Control: Meaning and Definition - Objectives of Budgetary Control - Advantages and Limitation Preparation of Different Types of Budgets.

UNIT V
TEXT BOOKS
1. N.Vinayakam, Mani and Nagarajan, “Principles of Accountancy”.
2. S.N.Maheswari, “Principles of Management Accounting”.
3. S.P.Jain and Narang, “Cost Accounting”.

REFERENCE BOOKS
1. Sharma and Sasi.K.Gupta, “Management Accounting”.
2. T.S.Grewal, “Introduction to Accountancy”.
3. Ramachandran and Srinivasan, “Management Accounting”.

List of Practical Programs

1. Create an employee package to maintain the information about the employee. Use constructors to initialize the employee number and use overloading method to set the basic pay of the employee. By using this package, create a Java program.

2. Create a set of classes with the relationship as shown in the diagram and use inheritance relationship to define the sub classes.
   Get and display the details of some of the two and four wheelers and display them in a neat format.

Create a frame with user specific size and position it at user specific position (use command line argument). Then different shapes with different colors (use menus).

3. Java program to handle different mouse events.
4. Create an applet for a calculator application.
5. Java program to maintain the student information in text file.
6. Animate images at different intervals by using multi-threading concepts.
7. Program to send a text message to another system and receive the text message from the system (use socket programming)
8. Java program by using JDBC concepts to access a database.
9. Java program by using to implement the tree viewer.
10. Implementation of Binary Search Tree
1. Write a networking program in Java to implement a TCP server that provides services for a TCP Client.

2. Write a networking program to implement socket programming using User datagram Protocol in Java.

3. Implement an FTP server using socket programming.

4. Implement a chat server using socket programming.

5. Implement an ECHO server using socket programming.


7. Implement Ping server and Ping client using socket programming.

8. Using UDP to transfer a text file from one host to another.


10. Simulate simple Web Browser.

11. Write a Java program to check whether the given DNS is found in the internet or not.

12. Write a network program using HTTP to print the document for the given URL.
1. Develop a website for Karpagam University using HTML.
2. Write Online Quiz program (Include Style Sheets)
3. Create a simple animation using DHTML.
4. Write a program to apply Mask into an Image Using Filters in DHTML.
5. Generate web page that represents clock-every 60 see the page updated with server current time Using JavaScript.
6. Design a form and validate it using JavaScript.
7. Show the demo of AD Rotator Component
8. Write Database Access program using ASP.
9. Program to retrieve Cookies information using ASP
10. Program to count web page hits using ASP
11. Program to create Date & Time, String Manipulation using ASP
12. Write a program to find the visitor’s Browser Type, IP Address and More Information.
Scope: The scope of this course is to provide an introduction to basic techniques for designing efficient algorithms, analyzing their complexity and applying these algorithms to broad range of applications.

Objective:
- Be able to analyze algorithms
- Understand the fundamental skills in back tracking logic, dynamic programming.
- Understand the fundamental algorithms such as searching, and sorting

UNIT I
INTRODUCTION: Algorithm - pseudo code for expressing algorithms – analysis - time complexity and space complexity - efficiency of algorithms - O-notation - Omega notation and Theta notation.

UNIT II

UNIT III
DYNAMIC PROGRAMMING: General method - Multistage Graphs – All pairs shortest paths, Single source shortest paths - optimal binary search trees - O/1 Knapsack problem - Traveling sales person problem.

UNIT IV

UNIT V
BRANCH AND BOUND: Least Cost(LC) search, Bounding - LC branch and bound - FIFO branch and bound - Travelling sales person problem.

TEXT BOOK

REFERENCE BOOKS

Scope: This is designed to rapidly learn Java web programming with J2EE. To understand object-oriented programming with J2EE and learn how to write increasingly sophisticated J2EE programs and to get started fast in J2EE programming.

Objective:
- Understand J2EE as an architecture and platform for building and deploying web-based, n-tier, transactional, component-based enterprise applications
- Understand the fundamental concepts of XML and related technologies
- Acquire knowledge on how XML is currently being used in various application areas
- Know how to parse and transform XML documents via tools and through programming APIs
- Understand the EJB architecture and have a good grasp on when to use and how to use various EJB bean types and acquire relevant Java programming experience

UNIT I

UNIT II

UNIT III
Java Servlets: Benefits – Anatomy – Reading Data from Client –Reading HTTP Request Headers – Sending Data to client – Working with Cookies.

UNIT IV

UNIT V
Java Server Pages – Java Remote Method Invocation.

TEXT BOOK
REFERENCE BOOKS

WEBSITES
1. java.sun.com/javaee/
2. java.sun.com/j2ee/1.4/docs/tutorial/doc/
3. www.j2eebrain.com/
**15CAP403 .NET FRAMEWORK 4H -3C**

Instruction Hours / week: L: 4 T: 0 P: 0 C : 3  
Marks: Internal: 40 External: 60 Total: 100  
End Semester Exam: 3 Hours

**Scope:** This course provides functional understanding of the components of the .NET Framework and the common language runtime (CLR) and know how COM components. Also provides better knowledge of Architecture and Basic Concepts required for building database application with ADO.NET Programming. It also provides an platform to apply all the window controls.

**Objective:**
- Become a good .NET programmer.
- Know how COM components and the .NET Framework interoperate with each other.
- Identify and use the classes and namespaces in the .NET Framework class library.
- Build WEB Applications using Microsoft ASP.NET programming.

**UNIT I**

**UNIT II**
Basic Windows Controls: Textbox Control- ListBox, CheckedListBox-Scrollbar and TrackBar Controls-More Windows Control-The common Dialog Controls-The Rich TextBox Control - Handling Strings, characters and Dates. The TreeView and ListView Controls: Examining the Advanced Controls-The TreeView Control-The ListView Control

**UNIT III**
The Multiple Document Interface-Databases: Architecture and Basic Concepts-Building Database Application with ADO.NET-Programming with ADO.NET

**UNIT IV**
Goal of ASP.NET –ASP.NET Web Server Control-Validation Server Controls-Themes and Skins -Content Page Holder

**UNIT V**
Data Binding in ASP.Net: Data source Controls – Configuring data source control caching – storing connection information-Using Bound list controls with Data Source Controls – Other Databound Controls-Data Management with ADO.Net
TEXT BOOKS

REFERENCE BOOKS

WEB SITES
1. www.microsoft.com/NET/
3. www.w3schools.com/ngws/default.asp
4. www.vbtutot.com
**Scope:** The scope of this course is to teach students how to perform day-to-day database administrative tasks
- to keep database operational
- basic trouble shooting
- monitoring performance of tasks

**Objective:**
- install and configure database
- create users and assign roles
- Optimize schemas, tables, indexes and views
- Manage database services and clients
- Will be able to move the data from one database to another database.
- Take backup and perform recovery.

**UNIT I**

**UNIT II**

**UNIT III**
UNIT IV

UNIT-V
Creating Tablespaces in a VLDB Environment: Bigfile Tablespace Basics – Creating and Modifying Bigfile Tablespace – Bigfile Tablespace ROWID format – DBMS_ROWID and Bigfile Tablespaces - Advanced Oracle Table Types – Using Bitmap Indexes – Oracle Data Pump
Remote queries – Remote Data Manipulation: Two Phase Commit – Managing Distributed Data – Managing Distributed Transactions – Monitoring and Tuning Distributed Database

TEXT BOOK

REFERENCE BOOK

WEB SITES:
- www.oracle.com/technology/software/products/database/oracle10g/index.html
- www.oracle-base.com/articles/10g/
- www.adp-gmbh.ch/ora/misc/10g.html
Scope: To make student understand the goals, issues, technologies, algorithms, protocols and design criteria used in cryptography and data security and solution.

Objective:

- To teach fundamental aspects of security in a modern networked environment with the focus on system design aspects and cryptography in the specific context of network.
- To build protection mechanisms in order to secure computer networks.
- Write coding to encrypt “Plain Text” into “Cipher Text” and vice versa, using different encryption algorithms.
- The ability to choose a suitable ciphering algorithm according to the required security level.
- Build cryptosystems by applying encryption algorithms,
- Build secure authentication systems by use of message authentication techniques.

UNIT I

UNIT II

UNIT III

UNIT IV

UNIT V
TEXT BOOK

REFERENCE BOOKS

WEB SITES
1. williamstallings.com/Crypto3e.html
2. u.cs.biu.ac.il/~herzbea/book.html
3. www.flipkart.com/search-books/cryptography+and+network+security+William+stallings+ebook
Scope: Software testing is an investigation conducted to provide stakeholders with information about the quality of the product or service under test. Test techniques include, but are not limited to, the process of executing a program or application with the intent of finding software bugs.

Objective:
- To perform test each time they are run, thereby eliminating human error.
- To test how the software reacts under repeated execution of the same operations.
- To program sophisticated tests which bring out hidden information from the application.
- To reuse tests on different versions of an application, even if the user interfaces changes.

UNIT I
Introduction: Purpose of Testing – Dichotomies - Model for Testing – consequences of bugs-
Taxonomy for bugs

UNIT II
Flow/Graphs and Path Testing : Path testing basics-predicates, path predicates and
Achievable paths – Path sensitizing– path instrumentation-implementation and application of
path testing

UNIT III
Transaction flow testing: – Transaction flow - Transaction flow testing techniques –
implementation comments – testability tips -Data flow testing basics- Data flow testing
strategies.

UNIT IV
Domains and paths-Nice Domains and Ugly Domains-Domain testing-Domains and interface
testing-Domains and testability-Metrics-Linguistic metrics-structural metrics-Hybrid metrics-
metrics implementations-Testability tips

UNIT V
Risk Analysis : Benefits of Risk analysis – Project Management Strategies and Risk – MITs
risk Analysis – MITs Ranking Criteria - Using Risk Ranking in Forensics –Test estimation
process-MIT totals worksheet-Sizing worksheet
TEXT BOOKS

REFERENCE BOOKS

WEB SITES
1. http://my.safaribooksonline.com
2. http://www2.sas.com
4. www.cs.cmu.edu
5. www.softwaretestingmanagement.com
Scope:
XML is a universal standard for information markup that is revolutionizing the way of handling data. This course will help students to learn how to create websites using XML, understand the essentials of the XML standards and appreciate some of the performance characteristics, gain some practical experience and insight of using XML.

Objective: To help students to

- Be aware of a range of XML tools (Many of them are free).
- Know how to set out an XML document
- Define custom markup language
- Understand the purpose of using DTDs and Schemas to validate XML
- Use XSLT to write a style sheet for XML document to produce multiple output
- Combine XML with existing web technologies.

UNIT I

UNIT II

UNIT III

UNIT IV
XML Path Language: Nodes – Location Paths; XSLT: Templates - Creating Elements and attributes – Iteration and Sorting – Conditional Processing – Copying Nodes – Combining style sheets – variables. XLink, XPointer, XInclude and XBase.
UNIT V

TEXT BOOK

REFERENCE BOOKS

WEB SITES
1. en.wikipedia.org/wiki/XML
2. www.w3.org/XML/
3. www.w3schools.com/xml/default.asp
Scope: Learn how to handle mobility efficiently as well as different functional aspects of wireless communication, including security, privacy, file systems, resource discovery, resource management, personal on-line identities, and other areas.

Objective:
- Understand general concepts of transmission and wireless communication.
- Understand how mobility affects all layers of the Internet protocol stack, from the MAC layer up through the application layer.
- Understand the emerging applications enabled by wireless networks, and ad hoc mobile wireless networks.
- Work in groups to design and implement wireless applications.

Unit I
Mobile Data Introduction: The Rise of Mobile Data-Key Services for the Mobile Internet-Overview of the WAP-The origins of the WAP- WAP architecture-WAP Internal Structure-Components of the WAP Standards- WAP Gateways-Network Infrastructure Services Supporting WAP Clients-WAP Architecture Design Principles –Relationship with other standards.

Unit II

Unit III
Variables –Other Content you can include-Controls-Miscellaneous Markup- Sending Information-Application Security-Other Data; The Meta element- Document Type Declarations- Errors and browsers Limitations-Content generation- WML Version Negotiation.

Unit IV
User Interface Design: Making Wireless Applications, Easy to Use- Website Design-Computer Terminals Vs Mobile Terminals-Designing a usable WAP site-structured usability method-user interface design guidelines- Design guidelines for selected WML Elements.

Unit V

Text Book

References

Web Sites:
2. www.wap.com
3. www.w3schools.com/wap/
Scope: The Scope of this course is to expose the students to the architecture, design, and implementation of massive-scale data systems. The course also discusses foundational concepts of distributed database theory including design and architecture, security, integrity, query processing and optimization, transaction management, concurrency control, and fault tolerance.

Objectives: To impart necessary skills to students
- To design good performing distributed database schemas.
- To create optimized query execution plan.
- To efficiently distribute and manage the data.
- To manage distributed access control
- To know how to make secure the databases.

UNIT - I
Database concepts: Data Models- Database Operations- Database Management- DB Clients, Servers, and Environments. DBE Architecture: Services- Components and Subsystems- Sites - Expected Services- Expected Subsystems- Typical DBMS Services- DBE Taxonomy: COS Distribution and Deployment- COS Closedness or Openness- Schema and Data Visibility- Schema and Data Control.

UNIT - II

UNIT - III

UNIT - IV

UNIT - V
TEXT BOOK

REFERENCES

WEB SITES
1. en.wikipedia.org/wiki/Distributed_computing
Scope:

The goal is to understand the basics of the layered TCP/IP stack architecture for IPv4 and the key protocols in the TCP/IP suite including Internet Protocol version 4 (IPv4), Internet Control Message Protocol (ICMP), Transmission Control Protocol (TCP), and User Datagram Protocol (UDP). Knowledge of IPv6 address auto configuration works for stateless address auto configuration.

Objective: To educate students

- Learn about IPv4 forwarding and routing.
- Learn about host name resolution and the Domain Name System (DNS).
- Learn about IPv6 addresses.
- know Architectural Overview of the TCP/IP Protocol Suite

UNIT I

UNIT II

UNIT III

UNIT IV

UNIT V
TEXT BOOKS

REFERENCE BOOKS

WEB SITES
1. en.wikipedia.org/wiki/Internet_protocol_suite
3. www.yale.edu/pclt/COMM/TCPIP.HTM
4. www.w3schools.com/tcpip/default.asp
Scope: Introducing students to the concepts and terms used in the object-oriented approach to systems analysis and design. Highlighting the importance of Object Oriented Analysis and Design concepts and apply them to solve problems. Showing how we apply the process of Object Oriented Analysis and Design documents for a given problem using Unified Modelling Language. Pointing out the importance and function of each UML model throughout the process of object-oriented analysis and design.

Objective:
- Use object-oriented technologies
- Use Unified Modeling Language 2.2
- Perform object-oriented analysis and design
- Explain how the Unified Modeling Language (UML) represents an object-oriented system using a number of modeling views.
- Construct various UML models (including use case diagrams, class diagrams, interaction diagrams, state chart diagrams, activity diagrams, and implementation diagrams) using the appropriate notation.

UNIT-I
Complexity: The structure of complex systems-the inherent complexity of software-The five attributes of a complex system-Organized and Disorganized complexity-Bringing order to Chaos-On Designing complex systems. The Object model: The evolution of the object model-Foundations of the object model-Elements of the object model-Applying the object model.

UNIT II
Classes and Objects: The nature of an object-Relationship among objects-The nature of a class-Relationships among classes-The interplay of classes and objects-On building quality classes and objects. Classification: The importance of proper classification-Identifying classes and objects-Key abstraction and mechanisms.

UNIT III

UNIT IV

UNIT V
TEXT BOOK

REFERENCES

WEB SITES
1. uml-tutorials.trireme.com/
Scope: This course will give students an overview of the web services architectures, and then learn the standard APIs for SOAP messaging and WSDL-driven, component-based service development.

Objective:
- Be able to describe the interoperable web services architecture, including the roles of SOAP and WSDL.
- Use lower-level SOAP and XML APIs for services and/or clients.
- Build and Host Web Services.

UNIT I
Introduction: What are Web Services – Importance of web services – Web services and enterprises; XML Fundamentals:: XML Documents - Namespaces – Schema – Processing XML.

UNIT II

UNIT III

UNIT IV
UNIT V

TEXT BOOK

REFERENCE BOOKS

WEBSITES
1. www.w3schools.com/webservices/default.asp
2. en.wikipedia.org/wiki/Web_service
3. www.webservices.org/
Scope: To teach students the various fundamental layers in ad-hoc network, the issues pertaining to major obstacles in establishment and efficient, management of ad-hoc and sensor networks and applications of ad-hoc and sensor networks.

Objective: To make the students
- To understand sensor network architectures
- Select appropriate routing algorithms for different network environments
- The various security practices and protocols of Ad-hoc and Sensor Networks.
- Deploy security mechanisms in the wireless ad-hoc and sensor networks

UNIT I

UNIT II

UNIT III

UNIT IV
UNIT V

TEXT BOOKS

REFERENCE BOOKS

WEB SITES
• en.wikipedia.org/wiki/Wireless_ad_hoc_network
• searchmobilecomputing.techtarget.com/definition/ad-hoc-network
• www.cs.jhu.edu/~cs647/intro_adhoc.pdf
1. Create a sign in form in servlets.
2. Write a servlet Program to lock a server.
3. Write a servlet program that returns list of information in table format.
4. Design a counter that counts number of times user has visited the site in current browsing session.
5. Write a program to retrieve cookies information
6. Build a JAVA Bean for opening an applet from JAR file.
7. Write a program to add controls in BEAN.
8. Design a counter in JAVA BEAN.
9. Write a program to stream contents of a file using JSP.
10. Write a program to insert a menu applet into JSP page.
VB.Net
1. Write a Program to perform various string manipulation functions.
2. Using windows application form, Develop a program to create a form, place controls and manipulate data.
3. Write a program to create inventory control using class library.
4. Write a program to create Web Services Using Vb.Net.
5. Write a program to create a screen saver using controls
6. Create an ActiveX program with simple example.
7. Using windows Application: Design Employee Details, use Sql Server as back end and also use checked list box.

Asp.Net
1. Write a program to create an on-line quiz using content page holder.
2. Write a program to retrieve Cookies information.
3. Write a program to count web page hits.
4. Write program to retrieve environment variables browser capability information.
5. Write a program for database connectivity to retrieve student information
<table>
<thead>
<tr>
<th>15CAP413D</th>
<th>DBA LAB</th>
<th>4H - 2C</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Instruction Hours / week:</strong> L: 0 T: 0 P: 4 C: 2</td>
<td><strong>Marks:</strong> Internal: 40 External: 60 Total: 100</td>
<td></td>
</tr>
<tr>
<td><strong>End Semester Exam:</strong> 3 Hours</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1. Demo for Globalization Support
2. Setup Listener Security
3. Configuring Recovery Manager
4. Write a program Using Recovery Manager
5. Write a program for Managing Diagnostic Sources
6. Implement Database Recovery
7. Demo for Flashback Database
8. Implement User Error Recovery
9. Write a program for Dealing with Corruption
10. Show the demo for Automated Management
11. Creating a database and do the manipulation.
12. Managing index tables
1. Write a program to convert your college name from plain text to cipher text using Transposition cipher method of encryption.

2. Write a program to convert your name from plain text to cipher text using the One Time Pads method of encryption.

3. Write a program to encrypt a paragraph using the Data Encryption Standard Algorithm.

4. Write a program to encrypt your biodata using the Advanced Encryption Standard Algorithm.

5. Write a program to decrypt the “Network Security” theory syllabus using the RSA Algorithm.

6. Write a program that takes a binary file as input and performs bit stuffing and Cyclic Redundancy Check Computation.

7. Write a program to Simulate the working of Sliding-Window protocol.

8. Write a program to find the shortest path in a network using Dijkstra's Algorithm.

9. Write a program to implement the Token Bucket Algorithm for Congestion Control.

10. Write a program for the following chat application:

    One to One : Open a Socket connection and display what is written by one to another.

    Many to Many : Each Client Opens a Socket connection to the client server and writes to the socket. Whatever is written by one can be seen by all. Implement symmetric key cryptography.
SOFTWARE TESTING

1. Create a VB form with the following fields and create the database also for them. Insert 3 records. Using Win Runner tool record the above 3 transaction and test them and produce the Report. (Blackbox Testing).

2. Create a VB form and then add login dialog form. Using Win Runner tool check the Username and Password and produce the Report. (Security testing).

3. Create a VB form with the following fields and check the calculation is correct or not by using the test toll Win Runner. (Functional Testing) Fields – Name, Designation,Department,Basic,HRA,DA,PF and netsal.

4. using Win Runner test tool check the database values after changing. Using Flight database. (Regression testing).

5. Write a C program for Boundary Testing.

6. Write a C program for Loop Testing.

7. Write a C program for Integration Testing.

8. Write a C program for Interface Testing.

9. Write a C program for Unit testing.

Software Quality Assurance

1. To develop a banking application, perform the requirement analysis and give a quality status report.
2. Perform the system testing to develop a electricity application and give a quality status report.
3. Perform the report design and give a quality status report.
4. Develop a library management system and give a quality status report.
5. Develop a hospital management system and a quality status report and give a comparison performance report in Linux and windows operating system.
1. Create a menu in XML.
2. Create a demo for XSLT.
3. Display XML information in Tree structure format.
4. Integrate XML in Web Applications
5. Write a program to navigate the records in the file.
6. Write a database access with XML.
7. Write a program to save data to an XML file.
8. Write a program to show the function of CDATA.
9. Write a program to generate XML file on the server.
10. Write a program to generate XML file from the Database
11. Write a program to load a text file into a div element with XML HTTP.
12. List data from an XML file with XML HTTP.
1. Demonstrate the creation of output files for Xgraph.
2. Illustrate NS2 script to send data between two nodes.
3. Create a simple simulation topology to generate TCP and UDP traffic
4. Simulate a NS2 scenario to handle link failures.
5. Write a NS2 script to handle Multicast traffic.
6. Demonstrate OSPF routing.
7. Create a simulation topology and analyze QOS parameters.
8. Demonstrate the creation of wireless topology.
SCOPE: Web design focuses on the structure of the website including the information architecture, the layout or the pages and the conceptual design with branding.

OBJECTIVE:
- Understand the fundamentals of HTML
- Use of different formatting options
- Creation of tables and frames
- Relate with DHTML and CSS


UNIT-II: Tables: Creating Table – Dividing Table into Columns – Dividing Table into Rows Creating Headers- Adding Border – Putting a Background Image – Heading across two or more columns – Changing color of the cell-aligning the content – Display of Tables. Frames- Forms: Working with forms- Creating forms – working with menus – working with Radio buttons – check boxes- textboxes- text areas – password boxes- submit button- Resetting the form.


UNIT-V

TEXT BOOK

REFERENCES

WEB SITES
- www.w3schools.com/
- alexle.net/archives/category/web-technology
Scope: To teach to the students the current practices of cloud computing, cloud computing models, techniques, and architectures

Objective:
- Identify key elements of the cloud computing
- Understand and appreciate the need for cloud computing, and identify their use in industrial applications
- Analyse the current issues in cloud computing
- Develop an in-depth understanding of selected parts of the material

UNIT I

UNIT II

UNIT III
Infrastructure as a Service (IaaS) - Platform as a Service (PaaS) - Software as a Service (SaaS) - Identity as a Service (IDaaS) - Compliance as a Service (CaaS) - Cloud storage.

UNIT IV
UNIT V
Case Study on Open Source and Commercial Clouds: Microsoft Azure- Amazon EC2- Google Web services.

TEXT BOOK

REFERENCE BOOK

WEB SITES
1. en.wikipedia.org/wiki/Cloud_computing
Scope: PHP course is designed to learn how to do client-side programming, which will run on either a LAMP or a Windows web server.

Objective: To help students to
- Get hands-on experience in scripting, debugging, testing.
- Establish a working environment for PHP web page development
- Use variables, constants, and environment variables in a PHP program
- Learn to create dynamic interactive pages with PHP.
- Learn to manipulate files with PHP.
- Understand how MySQL works.
- Learn to use SQL to output reports with MySQL

UNIT I

HTML Primer: The HTML Document type definition- The Form and Input Elements; Accessing PHP and HTTP Data: Predefined Variables- Variables in HTTP Request and Response- Super Global Arrays; Links; Query Strings; HTML(Web) Forms; HTML Form Elements-HTML Form Fields(Controls) and PHP; The Concept Of State: State Maintenance- Native Sessions in PHP.


UNIT II
Testing and Debugging: Values that break your code- Basic error types; Debugging PHP Script: Understanding PHP error Massages- Syntax Errors- Logic Errors-Runtime Errors; Debugging and Handling Errors in PHP5: Preventing the display of private information-

Roll your-Own Debugging tools; Form Validation: Using the Exit statement- string validation and regular expressions- validating data entry- using reg exps to check file path parameters; Handling Errors: Gracefully- Configuring PHP for error handling- Try/Catch New in PHP5.
Development planning: Formal software Development processes – optimizing your code-
Using Coding standard; Writing user-defined functions in PHP: The Structure of Functions-
Switching Functions – How Values Get Inside functions; Scope of variables: Global and
Local Variables; Creating Static Function Variable-Nesting-Recursion-The Include and
Require Statements-Things to be careful about with include and require.

UNIT III
Files and Directories: Files and Directory Handling- Working with Files- Opening and
Closing files-Getting Information about a file-reading and writing to files-Reading and
writing characters in files-Reading Entire files-Random Access to file data-Getting
Information on Files-Ownership and permissions; Working with files you own: Splitting the
Name and path from a file-copying ,renaming and deleting files; Working with Directories:
other Directory Functions –Traversing a directory hierarchy-creating a directory navigator-
Building a Text Editor-Uploading Files.

Classes- Objects: Creating class- Adding a Method- Adding a Property- Protecting Access to
Member Variables- Using _get and _set- Initializing objects- Destroying Objects-
Inheritance- Overriding Methods- Interfaces- Encapsulation

UNIT IV
The SQL Framework- Managing databases-Creating & Managing tables- Managing indexes;
Inserting & Updating data in a MYSQL database-Deleting & Retrieving data from a MySQL
database; SELECT statement- Optional clauses of a SELECT statement; Creating MySQL
Expressions-using operators in expressions-Comparing and Converting Data; Managing
different types of data: String functions-Numeric function- Date/Time functions-
Summarizing date-Summary functions

Performing System Operations: Encryption functions- System related Functions- Query and
Insert Functions; Accessing data from Multiple tables: Creating joins in your SQL
statement- Creating subqueries in your SQL statements; Creating Unions that join SELECT
statements.

Exporting, Copying and importing data; Managing transactions: Introducing transactions-
Performing a transaction- Setting the auto commit mode and transaction isolation level-
Locking Nontransactional tables

UNIT V
Connecting to MySQL from a PHP application- Inserting and updating records in table-
Deleting and retrieving data from table- Creating a user Registration Script. Structure of an
E-Mail Message-sending E-mail with PHP- Working with Raster Images- Manipulating
Raster Images- Using Text in Images
TEXT BOOK


REFERENCE BOOKS


WEBSITES

1. www.php.net/
2. en.wikipedia.org/wiki/PHP
3. www.w3schools.com/PHP/DEfaULT.asp
Scope: Course explores new technologies driving mobile computing, and their implications. Expose students to software and hardware capabilities of mobile computing systems.

Objectives: To impart to students
- the principles of mobile computing technologies;
- knowledge of applications that mobile computing offers to people, employees, and businesses;
- knowledge of GPRS network architecture and services .
- future of mobile computing technologies and applications.

UNIT I

UNIT II

UNIT III

UNIT IV
UNIT V
Android: Getting to know Android - Android development environment - Android development environment for real applications - start up code, M J Android applications - debugging Android applications

TEXT BOOKS

REFERENCE BOOKS

WEBSITES:
1. en.wikipedia.org/wiki/Mobile_computing
2. www.cse.iitk.ac.in/users/rkg/Talks/mobile_main.pdf
3. www.tutorialspoint.com/android/
4. pl.cs.jhu.edu/oose/resources/android/Android-Tutorial.pdf
Scope: Scope is to teach the techniques for preprocessing data before mining, and the concepts related to data warehousing, on-line analytical processing (OLAP), and data generalization. It presents methods for mining frequent patterns, associations, and correlations.

Objective: To help the students

- To Possess knowledge of the concepts and terminology associated with database systems, statistics, and machine learning
- To develop skills of using recent data mining software for solving practical problems.
- To gain experience of doing independent study and research.

UNIT I
Introduction to Data Mining: Motivation and importance, Data Mining, Relational Databases, Data Warehouses, Transactional Databases, Advanced Database Systems and Advanced Database Applications, Data Mining Functionalities, Pattern Classification of Data Mining Systems, Major issues in Data Mining. Pre-process the Data- Data Cleaning, Data Integration and Transformation.

UNIT II

UNIT III
Mining Association rule in large Databases Association Rule Mining, Mining Single - Dimensional Boolean Association Rules from Transactional Databases, Mining Multilevel Association Rules from Transaction Databases, Mining Multidimensional Association Rules from Relational Databases and Dataware houses, From Association Mining to Correlation Analysis, Constraint-Based Association Mining.
UNIT IV
Mining Complex Types of Data: Mining Spatial Databases – Multimedia Databases – Time-series and Sequence Data – Text Databases – Web Data Mining – Search Engines.

UNIT V

TEXT BOOKS
1. Jiawei Han and Micheline Kamber. 2011. Data Mining Concepts and Techniques, 3rd Edition, Elsevier, India (Unit I, III, IV, V)

2. G.K.Gupta .2006. Prentice Hall India, New Delhi. Introduction to Data Mining with Case Studies (Unit – IV)


REFERENCE BOOKS


WEB SITES:
2. www.anderson.ucla.edu/faculty/jason.frand/teacher/technologies/palace/datamining.htm
15CAP504N  NETWORK ARCHITECTURE AND MANAGEMENT  4H - 4C

Instruction Hours / week: L: 4 T: 0 P: 0 C:4  Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3Hours

**Scope:** Scope of this course is to teach the concepts and techniques of network architecture management. Know how to Remote Network Monitoring in TCP/IP Networks and general concepts and architecture behind standards based network management.

**Objective:**
- Understand concepts and terminology associated with SNMP and TMN
- Decide routing protocol for complex network.
- Gain knowledge the internal architecture of routers
- Understand the fundamentals and requirements for packet routing in computer communication network.

**UNIT I**

**UNIT-II**

**UNIT-III**
UNIT-IV

UNIT-V

TEXT BOOKS

REFERENCE BOOKS

Scope:
To deliver students skills for planning, implementing a software projects and member of a team to develop and deliver.

Objective: To help students to

- Design, select and apply the most appropriate software engineering process for a given project, plan for a software project, identify its scope and risks, and estimate its cost and time.
- Implement software.
- Analyze, specify and document software requirements for a software system.
- Express and understand the importance of negotiation, effective work habits, leadership, and good communication with stakeholders, in written and oral forms, in a typical software development environment.

UNIT I
Introduction-Software Project Management -Project evaluation and programme Management- An overview of Project planning- Stepwise planning-Selection of an appropriate project Approach.

UNIT II
Software effort estimation: Problems with over- and underestimates-Software effort estimation Techniques - Estimating by analogy -Albrecht function point analysis -Function points Mark II –COSMIC full function points - COCOMO 13: a parametric productivity model. Activity planning: The objectives of activity Planning-Project schedules - Projects and activities - Sequencing and scheduling activities - Network planning models - Formulating a network model - Adding the time dimension - The forward pass - The backward pass - Identifying the critical path.

UNIT III
UNIT IV
Monitoring and control: Creating the framework-Collecting the data-Visualizing progress-Cost monitoring-Earned value analysis-Prioritizing monitoring - Getting the project back to target - Change control. Managing people in software environments: Understanding behavior -Organization behavior: a background - Selecting the right person for the job - Instruction in the best methods - Motivation - Stress -Health and safety -Some ethical and professional concern

UNIT V

TEXT BOOK

REFERENCE BOOK

WEB SITES
SEMESTER-V

15CAP504W  RUBY  PROGRAMMING  4H - 4C

Instruction Hours / week: L: 4 T: 0 P: 0 C: 4  Marks: Internal: 40 External: 60 Total: 100
End Semester Exam: 3 Hours

Scope:
This course covers the fundamental components of the Ruby Programming Language. Emphasis is placed on the object oriented aspects of Ruby. Topics include arrays, hashes, regular expressions, I/O, exceptions, modules, and applications areas. Ruby is a programming language with a focus on simplicity and productivity.

Objective: To help students to
- Develop server-side Ruby scripts for publishing on the Web
- Employ control structures, methods, procs, arrays and hashes to create Ruby programs
- Distinguish and use various Ruby datatypes
- Master the use of arrays and hashes
- Use the extensive pre bundled classes
- Use the I/O facilities of Ruby to read and write binary and text files
- Master the use of Iterators to loop through various data structures
- Use Exceptions in handling various run time errors
- Create Ruby modules
- Use the wide variety of Ruby Modules that come with the Ruby distribution
- Use object-oriented programming conventions to develop dynamic interactive Ruby applications

UNIT I

UNIT II
UNIT III

UNIT IV

UNIT V

TEXT BOOKS:

REFERENCES:

WEB SITES :
2. www.fincher.org/tips/Languages/Ruby
3. www.troubleshooters.com/codecorn/ruby/basictutorial.htm
Scope: Wireless Communications Systems are used to transfer information over short distance or long distances. It encompasses various types of fixed, mobile and portable two-way radios, cellular telephones, personal digital assistants and wireless networking. This course enriches the application of wireless technology in today’s world.

Objective:
- To know the fundamental concepts of Wireless Communication Systems.
- To learn the latest technology of Mobile Communication.
- To provide a strong foundation in the field of Wireless Networks.
- To know about the various IEEE Standards.

UNIT I

UNIT II

UNIT III

UNIT IV

UNIT V

TEXT BOOKS

REFERENCE BOOKS
SCOPE: This scope of this course to explain the students the fundamentals of big data analytics and the methodologies used in storing, manipulating, and analyzing big data.

OBJECTIVES: To impart to students the skills required to design scalable systems that can accept, store, and analyze large volumes of unstructured data.

UNIT-I


UNIT-II


UNIT-III

Hadoop - Hadoop Distributed File System - Hadoop MapReduce- The Hadoop foundation and Ecosystem.

UNIT-IV

Big Data Analytics-Text Analytics and Big Data-Customized Approaches for Analysis of Big Data

UNIT-V

Integrating Data Sources-Real-Time Data Streams and Complex Event Processing-Operationalizing Big Data.
TEXT BOOK


REFERENCES


WEB SITES

1. www.oracle.com/BigData
4. www.solacesystems.com
5. en.wikipedia.org/wiki/Big_data
Scope: This course helps to distinguish between local programming (on a single machine) and distributed programming using multiple components via a network. This course tends to cover what designers and programmers need to consider in developing applications in a distributed parallel computing environment.

Objective: To help students to
- Understand the architecture and topology of network
- Understand the design process of a distributed systems
- Examine distributed and parallel computing operating system
- Know the need and challenges of distributed database

UNIT – I

UNIT – II
Message ordering and Group Communication: message ordering paradigm – Asynchronous execution with synchronous communication – classification of application level multicast algorithm – distributed multicast algorithm at the network layer.

UNIT – III

UNIT – IV

UNIT – V
TEXT BOOK

REFERENCES BOOKS

WEB SITES
- wikipedia.org/wiki/Distributed_computing
- www.webopedia.com/TERM/D/distributed_computing.html
- www.tech-faq.com/distributed-computing.shtml
Scope:
Software metrics provide a quantitative basis for the development and validation of models of the software development process. Metrics can be used to improve software productivity and quality.

Objective: To help students to

- Understand the basic terminology and state fundamental facts about software metrics and process models.
- Identify the essential elements of a given metric or model, describe the interrelationships among its various elements.
- Understand software process assessment cycles, complexity metrics and models.

UNIT I

UNIT-II
Applying the seven basic quality tools in software development-Defact removal effectivenes-The rayaleigh model-Exponential distribution and reliability growth models-Quality management models

UNIT-III
In-process metrics for software testing-Complexity metrics and models-Metrics and lessons learned for object oriented projects-Availability metrics

UNIT-IV
Measuring and analyzing customer satisfaction-Conducting in-process quality assessments

UNIT-V
Software project assessments-Dos and don’t of software process improvement-Using function point metrics to measure software process improvement-Concluding remarks

TEXT BOOKS:

REFERENCE BOOK
Scope: This course provides the methods to discover, classify and build ontology for searching. To build and implement a small ontology that is semantically descriptive of chosen problem domain. To implement applications that can access, use and manipulate the ontology.

Objectives:
- To represent data from a chosen problem in XML with appropriate semantic tags obtained or derived from the ontology.
- To understand the semantic relationships among these data elements using Resource.
- To design and implement a web services application that “discovers” the data and/or other Description Framework (RDF).web services via the semantic web. Able to discover the capabilities and limitations of semantic web technology for many applications.

UNIT I

UNIT II

UNIT III
UNIT IV

UNIT V

TEXT BOOK:

REFERENCES:
Scope: This course provides a solid understanding of ATM in terms of the technology as well as how it is applied to real-life applications.

Objective:
- To understand about the ATM protocols
- To learn the latest technology ATM network.
- To provide a strong foundation in routing issues.
- To know about the various Real Time Transport Protocol.

UNIT I

UNIT II

UNIT III

UNIT IV
High Speed LANs: Fast Ethernet – ATM LAN’s – LANE.

UNIT V

TEXT BOOKS

REFERENCES
2. Uyless Black.1999. ATM Vol.1 and 2,PHP TR.
1. Design an online loan application form.
2. Design a form to submit your resume in net.
3. Design an application for Library Management System.
4. Design form for online reservation in air ways.
5. Design form for online shopping.
6. Design an application for creating an online Advertisement.
7. Design an application for student Information System.
8. Design an application to display cookies information.
9. Write a program for display environment variables in MySQL.
10. Write a program to count web page hits.
11. Design an email form that validates the inputs, produces errors when inputs are typed incorrectly, and send an email to you when submitted.
12. Design an application to upload multiple files.
1. Write a program to build your first Android Application “Hello World” with common activity.

2. Write a program which will implement Sub menu in android application.

3. Write a program which will implement Context menu (Floating List of Menu Items) in android application.

4. Write a program to displays the use of Relative Layout Views with different attributes.

5. Write a program to displays the use of Linear Layout Views with different attributes.

6. Write a program to implement a Custom Button and handle the display message on button press.

7. Write a program to implement the List View in your android application.

8. Write a program to implement between animations and rotate the text in your android application.

9. Write a sample program to create a progress bar for your android applications.

10. Write a program to show how to use Date picker control of ADK in your android applications.

11. Write a program which enables you to draw an image using bitmap class object.

12. Write a program which allows you to get image from web and displayed them using the Image View.
1. Use the following learning schemes, with the default settings to analyze the weather data (in weather.arff). For test options, first choose "Use training set", then choose "Percentage Split" using default 66% percentage split. Report model percent error rate.

2. Using iris dataset preprocess and classify it with J4.8 and Naïve Bayes Classifier. Examine the tree in the Classifier output panel.

3. Using the datasets ReutersCorn-Train and ReutersGrain-Train. Classify articles using binary attributes and word count attributes.

4. Apply any two association rule based algorithm for the supermarket analysis.

5. Using weka Experimenter perform comparison analysis of J48, oneR and ID3 for vote dataset.

6. Using Weka Experimenter perform comparison analysis of Naïve Bayes with different datasets.

7. Apply ZeroR, OneR, and J48, to classify the Iris data in an experiment using 10 train and test runs, with 66% of the data used for training and 34% used for testing.

8. Using Weka Knowledge flow Set up a flow to load an ARFF file (batch mode) and perform a cross-validation using J48 (WEKA’s C4.5 implementation).

9. Draw multiple ROC curves in the same plot window, using J48 and RandomForest as classifiers.

10. Use any three clustering algorithm on Vehicle data set and find best among them.

11. Perform Preprocessing, feature selection and apply any one of the algorithm each from clustering, Association and classification to find their performance.

12. Examine the performance of different filters for the breast cancer dataset.
1. Simple router configuration.
2. Access and utilize the router to set basic parameters.
3. Connect, configure, and verify operation status of a device interface.
4. Implement static and dynamic addressing services for hosts in a LAN environment.
5. Identify and correct common problems associated with IP addressing and host configurations.
6. Describe basic routing concepts (including: packet forwarding, router lookup process).
7. Configure, verify, and troubleshoot RIPv2.
8. Perform and verify routing configuration tasks for a static or default route given.
9. Configure, verify and troubleshoot DHCP and DNS operation on a router.
10. Configure and verify a PPP connection between routers.
Prepare a more detailed, organized and easy-to-read documentation, for any application software, which should describe the following using Moodle tool:

1. User Requirement Documentation (USD)
2. Requirement Analysis Documentation. (RAD)
3. User Interfaces Specification. (UIS)
4. Object Oriented Design (OOD) or Low Level Design (LLD)
5. Code Documentation (CD)
6. Testing Documentation (TD)
7. User’s Guide (UG)
1. Write a ruby program to perform basic array and hash operations
2. Write a code to choose random numbers and find the behaviour of the number
3. Develop a program which gets raised when you handle an exception
4. Write a ruby code to display grade sheet of students using case.
5. Write a ruby program to evaluate polynomial.
6. Write a ruby program to draw box and fill with special characters.
7. Write a program to copy each line from input file to output file.
8. Write a program to create a button and fill the button with colors.
9. Write a program to create different color balls and make it bounce on window.
10. Write ruby program to display notebook widget.
11. Develop a ruby program to manipulate text with font color and images.
12. Write ruby program to create a main thread and execute multiple process through the main thread.
13. Write a ruby code to display color pallet and open dialog with the help of tk controls.
14. Design an application form using tk classes and validate all fields on Rails framework.
(Any 8 Experiments)
1. AM Modulation
2. AM Demodulation
3. FM Modulation
4. FM Demodulation
5. Pulse Amplitude Modulation and Detection
6. Pulse Width Modulation
7. Pulse Position Modulation and Detection
8. Pulse Code Modulation and Detection
9. Amplitude Shift Keying and Detection
10. Frequency Shift Keying and Detection