Andhra Pradesh State Council of Higher Education

BCA Under CBCS w.e.f. 2016-2017

Table-1: B.C.A. SEMESTER – I

Sno	Course	Total Marks	Mid Sem Exam*	Sem End Exam	Teaching Hours	Credits
1	First Language English	100	25	75	4	3
2	*Foundation Course - 1* HVPE (Human Values & Professional Ethics)	50	0	50	2	2
3	*Foundation course -2* Communication & Soft Skills -1	50	0	50	2	2
4	Elementary Mathematics	100	25	75	6	5
5	Computer Fundamentals & MS office	100	25	75	4	3
6	MS OFFICE LAB	50	0	50	2	2
7	Programming Using "C"	100	25	75	4	3
8	Programming Using "C" Lab	50	0	50	2	2
9	Photoshop Lab	50	0	50	2	2
	Total	650			28	24

Andhra Pradesh State Council of Higher Education
BCA Under CBCS with effect from the academic year 2016-2017 course of study

Table-2: B.C.A. SEMESTER – II

Sno	Course	Total Marks	Mid Sem Exam*	Sem End Exam	Teaching Hours	Credits
1	First Language English	100	25	75	4	3
2	*Foundation course - 3* Environmental Sci	50	0	50	2	2
3	*Foundation course – 4A* **Adobe In Design**	50	0	50	2	2
4	Statistical Methods and their Applications	100	25	75	6	5
5	Operating Systems	100	25	75	4	3
6	Operating Systems Lab	50	0	50	2	2
7	Object Oriented Programming Using "C++"	100	25	75	4	3
8	Object Oriented Programming Using "C++" Lab	50	0	50	2	2
9	Adobe In Design Lab	50	0	50	2	2
	Total	650			28	24

Andhra Pradesh State Council of Higher Education

BCA Under CBCS with effect from the academic year 2016-2017 course of study

Table-3: B.C.A. SEMESTER – III

Sno	Course	Total Marks	Mid Sem Exam*	Sem End Exam	Teaching Hours	Credits
1	First Language English	100	25	75	4	3
2	*Foundation Course - 5* Entrepreneurship	50	0	50	2	2
3	*Foundation course -2B* Communication & Soft Skills -2	50	0	50	2	2
4	Accounts and Financial Mgmt	100	25	75	6	5
5	DBMS	100	25	75	4	3
6	DBMS Lab	50	0	50	2	2
7	Programming with JAVA	100	25	75	4	3
8	JAVA LAB	50	0	50	2	2
9	Tally Software	50	0	50	2	2
	Total	650			28	24

Andhra Pradesh State Council of Higher Education
BCA Under CBCS with effect from the academic year 2016-2017 course of study

Table-4: B.C.A. SEMESTER – IV

Sno	Course	Total Marks	Mid Sem Exam*	Sem End Exam	Teaching Hours	Credits
1	*Foundation Course – 2C* Communication & Soft Skills -3	50	0	50	2	2
2	*Foundation Course – 6* Analytical Skills	50	0	50	2	2
3	*Foundation Course - 7 ** CE (Citizenship Education)	50	0	50	2	2
4	*Foundation course – 4B* ICT – 2 **Dreamweaver**	50	0	50	2	2
5	Unix	100	25	75	4	3
6	Data Structures Using Java	100	25	75	4	3
7	Data Structures Using Java Lab	50	0	50	2	2
8	We**b** Programming	100	25	75	4	3
9	Web Programming Lab	50	0	50	2	2
10	Unix Lab	50	0	50	2	2
	Total	650			28	23

Andhra Pradesh State Council of Higher Education
BCA Under CBCS with effect from the academic year 2016-2017 course of study
Table-5: B.C.A. SEMESTER – V

Sno	Course	Total Marks	Mid Sem Exam*	Sem End Exam	Teaching Hours	Credits
1	Skill Development Course – 1(University's Choice)	50	0	50	2	2
2	Network Security	100	25	75	5	4
3	Software Engineering	100	25	75	5	4
4	OOAD	100	25	75	5	4
Elective – 1						
5.1	Data Mining & Ware Housing	100	25	75	3	2
5.2	Computer Networks					
5.3	Cyber Forensics					
Elective – 2						
6.1	Android Basics	100	25	75	3	2
6.2	Principles of Animation					
6.3	Software Testing Methodologies					
Elective – 1 (LAB)						
7.1	Data Mining Lab	50	0	50	2	2
7.2	Computer Networks Lab					
7.3	Cyber Forensics Lab					
Elective – 2 (Lab)						
8.1	Android Basics Lab	50	0	50	2	2
8.2	Computer Animation Lab					
8.3	Testing Tools Lab					
	Total	650			27	22

Andhra Pradesh State Council of Higher Education
BCA Under CBCS with effect from the academic year 2016-2017 course of study
Table-6: B.C.A. SEMESTER – VI

Sno	Course	Total Marks	Mid Sem Exam*	Sem End Exam	Teaching Hours	Credits
1	Skill Development Course – 2 (University's Choice)	50	0	50	2	2
2	Ecommerce	100	25	75	5	4
3	Open Source Software	100	25	75	5	4
4	Cloud Computing	100	25	75	5	4
Elective – 1						
5.1	Hadoop & R Language					
5.2	Network programming	100	25	75	5	4
5.3	Cyber Laws					
Elective – 2						
6.1	Advanced Android					
6.2	Design of Video Games	100	0	50	5	4
6.3	Advanced Software Testing					
Project Lab						
7	Main Project	100	0	100	2	2
	Total	650			29	24

Andhra Pradesh State Council of Higher Education
BCA I year I semester

COMPUTER FUNDAMENTALS AND MS OFFICE

Course Objectives
The objective of the course is to introduce the concepts of computer fundamental & their applications for the efficient use of office technology in a business environment.

Course Outcomes
1. Demonstrate the basic technicalities of creating Word documents for office use.
2. Create and design a spreadsheet for general office
3. Demonstrate the basic technicalities of creating a PowerPoint presentation.
4. Demonstrate the practices in data & files management

UNIT-I
Introduction to computers, characteristics and limitations of computer, Block diagram of computer, types of computers, uses of computers, computer generations. Number systems:binary,hexaand octal numbering system

UNIT-II
Input and output devices: Keyboard and mouse,inputting data in other ways, Types of Software:system software,Application software,commercial,open source,domain and free ware software , Memories: primary,secondary and cache memory.windows basics: desk top,start menu,icons.

UNIT III
System Software, Compilers, assemblers, loaders, Operating Systems fundamentals, Introduction to Algorithms and Programming Languages

UNIT IV
MS Word: Getting Started Working with Microsoft Office 2007. Understanding Word Basics Editing and Formatting Text. Formatting Documents Working with Graphic Objects

UNIT V
Microsoft Excel: Understanding Excel Basics. Formatting and Editing the Worksheet ,Using Formulas and Functions. Working with Charts.
Microsoft PowerPoint: Understanding PowerPoint Basics. Formatting and Modifying Presentations   Enhancing   the   Presentation

REFERENCE BOOK

1. Fundamentals Of Computers '' by REEMA THAREJA from OXFORD UNIVERSITY PRESS

2. Microsoft Office 2007 Fundamentals, 1st Edition By Laura Story, Dawna Walls (UNIT I, UNIT II, UNIT III, UNIT IV)

3. "Computer Fundamentals and Programming in C" by REEMA THAREJA from OXFORD UNIVERSITY PRESS

4. PC SOFTWARE UNDER WINDOWS by Puneet Kumar And Sushil Bhardwaj From Kalyani Publishers

Student Activity:

1. Identify the parts of your computer/laptop

2. Load trail version of recent MS office suit in your system

3. Prepare your profile in MS PP using animations and sound effects

COMPUTER FUNDAMENTALS AND MS OFFICE LAB

1. Prepare your class time table using different Text formatting's in a table.

2. Send a Call Letter for All Applicants to Inform Interview Details using Mail Merge

3. Type your mathematical problems in MS word using Mathematical Equation editor

4. Create Water Marking

5. Create Backup file

6. Create a short film with animation and sound effects

7. Create a payslip with details of employee salary

8. Calculate student grades using his internal and external marks details

9. Draw different types of charts for weather analysis of 5 successive years

10. Prepare an excel sheet for posting attendance of students in various subjects and create a formula for promoting students having 75% minimum attendance

11. Prepare an excel sheet for conducting objective entrance test having multiple choice answers.

12. Prepare an excel sheet for student details and create formulas for accessing student addresses, category etc.

BCA I year I semester

PROGRAMMING USING C

Objectives:
1. Learn how to solve common types of computing problems.
2. Learn data types and control structures of C
3. Learn to map problems to programming features of C.
4. Learn to write good portable C programs.

Outcomes:
Upon successful completion of the course, a student will be able to:
1. Appreciate and understand the working of a digital computer
2. Analyze a given problem and develop an algorithm to solve the problem
3. Improve upon a solution to a problem
4. Use the 'C' language constructs in the right way
5. Design, develop and test programs written in 'C'

UNIT I

Introduction to Algorithms and Programming Languages: Algorithm – Key features of Algorithms – examples of Algorithms – Flow Charts – Pseudo code – Programming Languages – Generation of Programming Languages – Structured Programming Language.

Introduction to C: Introduction – Structure of C Program – Writing the first C Program – File used in C Program – Compiling and Executing C Programs – Using Comments – Keywords – Identifiers – Basic Data Types in C – Variables – Constants – I/O Statements in C- Operators in C- Programming Examples – Type Conversion and Type Casting.

UNIT II

Decision Control and Looping Statements: Introduction to Decision Control Statements – Conditional Branching Statements – Iterative Statements – Nested Loops – Break and Continue Statement – Goto Statement.

Functions: Introduction – using functions – Function declaration/ prototype – Function definition – function call – return statement – Passing parameters – Scope of variables – Storage Classes – Recursive functions – Type of recursion – Towers of Hanoi.

UNIT III

Arrays: Introduction – Declaration of Arrays – Accessing elements of the Array – Storing Values in Array – Calculating the length of the Array – Operations that can be performed on Array – one dimensional array for inter-function communication – Two dimensional Arrays – Operations on Two Dimensional Arrays.

Strings: Introduction - String Operations – String and Character functions.

UNIT IV

Pointers: Understanding Computer Memory – Introduction to Pointers – declaring Pointer Variables – Pointer Expressions and Pointer Arithmetic – Null Pointers - Passing Arguments to Functions using Pointer – Pointer and Arrays – Passing Array to Function– Memory Allocation in C Programs – Memory Usage – Dynamic Memory Allocation – Drawbacks of Pointers

Structure, Union, and Enumerated Data Types: Introduction – Nested Structures – Arrays of Structures– Self referential Structures – Union– Enumerated Data Types.

UNIT V

Files: Introduction to Files – Using Files in C – Reading Data from Files – Writing Data from Files – Detecting the End-of-file – Error Handling during File Operations .

REFERENCE BOOKS

1. Computer Fundamentals and Programming in C by REEMA THAREJA from OXFORD UNIVERSITY PRESS

2. E Balagurusamy: ―COMPUTING FUNDAMENTALS & C PROGRAMMING‖ – Tata McGraw-Hill, Second Reprint 2008, ISBN 978-0-07-066909-3.

3. Ashok N Kamthane: Programming with ANSI and Turbo C, Pearson Edition Publ, 2002.

4. 2. Henry Mullish & Huubert L.Cooper: The Sprit of C, Jaico Pub. House,1996.

5. Teach your C Skills-Kanithker

Student Activity:

1. Create time table using faculty workload, subjects etc.

2. Prepare a complete note on recursion and its types

3. Prepare a complete note types of files and file formats for different inputdata

PROGRAMMING USING C LAB

1. Write a C program to calculate the expression: ((a*b)/c)+(a+b-c)
2. Write a C program to calculate $(a+b+c)^3$.
3. Program to convert temperature from
       a. Celsius to Fahrenheit.
       b. Fahrenheit to Celsius.
4. Write a C program to calculate the Compound Interest.
5. Program to convert Hours into seconds.
6. Write a C program to Fine Biggest of Three numbers.
7. Write a C program to read student marks in five subjects and calculate the Total, Average and Grade according to the following conditions:
       i. If average >=75 grade is _A'.
       ii. If average >=60 and <75 grade is _B'.
       iii. If average >=50 and <60 grade is _C'.
       iv. Otherwise grade is _D'.
       v. Check that marks in each subject >= 35.

8. Write a C program to find biggest of two numbers using Switch – Case.
9. Program to display number of days in given month using Switch – -Case.
10. Write a C program to check whether the given number is Prime or Not.
11. Write a program to
       i. Check whether given number is Palindrome or Not.
       ii. Find the Reverse of a given number.

12. Program to check whether a given number is
       i. Strong or Not.
       ii. Armstrong or Not.
       iii. Perfect or Not.

13. Write a C program to print Fibonacci Series.
14. Write a C Program to print Prime Numbers up to given range.

15. Write a program to print multiplication tables up to given range.
16. Write a C program to perform
       i. Matrix Multiplication.
17. Program to display Student Details using Structures.
18. Program to swap two numbers using different parameter passing techniques.
19. Write a C program to
       i. Write data into a File.
       ii. Read data from a File.

BCA I Year I Semester

Photo Shop Lab

Create following items using different options in photo shop

1. Visiting card
2. Cover page of a book
3. Paper add for calling tenders
4. Passport photo design
5. Pamphlet
6. Broacher designing
7. Titles designing
8. Custom shapes creation
9. Web template design
10. Black & white and color photo conversion
11. Image size modification
12. Wedding album designing
13. Background changes
14. Box package cover designing
15. Texture and patterns designing
16. Filter effects & Eraser effects

BCA I Year II Semester

OPERATING SYSTEMS

Course Objectives

1. To understand the services provided by and the design of an operating system.
2. To understand the structure and organization of the file system.
3. To understand what a process is and how processes are synchronized and scheduled.
4. To understand different approaches to memory management.
5. Students should be able to use system calls for managing processes, memory and the file system.
6. Students should understand the data structures and algorithms used to implement an OS.

Course Outcomes

1. Analyze the concepts of processes in operating system and illustration of the scheduling of processor for a given problem instance.
2. Identify the dead lock situation and provide appropriate solution so that protection and security of the operating system is also maintained.
3. Analyze memory management techniques, concepts of virtual memory and disk scheduling.
4. Understand the implementation of file systems and directories along with the interfacing of IO devices with the operating system.

UNIT - I

Operating System Introduction: Operating Systems Objectives and functions, Computer System Architecture, OS Structure, OS Operations, Evolution of Operating Systems - Simple Batch, Multi programmed, time shared, Parallel, Distributed Systems, Real-Time Systems, Operating System services.

UNIT - II

Process and CPU Scheduling - Process concepts - The Process, Process State, Process Control Block, Threads, Process Scheduling - Scheduling Queues, Schedulers, Context Switch, Preemptive Scheduling, Dispatcher, Scheduling Criteria, Scheduling algorithms, Case studies: Linux, Windows.

Process Coordination - Process Synchronization, The Critical section Problem, Synchronization Hardware, Semaphores, and Classic Problems of Synchronization, Monitors, Case Studies: Linux, Windows.

UNIT - III

Memory Management and Virtual Memory - Logical & physical Address Space, Swapping, Contiguous Allocation, Paging, Structure of Page Table. Segmentation, Segmentation with Paging, Virtual Memory, Demand Paging, Performance of Demanding Paging, Page Replacement Page Replacement Algorithms, Allocation of Frames.

UNIT - IV

File System Interface - The Concept of a File, Access methods, Directory Structure, File System Mounting, File Sharing, Protection, File System Structure,

Mass Storage Structure - Overview of Mass Storage Structure, Disk Structure, Disk Attachment, Disk Scheduling.

UNIT - V

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

.

REFERENCES BOOKS:

1. Operating System Principles, Abraham Silberchatz, Peter B. Galvin, Greg Gagne 8th Edition, Wiley Student Edition.
2. Principles of Operating Systems by Naresh Chauhan, OXFORD University Press
3. Operating systems - Internals and Design Principles, W. Stallings, 6th Edition, Pearson.
4. Modern Operating Systems, Andrew S Tanenbaum 3rd Edition PHI.
5. Operating Systems A concept - based Approach, 2nd Edition, D. M. Dhamdhere, TMH.
6. Principles of Operating Systems, B. L. Stuart, Cengage learning, India Edition.
7. Operating Systems, A. S. Godbole, 2nd Edition, TMH

Student Activity:
1. Load any new operating system into your computer.
2. Partition the memory in your system
3. Create a semaphore for process synchronization

Operating Systems Lab

1. Given the list of processes, their CPU burst times and arrival times, display/print the Gantt chart for FCFS and SJF. For each of the scheduling policies, compute and print the average waiting time and average turnaround time. (2 sessions)

2. Given the list of processes, their CPU burst times and arrival times, display/print the Gantt chart for Priority and Round robin. For each of the scheduling policies, compute and print the average waiting time and average turnaround time. (2 sessions)

3. Developing applications using Inter Process Communication (using shared memory, pipes or message queues)

4. Implement the Producer – Consumer problem using semaphores

5. Implement any two memory management schemes

6. Implement any two file allocation techniques (Linked, Indexed or Contiguous)

7. Implement any two Page Replacement Algorithms

8. Implement Deadlock prevention algorithm.

9. Implement any two disk scanning algorithms

BCA I Year II Semester

OBJECT ORIENTED PROGRAMMING USING C++

Course Objectives

This course covers object-oriented programming principles and techniques using C++. Topics include pointers, classes, overloading, data abstraction, information hiding, encapsulation, inheritance, polymorphism, file processing, templates, exceptions, container classes, and low-level language features. This course also covers basic concepts for software design and reuse.

Course Outcomes
1. Understand concepts of objects and their significance in real world
2. Investigate software problem in terms of objects and entities
3. Learn to co-relate relationship among different entities involved in a system
4. Find dependency and roles in an environment
5. Develop software in terms of objects, associations, and integrity constraints
6. Generalize and aggregate business entities and transform behavior into functions
7. Identify, understand and analyze various sample development models

UNIT I
Principles of OOP: Software Crisis. Software Evolution- Programming Paradigms. Object Oriented Technology- Basic concepts and benefits of OOP – Application of OOP, OOP languages
Introduction to C++: History of C++, Structure of C++, Application of C++, tokens, keywords, identifiers, basic data types, derived data types, derived data types, symbolic constant, dynamic initialization, reference variables, scope resolution operator, type modifiers, type casting operators and control statements, input and output statements in C++, Function prototyping and components, Passing parameters: Call by reference, Return by reference, Inline function, Default arguments, Over loaded function.

UNIT II
Classes and Objects: Class specification, Member function definition – nested member function, access qualifiers, static data members and, member functions. Instance creation - Array of objects - Dynamic objects - Static Objects – Objects as arguments -Returning objects
Constructors and Destructors: Constructors- Parameterized constructors, Overloaded Constructors, Constructors with default arguments, copy constructors, Destructors.

UNIT III
Operator Overloading: Operator function-overloading unary and binary operators, overloading the operator using Friend function, Stream operator overloading, Data conversion.

Inheritance: Defining derived classes. Single Inheritance - Protected data with private inheritance - Multiple Inheritances - Multi Level Inheritance - Hierarchical Inheritance. Hybrid Inheritance - Multi path Inheritance - Constructors in derived and base Class -Template in Inheritance - Abstract classes - Virtual function and Dynamic polymorphism. -
Virtual destructor - Nested Classes

UNIT- IV
Functions in C++ : Virtual functions- need for Virtual function, , Pure Virtual functions, Generic Programming with Templates. Introduction, function templates, overloaded function templates, user defined templates arguments, class templates, Inheritance of class templates.

UNIT-V

Files: file stream, file pointer and manipulation, file open and close, sequential and random access.
Exception Handling: Principle of Exception handling, Exception handling mechanism ,Multiple catch, Nested try, re throwing the Exception.

REFERENCE BOOKS:

1.1 Object Oriented Programming with C++ by Reema Thareja, OXFORD University Press

2.The Complete Reference C++, Herb Schildt, Tata McGraw-Hill, Fourth Edition.

3.Robert Lafore, "Object Oriented Programming in C++", Galgotia Publication Pvt. Ltd,4 th
  edition, New Delhi, 2002

4.Ashok N Kamathane, "Object Oriented Programming with ANSI & Turbo C++", Pearson
  Education, New Delhi, 2003.

5.Bjarne Stroustrup," C++ Programming language", Pearson Education, New Delhi, 2001.

6.Venugopal K R, Rajkumar Buyya and Ravishankar T," Mastering C++", TMH, ND, 2006

Student Activity:
1. Create a class diagram for academic process in your college
2. Write a program to implement "Vikuntapali"game

OBJECT ORIENTED PROGRAMMING USING C++ LAB

1. Write a C++ program to find the sum of individual digits of a positive integer.

2. A Fibonacci sequence is defined as follows: the first and second terms in the sequence are 0 and 1. Subsequent terms are found by adding the preceding two terms in the sequence. Write a C++ program to generate the first n terms of the sequence.

3. Write a C++program to generate all the prime numbers between 1 and n , where n is a value supplied by the user.

4. Write a C++program to find the factorial of a given integer

5. Write a C++program to find the GCD of two given integers

6. Write a C++ program that uses a recursive function for solving Towers of Hanoi problem.

7. Write a C++program to implement call by value and call by reference parameters passing

8. Write a C++ program to implement function templates

9. Write a program to implement Overloading and Overriding

10. Write a C++ program to implement the matrix ADT using a class. The operations supported by this ADT are:

a. Reading a matrix.

b. Printing a matrix

c. Addition of matrices

d. Subtraction of matrices

e. Multiplication of matrices

11. Write C++programs that illustrate how the Single inheritance, Multiple inheritance Multi level inheritance and Hierarchical inheritance forms of inheritance are supported

12. Write a C++program that illustrates the order of execution of constructors and destructors when new class is derived from more than one base class

13. Write a C++ program that illustrates how run time polymorphism is achieved using virtual functions

Adobe In Design LAB

Implement the following tasks using ADOBE

1. **Resume designing**
2. **Paragraph setting**
3. **Text column wise designing**
4. **Text base paper add**
5. **Create college Logo**
6. **Table creation**
7. **Student marks list**
8. **Book work**
9. **Picture insertion**
10. **Application form**
11. **Text based Visiting card**
12. **Notice designing**
13. **Typographic alignment styles**
14. **Wedding card designing**
15. **Letter models**

DATABASE MANAGEMENT SYSTEMS

Course Objectives
The objective of the course is to enable students to understand and use a relational database system. Introduction to Databases, Conceptual design using ERD, Functional dependencies and Normalization, Relational Algebra is covered in detail. Students learn how to design and create a good database and use various SQL operations. The course concludes with an overview of transaction management and introduction to advanced and non-relational databases.

Course Outcomes
1. Able to master the basic concepts and understand the applications of database systems.
2. Able to construct an Entity-Relationship (E-R) model from specifications and to transform to relational model.
3. Able to construct unary/binary/set/aggregate queries in Relational Algebra.
4. Understand and apply database normalization principles.
5. Able to construct SQL queries to perform CRUD operations on database. (Create, Retrieve, Update, Delete)
6. Understand principles of database transaction management, database recovery, security.

Unit-1

Database Systems: Introducing the database and DBMS, Files and File Systems, Problems with File System and advantages of Database Management systems.

Data Models: The importance of Data models, Data Model Basic Building Blocks, Business Rules, The evaluation of Data Models, Degree of Data Abstraction.

Unit-II

The Relational Database Model: A logical view of Data, Keys, Integrity Rules, Relational Set Operators, The Data Dictionary and the system catalog, Relationships with in the Relational Database, Data Redundancy revisited, Indexes, Codd's relational database rules.

Entity Relationship Model: The ER Model, Developing ER Diagram,

Unit-III

Normalization of database tables: Database Tables and Normalization, The need for Normalization, The Normal forms and High level Normal Forms, denormalization.

Unit-IV

Introduction to SQL: Data Definition Commands, Data Manipulation Commands, Select queries, Advanced Data Definition Commands, Advanced Select queries, Virtual Tables, Joining Database Tables.

Advanced SQL: Relational Set Operators, SQL Join Operators, Subqueries and correlated queries, SQL Functions, Oracle Sequences, and Procedural SQL.

Unit-V

Transaction Management and Concurrency Control: What is transaction, Concurrency control, Concurrency control with locking Methods, Concurrency control with time stamping methods, concurrency control with optimistic methods, database recovery management.

Reference Books:

1. Peter Rob, Carlos Coronel, Database Systems Design, Implementation and Management, Seventh Edition, Thomson (2007)

2. Elimasri / Navathe, Fundamentals of Database Systems, Fifth Edition, Pearson Addison Wesley (2007).

3. Raman A Mata – Toledo/Panline K Cushman, Database Management Systems, Schaum's Outlibe series, Tata McGraw Hill (2007).

4. C.J.Date, A.Kannan, S.Swamynathan, An Introduction to Database Systems, Eight Edition, Pearson Education (2006).

5. Atul Kahate, Introduction to Database Management Systems, Pearson Education (2006).

Student Activity:
1. Create student database of your college for placement purpose
2. Create student database of your college for fees reimbursement, ADHAAR as primary key

BCA II Year III Semester

DATABASE MANAGEMENT SYSTEMS LAB

1.Order Tracking Database

The Order Tracking Database consists of the following defined six relation schemas.

Employees(eno,ename,zip,hdate)

Parts(pno,pname,qoh,price,level) (hint: qoh: quality on hand)

Customers(cno,cname,street,zip,phone)

Orders(ono,cno,eno,received date,shipped date)

Odetails(ono,pno,qty)

Zipcodes(zip,city)

Solve the following queries

1. Get all pairs of customer numbers for customers based on same   zip   code.

2. Get part numbers for parts that have been ordered by at least two   different customers.

3. For each odetail row, get ono,pno,pname,qty and price values along with   the total price for the item. (total price=price*qty)

4. Get customer name and employee pairs such that the customer with name   has placed an order through the employee

5. Get customer names living in fort dodge or liberal.

6. Get cname values of customers who have ordered a product with   pno 10506.

7. Get pname values of parts with the lowest price.

8. Get cname values of customers who have placed at least one order through the employee with number 1000.

9. Get the cities in which customers or employees are located.

10. Get the total sales in dollars on all orders.

11. Get part name values that cost more than the average cost of all parts.

12. Get part names of parts ordered by at least two different Customers.

13. Get for each part get pno,pname and total sales

14. For each part, get pno,pname, total sales, whose total sales exceeds 1000

15. Get pno, part names of parts ordered by at least two different customers.

16. Get cname values of customers who have ordered parts from any one employee based in wichita or liberal.

2.Shipment database

An enterprise wishes to maintain the details about his suppliers and other corresponding details. For that it uses the following tables

Table s(sid,sname,address)

      primary key   : sid

Table p(pid,pname,color)

      primary key   : pid

Table cat(sid,pid,cost)

      primary key   : sid+pid

      reference key : sid references s.sid

                         pid references p.pid

Solve the following queries

1. Find the pnames of parts for which there is some supplier

2. Find the snames of suppliers who supply every part.

3. Find the snames of suppliers who suppy every red part.

4. Find the pnames of parts supllied by london supplier and by no

   one else

5. Find the sids of suppliers who charge more for some part other than the average cost of

   that part

6. Using group by with having clause get the part numbers for all the parts supplied by more

   than one supplier.

7. Get the names of the suppliers, who do not supply part p2.

8. Find the sids of suppliers who supply a red and a green part

9. Find the sids of suppliers who supply a red or a green part

10. find the total amount has to pay for that supplier by part

    located from london

3.Employee database

An enterprise wishes to maintain a database to automate its operations. Enterprise divided into to certain departments and each department consists of employees. The following two tables describes the automation schemas

Dept (deptno, dname, loc)

Emp (empno,ename,job,mgr,hiredate,sal,comm,deptno)

1.   Create a view, which contain employee names and their manager names working in sales department.

2.   Determine the names of employee, who earn more than their managers.

3.   Determine the names of employees, who take highest salary in   their departments.

4. Determine the employees, who located at the same place.
5.   Determine the employees, whose total salary is like the minimum salary
6.   of any department.
7.   Update the employee salary by 25%, whose experience is greater than 10 years.
8. Delete the employees, who completed 32 years of service.
9.   Determine the minimum salary of an employee and his details, who join on the same date.
10.   Determine the count of employees, who are taking commission and not taking Commission.

11.   Determine the department does not contain any employees.
12. Find out the details of top 5 earner of company.
13. Display those managers name whose salary is more than average salary of his employees.
1. Display those employees who joined the company before 15th of the month?
2. Display the manager who is having maximum number of employees working under him?
3. Print a list of employees displaying 'less salary' if less than 1500 if exactly 1500 display as 'exact salary' and if greater than 1500 display 'more salary'?
4. Display those employees whose first 2 characters from hire date-last 2 characters of salary?
5. Display those employees whose 10% of salary is equal to the year of joining?
6. In which year did most people join the company? Display the year and number of employees.
7.   Display the half of the enames in upper case and remaining lower case

8. Display ename, dname even if there no employees working in a particular department(use outer join).

4.Pl/sql programs

1. Write a pl/sql program to check the given number is strong or not.

2. Write a pl/sql program to check the given string is palindrome or not.

3. Write a pl/sql program to swap two numbers without using third variable.

4. Write a pl/sql program to generate multiplication tables for 2,4,6

5. Write a pl/sql program to display sum of even numbers and sum of odd

   numbers in the given range.

6. Write a pl/sql program to check the given number is pollinndrome or not.

7. write a pl/sql procedure to prepare an electricity bill by using

   following table

   table used: elect

name	null?	Type
mno	not null	number(3)
cname		varchar2(20)
cur_read		number(5)
prev_read		number(5)
no_units		number(5)
amount		number(8,2)
ser_tax		number(8,2)
net_amt		number(9,2)

8. Write a procedure to update the salary of employee, who belongs to
   certain department with a certain percentage of raise.

BCA II Year III Semester

Object Oriented Programming Using JAVA

Course Objectives

As the business environment becomes more sophisticated, the software development (software engineering is about managing complexity) is becoming increasingly complex. As of the best programming paradigm which helps to eliminate complexity of large projects, Object Oriented Programming (OOP) has become the predominant technique for writing software in the past decade. Many other important software development techniques are based upon the fundamental ideas captured by object-oriented programming.

Course Outcomes

At the end of this course student will:
1. Understand the concept and underlying principles of Object-Oriented Programming
2. Understand how object-oriented concepts are incorporated into the Java programming language
3. Develop problem-solving and programming skills using OOP concept
4. Understand the benefits of a well structured program
5. Develop the ability to solve real-world problems through software development in high-level programming language like Java
6. Develop efficient Java applets and applications using OOP concept
7. Become familiar with the fundamentals and acquire programming skills in the Java language.

UNIT-1

FUNDAMENTALS OF OBJECT – ORIENTED PROGRAMMING : Introduction, Object Oriented paradigm, Basic Concepts of OOP, Benefits of OOP, Applications of OOP, Java features: **OVERVIEW OF JAVA LANGUAGE**: Introduction, Simple Java program structure, Java tokens, Java Statements, Implementing a Java Program, Java Virtual Machine, Command line arguments. **CONSTANTS, VARIABLES & DATA TYPES:** Introduction, Constants, Variables, Data Types, Declaration of Variables, Giving Value to Variables, Scope of variables, Symbolic Constants, Type casting, Getting Value of Variables, Standard Default values; **OPERATORS & EXPRESSIONS**.

UNIT-II

DECISION MAKING & BRANCHING: Introduction, Decision making with if statement, Simple if statement, if. Else statement, Nesting of if. else statements, the else if ladder, the switch statement, the conditional operator. **LOOPING**: Introduction, The While statement, the do-while statement, the for statement, Jumps in loops.

CLASSES, OBJECTS & METHODS: Introduction, Defining a class, Adding variables, Adding methods, Creating objects, Accessing class members, Constructors, Method overloading, Static members, Nesting of methods;

UNIT-III

INHERITANCE: Extending a class, Overloading methods, Final variables and methods, Final classes, Abstract methods and classes;

ARRAYS, STRINGS AND VECTORS: Arrays, One-dimensional arrays, Creating an array, Two – dimensional arrays, Strings, Vectors, Wrapper classes;

INTERFACES: MULTIPLE INHERITANCE: Introduction, Defining interfaces, Extending interfaces, Implementing interfaces, Assessing interface variables;

UNIT-IV

MULTITHREADED PROGRAMMING: Introduction, Creating Threads, Extending the Threads, Stopping and Blocking a Thread, Lifecycle of a Thread, Using Thread Methods, Thread Exceptions, Thread Priority, Synchronization, Implementing the 'Runnable' Interface.

MANAGING ERRORS AND EXCEPTIONS: Types of errors : Compile-time errors, Run-time errors, Exceptions, Exception handling, Multiple Catch Statements, Using finally statement,

UNIT-V

APPLET PROGRAMMING: local and remote applets, Applets and Applications, Building Applet code, Applet Life cycle: Initialization state, Running state, Idle or stopped state, Dead state, Display state.

PACKAGES: Introduction, Java API Packages, Using System Packages, Naming conventions, Creating Packages, Accessing a Package, using a Package.

MANAGING INPUT/OUTPUT FILES IN JAVA: Introduction, Concept of Streams, Stream classes, Byte Stream Classes, Input Stream Classes, Output Stream Classes, Character Stream classes: Reader stream classes, Writer Stream classes, Using Streams, Reading and writing files.

Reference Books:

1. E.Balaguruswamy, Programming with JAVA, A primer, 3e, TATA McGraw-Hill Company.
2. Programming in Java by Sachin Malhotra, OXFORD University Press
3. John R. Hubbard, Programming with Java, Second Edition, Schaum's outline Series, TATA McGraw-Hill Company.
4. Deitel &Deitel. Java TM: How to Program, PHI (2007)
5. Java Programming: From Problem Analysis to Program Design- D.S Mallik
6. Object Oriented Programming Through Java by P. Radha Krishna, Universities Press (2008)

Student Activity:

1. Create a front end using JAVA for the student database created

2. Learn the difference between ODBC and JDBC

OBJECT ORIENTED PROGRAMMING USING JAVA LAB

1. Java program to demonstrate the use of Harmonic Series.

2. Java program to display a number of even, odd and sum of even, odd program.

3. Java program to find a sub string in the given string.

4. Java program to arrange the given strings in Alphabetic Order.

5. Java program to implements Addition and multiplication of two Matrices.

6. Java program to demonstrate the use of Constructor.

7. Java program to display a use of method overloading.

8. Java program to demonstrate the use of overriding Method.

9. Java program for single Inheritance.

10. Java program for implementing Interface.

11. Java program on Multiple Inheritance.

12. Java program for to implement Thread, Thread Priority,

13. Java program to demonstrate Exception handling.

14. Java program to demonstrate Applet program.

TALLY LAB

1) Kiran started a business with the following transactions

   i) Kiran started business with Rs. 1,00,000/-
   ii) Kiran purchased goods with Rs. 20,000/-
   iii) Kiran sold products for Rs. 40,000/-
   iv) And he spent Rs. 5,000/- towards for salaries.
   I. Practical exercise for the above transactions are

   a) Creating Company Transactions
   b) Creation of Ledgers.
   c) Record of Vouchers.
   d) Preparation of Balance Sheet.
   e) Preparation of Profit and Loss Account
   f) Trial Balance
   g) Day Book

2) Create the above records for any organization and get certified by them with comments

UNIX

Course Objectives
1. To understand Unix Operating System
2. To explore the Basic Shell Commands

Course Outcomes
After this course, the student will be able to
1. Implement and innovate commands using the basic tool kit.
2. Develop shell programs in vi/vim editor

Unit I

UNIX OPERATING SYSTEM
Overview of UNIX Operating System, basic features of Unix operating System, File Structure, CPU Scheduling, Memory Management, File System Implementation of Operating System Functions in UNIX.

Unit II

Starting Of Unix and Text Manipulation and user-to-user communication User Names and Groups, Logging In, Format of Unix Commands, Changing your password, Unix Documentation,

Unit III

Files and Directories: , File permission, Basic Operation on Files, Changing Permission Modes, Standard files , Processes Inspecting Files, Operating On Files, Printing Files, Rearranging Files, Sorting Files, Splitting Files, Translating Characters, On line communication, Off line communication.

Unit IV

VI EDITORS
General characteristics, Adding text and Navigation, changing text, searching for text, copying and Moving text, Features of Ex, Line Editors Ex and Ed, Stream editor SED, changing several file s in SED, AWK.

Unit V

Shell Programming:

Programming in the Bourne and C-Shell, Wild Cards, Simple Shell program, variables, Programming Construct, Interactive Shell scripts, Advanced Features, Unix Compiler, Maintaining program System Administration Define system Administration, Booting the system, Maintaining User Accounts, File System, and special files, Backup and Restoration.

References Books:
1. Unix and shell Programming by B.M Harwani, OXFORD University Press
2. Unix Concept and application- Sumitabhadas
3. Unix Shell Programming-Yashwant Kanetkar
4. Unix Programming Environment- RobPike
5. Unix in a Nutshell- Donill Gily

Student Activity:

1. Load unix/linux in your system in a separate drive

2. Create graphics in unix environment

BCA II Year IV Semester

Unix Lab

1. Execute of various file/directory handling commands.

2. Write a Simple shell script for basic arithmetic and logical calculations.

3. Write Shell scripts to check various attributes of files and directories.

4. Write Shell scripts to perform various operations on give n strings.

5. Write Shell scripts to explore system variables such as PATH, HOME etc.

6. Write Shell scripts to check and list attributes of processes.

7.Execute various system administrative commands

8.Write awk script that uses all of its features.

9.Use seed instruction to process /etc/password file.

10.Write a shell script to display list of users currently logged in.

11.Write a shell script to delete all the temporary files.

12.Write a shell script to search an element from an array using binary searching.

DATA STRUCUTURES USING JAVA

Course Objectives

To introduce the fundamental concept of data structures and to emphasize the importance of data structures in developing and implementing efficient algorithms. In addition, another objective of the course is to develop effective software engineering practice, emphasizing such principles as decomposition, procedural abstraction, and software reuse.

Course Outcomes

After completing this course satisfactorily, a student will be able to:
1. Describe how arrays, records, linked structures, stacks, queues, trees, and graphs are represented in memory and used by algorithms.
2. Describe common applications for arrays, records, linked structures, stacks, queues, trees, and graphs.
3. Write programs that use arrays, records, linked structures, stacks, queues, trees, and graphs
4. Demonstrate different methods for traversing trees
5. Compare alternative implementations of data structures with respect to performance
6. Compare and contrast the benefits of dynamic and static data structures implementations
7. Describe the concept of recursion, give examples of its use, describe how it can be implemented using a stack .
8. Discuss the computational efficiency of the principal algorithms for sorting, searching, and hashing.

UNIT I
Concept of Abstract Data Types (ADTs)- Data Types, Data Structures, Storage Structures, and File Structures, Primitive and Non-primitive Data Structures, Linear and Non-linear Structures.
Linear Lists - ADT, Array and Linked representations (Single and Double Linked lists), Pointers.

UNIT II
Stacks: Definition, ADT, Array and Linked representations, Implementations and Applications.
Queues: Definition, ADT, Array and Linked representations, Circular Queues, Dequeues, Priority Queues and Applications.

UNIT III

Trees: Binary Tree, Definition, Properties, ADT, Array and Linked representations, Implementations and Applications, Heaps Trees and Applications,

Binary Search Trees (BST) - Definition, ADT, Operations and Implementations, BST with Duplicates and Applications.

.

UNIT IV

Graphs – Graph and its Representation, Graph Traversals, Connected Components, Basic Searching Techniques, Minimal Spanning Trees.

UNIT- V

Sorting and Searching: Selection, Insertion, Bubble, Merge, Quick, Sequential and Binary Searching.

REFERENCE BOOKS

1. SamanthaD, Classic Data Structures, Prentice-Hall of India, 2001
2. Sahani S, Data Structures, Algorithms and Applications in C++, McGraw-Hill, 2002.
3. D S Malik, Data Structures Using C++, Thomson, India Edition 2006
4. Heilman G I,. Data Structures, Algorithms and Object-Oriented Programming, Tata McGraw-l lill. 2002. (Chapters I and 14).
5. Tremblay .1 P, and Sorenson P G, Introduction to Data Structures and Applications, Tata McGraw-Hill,
6. Drozdek A, Data Structures and Algorithms in C++), 2nd edition, Vikas Publishing House, 2002.
7. Kanetkar Y P, Data Structures through C ++, BPB Publications. 2003.
8. Data Structures by Allen Weiss

Student Activity:
1. Create Visual Stack using graphics in JAVA
2. Create Visual Queue using graphics in JAVA

DATA STRUCTURES USING JAVA LAB

1. Write Programs to implement the Stack operations using an array.
2. Write Programs to implement the Queue operations using an array.
3. Write Programs to implement the Stack operations using Pointers.
4. Write Programs to implement the Queue operations using Pointers.
5. Write a program for arithmetic expression evaluation.
6. Write a program for Binary search Tree Traversals
7. Write a program to implement dequeue using a doubly linked list.
8. Write a program to search an item in a given list using
   (i) Linear Search
   (ii) Binary Search.
9. Write a program for
   (i)Bubble Sort
   (ii) Quick Sort
   (iii)Merge Sort.
10. Write a program for polynomial addition using SLL

BCA II Year IV Semester

WEB PROGRAMMING

Course Objective

- To provide knowledge on web architecture, web services, client side and server side scripting technologies to focus on the development of web-based information systems and web services.
- To provide skills to design interactive and dynamic web sites.

Course Outcome

1. To understand the web architecture and web services.
2. To practice latest web technologies and tools by conducting experiments.
3. To design interactive web pages using HTML and Style sheets.
4. To study the framework and building blocks of .NET Integrated Development Environment.
5. To provide solutions by identifying and formulating IT related problems.

UNIT I

DNS – E-mail – FTP – TFTP – History of WWW – Basics of WWW and Browsing - Local information on the internet – HTML – Web Browser Architecture – Web Pages and Multimedia – Remote Login (TELNET).

UNIT II

Introduction to Web Technology: Web pages – Tiers – Concept of a Tier – Comparison of Microsoft and Java Technologies – Web Pages – Static Web Pages – Plug-ins – Frames – Forms. Dynamic Web Pages: Need – Magic of Dynamic Web Pages – Overview of Dynamic Web Page Technologies – Overview of DHTML – Common Gateway Interface.

UNIT III

ASP – ASP Technology – ASP Example – Modern Trends in ASP – Java and JVM – Java Servlets – Java Server Pages.
Active Web Pages: Active Web Pages in better solution – Java Applets – Why are Active Web Pages Powerful? – Lifecycle of Java Applets – ActiveX Controls – Java Beans. Middleware and Component-Based E-Commerce Architectures

UNIT IV

CORBA – Java Remote Method Invocation – DCOM. EDI: Overview – Origins of EDI – Understanding of EDI – Data Exchange Standards – EDI Architecture – Significance of EDI – Financial EDI – EDI and internet.

UNIT V

XML: SGML – Basics of XML – XML Parsers – Need for a standard. WAP: Limitations of Mobile devices – Emergence of WAP – WAP Architecture – WAP Stack – Concerns about WAP and its future – Alternatives to WAP.

REFERENCE BOOK
1. WEB TECHNOLOGIES TCP/IP to Internet Applications Architectures – Achyut S Godbole & Atul Kahate, 2007, TMH.
2. Web Technologies by Uttam Kumar Roy, Oxforn University Press
3. INTERNET AND WEB TECHNOLOGIES – Rajkamal, TMH.
4. TCP/IP PROTOCOL SUITE – Behrouz A. Forouzan, 3rd edition, TMH

Student Activity:

1. Design a website for your college

2. Design your personal web site

BCA II Year IV Semester

WEB PROGRAMMING LAB

1. Create a simple HTML page which demonstrates all types of lists.
2. Create a letter head of your college using following styles
   i. image as background
   ii. use header tags to format college name and address
3. Create a web page, which contains hyper links like fruits, flowers, animals. When you click on hyper links, it must take you to related web page; these web pages must contain with related images.
4. Create a hyperlink to move around within a single page rather than to load another page.
5. Create a leave letter using different text formatting tags.
6. Create a table format given bellow using row span and colspan.

RNO	NAME	MARKS				
		M1	M2	M3	M4	M5

   Insert 5 records.
7. Create a table with different formats as given bellow.
   i. Give different background and font colors to table header, footer and body.
   ii. Use table caption tag.
8. Divide a web page vertically and horizontally with scroll bars, name them as shown bellow decorate it with some items.

F1	F2
	F3

9. Create a student Bio-Data, using forms.
10. Create a web page using following style sheets
    i. Inline style sheets.
    ii. Embedded style sheets.
    iii. External style sheets
11. Write a JavaScript program to accept two values from form and apply any 5 mathematical functions

Write student database with XML

BCA III Year V Semester

Network Security

Course Objectives:

1. Understand Security: Attacks, Services & Mechanisms
2. Study of various security algorithm available for security and protection
3. Ciphering of plain text
4. Study of conventional encryption algorithm, key management issues
5. Cryptography and various encryption methods
6. Knowledge and implementation of hash function to messages
7. Digital signature and its importance in transaction processing
8. Concept of network security, directory authentication, e-mail
9. Viruses and their implication to business applications
10. Firewalls and network security principle

UNIT – I

INTRODUCTION:OSI Security Architecture - Classical Encryption techniques - Cipher Principles - Data Encryption Standard - Block Cipher Design Principles and Modes of Operation.

UNIT - II

PUBLIC KEY CRYPTOGRAPHY: Key Management - Diffie-Hellman key Exchange - Elliptic Curve Architecture and Cryptography - Introduction to Number Theory - Confidentiality using Symmetric Encryption - Public Key Cryptography and RSA.

UNIT – III

AUTHENTICATION AND HASH FUNCTION : Authentication requirements - Authentication functions - Message Authentication Codes - Hash Functions - Security of Hash Functions and MACs - MD5 message Digest algorithm - Secure Hash Algorithm - RIPEMD - HMAC Digital Signatures - Authentication Protocols - Digital Signature Standard

UNIT - IV

NETWORK SECURITY : Authentication Applications: Kerberos - X.509 Authentication Service - Electronic Mail Security - PGP - S/MIME - IP Security - Web Security.

UNIT – V

SYSTEM LEVEL SECURITY: Intrusion detection - password management - Viruses and related Threats - Virus Counter measures - Firewall Design Principles - Trusted Systems.

REFERENCES

1. William Stallings, "Cryptography And Network Security - Principles and Practices", Prentice Hall of India, Third Edition, 2003.

2. Atul Kahate, "Cryptography and Network Security", Tata McGraw-Hill, 2003.

3. Bruce Schneier, "Applied Cryptography", John Wiley & Sons Inc, 2001.

4. Charles B. Pfleeger, Shari Lawrence Pfleeger, "Security in Computing", Third Edition, Pearson Education, 2003.

Student Activity:

1. Create password verification using images

2. Create password verification using multimedia

BCA III Year V Semester
Software Engineering

Course Objectives

The Objective of the course is to assist the student in understanding the basic theory of software engineering, and to apply these basic theoretical principles to a group software development project.

Course outcomes
1. Ability to gather and specify requirements of the software projects.
2. Ability to analyze software requirements with existing tools
3. Able to differentiate different testing methodologies
4. Able to understand and apply the basic project management practices in real life projects
5. Ability to work in a team as well as independently on software projects

UNIT I

INTRODUCTION: Software Engineering Process paradigms - Project management - Process and Project Metrics – software estimation - Empirical estimation models - Planning - Risk analysis - Software project scheduling.

UNIT II
REQUIREMENTS ANALYSIS : Requirement Engineering Processes – Feasibility Study – Problem of Requirements – Software Requirement Analysis – Analysis Concepts and Principles – Analysis Process – Analysis Model

UNIT III
SOFTWARE DESIGN: Software design - Abstraction - Modularity - Software Architecture - Effective modular design - Cohesion and Coupling - Architectural design and Procedural design - Data flow oriented design.

UNIT IV
USER INTERFACE DESIGN AND REAL TIME SYSTEMS : User interface design - Human factors - Human computer interaction - Human - Computer Interface design - Interface design - Interface standards.

UNIT V
SOFTWARE QUALITY AND TESTING : Software Quality Assurance - Quality metrics - Software Reliability - Software testing - Path testing – Control Structures testing - Black Box testing - Integration, Validation and system testing - Reverse Engineering and Re-engineering.
CASE tools –projects management, tools - analysis and design tools – programming tools - integration and testing tool - Case studies.

REFERENCE BOOKS:

1. Roger Pressman S., "Software Engineering: A Practitioner's Approach", 7th Edition, McGraw Hill, 2010.
2.Software Engineering Principles and Practice by Deepak jain, Oxford University Press
3.Sommerville, "Software Engineering", Eighth Edition, Pearson Education, 2007
4.Pfleeger, "Software Engineering-Theory & Practice", 3rd Edition, Pearson Education, 2009
5. Carlo Ghazi, Mehdi Jazayari, Dino Mandrioli, "Fundamentals of Software Engineering", Pearson Education, 2003

Student Activity:

1.Develop requirement analysis report to develop software for any financial organization

2. Develop risk analysis report for any organization using software for its day to day transactions

BCA III Year V Semester
OOAD

Course Objective

- To develop background knowledge as well as core expertise in object oriented system.
- To provide the importance of the software design process.
- To assess the unified process and Unified Modeling Language

Course Outcome

- To describe the three pillars of object-orientation methodologies and explain the benefits of each .
- To create use case documents that capture requirements for a software system.
- To create class diagrams that model both the domain model and design model of a software system.
- To design the interface between the classes and objects.
- To create interaction diagrams that model the dynamic aspects of a software system.
- To understand the facets of the Unified Process approach to designing and building a software system.
- To describe how design patterns facilitate development and list several of the most popular patterns.
- To design the Axioms and corollaries.
- To build a model for the user interface (UI) of a software application
- To measure the Level of User satisfaction and software quality assurance.

UNIT I

Introduction to OOAD – What is OOAD? – What is UML? What are the United process(UP) phases - Case study – the NextGen POS system, Inception -Use case Modeling - Relating Use cases – include, extend and generalization.

UNIT II

Elaboration - Domain Models - Finding conceptual classes and description classes – Associations – Attributes – Domain model refinement – Finding conceptual class hierarchies-Aggregation and Composition- UML activity diagrams and modeling

UNIT III

System sequence diagrams - Relationship between sequence diagrams and use cases Logical architecture and UML package diagram – Logical architecture refinement - UML class diagrams - UML interaction diagrams

UNIT IV

GRASP: Designing objects with responsibilities – Creator – Information expert – Low Coupling –Controller – High Cohesion – Designing for visibility - Applying GoF design patterns – adapter, singleton, factory and observer patterns.

UNIT V

UML state diagrams and modeling - Operation contracts- Mapping design to code -UML deployment and component diagrams

REFERENCE BOOKS:
1. Object Oriented Analysis and Design By Grady Booch.
2. Craig Larman,"Applying UML and Patterns: An Introduction to object-oriented Analysis and Design and iterative development", Third Edition, Pearson Education, 2005
3. Mike O'Docherty, "Object-Oriented Analysis & Design: Understanding System Development with UML 2.0", John Wiley & Sons, 2005.
4. James W- Cooper, Addison-Wesley, "Java Design Patterns – A Tutorial", 2000.
5. Micheal Blaha, James Rambaugh, "Object-Oriented Modeling and Design with UML", Second Edition, Prentice Hall of India Private Limited, 2007
6. Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides,"Design patterns: Elements of Reusable object-oriented software", Addison-Wesley, 1995.

Student Activity:
1. Develop a class diagram for the flight services available in your near by air port
2. Develop a sequence diagram of activities of any automated device

BCA III Year V Semester

ELECTIVE 1: Data mining & Ware Housing

Course Objectives

The Objective of this course is to understand data mining principles and techniques: Introduce DM as a cutting edge business intelligence method and acquaint the students with the DM techniques for building competitive advantage through proactive analysis, predictive modeling, and identifying new trends and behaviors. Develop and apply critical thinking, problem-solving, and decision-making skills

Course Outcomes

1. Examine the types of the data to be mined and present a general classification of tasks and primitives to integrate a data mining system.
2. Apply preprocessing statistical methods for any given raw data
3. Discover interesting patterns from large amounts of data to analyze and extract patterns to solve problems , make predictions of outcomes
4. Comprehend the roles that data mining plays in various fields and manipulate different data mining techniques
5. Select and apply proper data mining algorithms to build analytical applications.
6. Evaluate and implement a wide range of emerging and newly-adopted methodologies and technologies to facilitate the knowledge discovery.

Unit I
Introduction to Data Mining, Fundamentals of data mining, data mining functionalities, data and attribute types, statistical description of data.
Data Preprocessing:
Data cleaning, data integration, data reduction, data transformation and data discretization.

Unit II
Data Warehousing: Basic concepts, data ware house modeling data cube and OLAP,
data warehouse design and implementation.

Unit III

Mining Frequent Patterns and Associations: Basic methods, frequent Item set mining methods any two algorithms, pattern evaluation methods.

Unit IV
Classification: Basic concepts, decision tree induction, Bayes classification, any two advanced methods, model evaluation.

Unit V

Cluster Analysis: Basic concepts, clustering structures, major clustering approaches, partitioning methods, hierarchical methods, density based methods, the expectation maximization method, cluster based outlier detection Essential Reading.

References:

1. 1.Data Mining by Vikram Pudi, P.Radha Krishna, Oxford Universith Press
2. Data Warehousing by Reema Thareja , Oxford University Press
3. 1.J. Han , M. Kamber and J. Pei , Data Mining: Concepts and Techniques , 3rd.edMorgan Kaufmann, 2011
4. Introduction to data mining –G.K.Gupta, PHI
5. 3.Data mining, Data warehouse & Olap-Berson, Tata McGraw Hill

Student Activity:
1.Predict the course taken by a student based on his activities and way of learning
2. Learn visual patterns of any real time data

Data Mining Lab

1. Demonstration of preprocessing on dataset student.arff
2. Demonstration of preprocessing on dataset labor.arff
3. Demonstration of Association rule process on dataset contactlenses.arff using apriori algorithm
4. Demonstration of Association rule process on dataset test.arff using apriori algorithm
5. Demonstration of classification rule process on dataset student.arff using j48 algorithm
6. Demonstration of classification rule process on dataset employee.arff using j48 algorithm
7. Demonstration of classification rule process on dataset employee.arff using id3 algorithm
8. Demonstration of classification rule process on dataset employee.arff using naïve bayes algorithm
9. Demonstration of clustering rule process on dataset iris.arff using simple k-means
10. Demonstration of clustering rule process on dataset student.arff using simple k-means.

BCA III Year V Semester

ELECTIVE 1: Computer Networks

Course Objectives

1. To provide an introduction to the fundamental concepts on data communication and the design of computer networks.
2. To get familiarized with the basic protocols of computer networks.

Course Outcomes
After this course, the student will be able to
1. Identify the different components in a Communication System and their respective roles.
2. Describe the technical issues related to the local Area Networks
3. Identify the common technologies available in establishing LAN infrastructure.

UNIT I
Network architecture – layers – Physical links – Channel access on links – Hybrid multiple access techniques - Issues in the data link layer - Framing – Error correction and detection – Link-level Flow Control.

UNIT II
Medium access – CSMA – Ethernet – Token ring – FDDI - Wireless LAN – Bridges and witches

UNIT III
Circuit switching vs. packet switching / Packet switched networks – IP – ARP – RARP –DHCP – ICMP – Queueing discipline – Routing algorithms – RIP – OSPF – Subnetting – CIDR – Interdomain routing – BGP – Ipv6 – Multicasting – Congestion avoidance in network layer

UNIT IV
UDP – TCP – Adaptive Flow Control – Adaptive Retransmission - Congestion control – Congestion avoidance – QoS

UNIT V
Email (SMTP, MIME, IMAP, POP3) – HTTP – DNS- SNMP – Telnet – FTP – Security – PGP - SSH

REFERENCE BOOKS:

1. Andrew S. Tanenbaum, "Computer Networks", Fourth Edition, 2003

2. Computer Networks by Bhushan Trivedi,Oxford University Press

3. James F. Kuross, Keith W. Ross, "Computer Networking, A Top-Down Approach Featuring the Internet", Third Edition, Addison Wesley, 2004.

4.Nader F. Mir, "Computer and Communication Networks", Pearson Education, 2007

5.Comer, "Computer Networks and Internets with Internet Applications", Fourth Edition, Pearson Education, 2003.

6. William Stallings, "Data and Computer Communication", Sixth Edition, Pearson vi, 2000

Student Activity:

1. Learn the functioning of various network devices used in your college network

2. Compare 2G,3G,4G and 5G networks

3. Prepare LAN deployment diagram of your organization

Computer Networks lab

1 Implementation of Stop and Wait Protocol and Sliding Window Protocol

2 Study of Socket Programming and Client – Server model

3 Write a code simulating ARP /RARP protocols.

4. Write a code simulating PING and TRACEROUTE commands

5. Create a socket for HTTP for web page upload and download.

6. Write a program to implement RPC (Remote Procedure Call).

7. Implementation of Subnetting.

8. Applications using TCP Sockets like
      a. Echo client and echo server   b. Chat   C. File Transfer

9. Applications using TCP and UDP Sockets like DNS, SNMP and File Transfer.

10. Study of Network simulator (NS).and Simulation of Congestion
Control Algorithms using NS

11. Perform a case study about the different routing algorithms to select the network path with its optimum and economical during data transfer.

BCA III Year V Semester

ELECTIVE 3: Computer Forensics

Course Objectives

- Explain the responsibilities and liabilities of a computer forensic investigator
- Plan and prepare for an incident requiring computer forensic skills
- Seize a computer from a crime scene without damaging it or risking it becoming inadmissible in a court of law
- Explain where digital evidence resides on computer storage devices
- Hire experts to perform detailed forensic analysis and expert testimony

Course Outcomes

1. understand the role of computer forensics in the business and private world
2. identify some of the current techniques and tools for forensic examinations
3. describe and identify basic principles of good professional practice for a forensic computing practitioner
4. Apply forensic tools in different situations.

Unit I

Introduction to Computer Forensics *:* Computer forensics definitions ,Computers' roles in crimes ,Computer forensics tasks ,Prepare for an investigation, Collect evidence ,Preserve evidence ,Recover evidence, Document evidence Challenges associated with making "cybercrime" laws, Jurisdictional issues.

Unit II

Computer Crimes :Crimes ,Violent crimes where computers are used include terrorism, assault threat, stalking, child pornography ,Nonviolent crimes where computers are used include trespass, theft, fraud, vandalism , Where evidence often resides for different types of crimes ,Address books, chat logs, e-mail, images, movies, Internet browser history, etc.

Unit III

Computer Criminals: Using evidence to create a crime timeline , Modify Access Create (MAC) dates associated with files ,Problems with using these (they don't change in a logical fashion in some cases) ,Criminals and crime fighters ,Understanding "cyber criminals" and their victims ,Understanding "cyber investigators.

Unit IV

Building a Cybercrime Case*: Bodies of law ,Constitutional law ,Criminal law ,Civil law ,Administrative regulations ,Levels of law ,Local laws ,State laws ,Federal laws ,International laws ,Levels of culpability ,Intent ,Knowledge ,Recklessness ,Negligence , Level and burden of proof ,Criminal versus civil cases ,Vicarious liability ,Laws related to computers ,CFAA, DMCA, CAN Spam, etc.

Unit V
Preserving and Recovering Digital Evidence*: Disk imaging ,Creating a message digest or hash code for a disk ,Where data hides; deleted and erased data ,File systems ,Files ,Modify Access Create (MAC) dates to establish time line ,File headers - info about file type

References books

1. Guide to Computer Forensics and Investigations ,By Bill Nelson, Amelia Phillips,

   christopher Steuart

2. Scene of the Cybercrime, by Debra Littlejohn Shinder.

3. John R. Vacca, Computer Forensics: Computer Crime Scene Investigation, 2nd Edition, Charles River Media, 2005

4. Christof Paar, Jan Pelzl, Understanding Cryptography: A Textbook for Students and Practitioners, 2 nd Edition, Springer's, 2010

5 . Ali Jahangiri, Live Hacking: The Ultimate Guide to Hacking Techniques & Countermeasures for Ethical Hackers & IT Security Experts, Ali Jahangiri, 2009

6. Computer Forensics: Investigating Network Intrusions and Cyber Crime (Ec-Council Press Series: Computer Forensics), 2010

Student Activity:

1. Collect calls made from a cell tower and analyze them
2. Trace the IP address of the machine from which you received a email

Computer Forensics lab

The Sleuth Kit and Autopsy

The Sleuth Kit (TSK) and the Autopsy Forensic Browser are open source Unix-based tools . TSK is a collection of over 20 command line tools that can analyze disk and file system images for evidence. To make the analysis easier, the Autopsy Forensic Browser can be used. Autopsy is a front end to the TSK tools and provides a point-and-click type of interface.

1) Use of disk tools to analyze the tool displays the total number of sectors and the user-accessible sectors.

2) Use of volume system tools to analyze the disk volume and partitions , whether they are allocated properly or not

3) File system tools to analyze the file system , its type and its description

4) Content category tool to analyze the data in the directory

5) Meta data category tool to analyze the data that describes a file

6) File name category tool to analyze The file name category of data includes the data that associates a name with a metadata entry.

7) Multiple category tool to analyze that combine the data from the various categories to produce the data sorted in a different order

8) Experiment to use the searching tools

BCA III Year V Semester
Elective -2 :

Android Basics

Course Objectives:
1. Describe the platforms upon which the Android operating system will run.
2. Create a simple application that runs under the Android operating system.
3. Access and work with the Android file system.
4. Create an application that uses multimedia under the Android operating system.
5. Access and work with databases under the Android operating system.

Course Outcomes

After completion of this course students should make Android apps for Android devices.Students will be able to write simple GUI applications, use built-in widgets and components, work with the database to store data locally, and much more.

UNIT-I

What is Android, Android Tools, Your First Android Application, Anatomy of Android Application, Workspaces, Editors in Eclipse, Eclipse Perspective, Refactoring

UNIT-II

Creating Android Emulator, Creating Snapshot, SD Card Emulation, Sending SMS Messages to the Emulator , Transferring Files into and out of the Emulator ,Resetting the Emulator

UNIT-III

Activity, Linking Activity using Intent, Fragments, Calling Build-In Application using Intent, Notifications

UNIT-IV

Components of a Screen, Display Orientation, Action Bar, Listening for User Inter

UNIT-V

Basic Views, Picker Views, List View, Specialized Fragment, Gallery and Image View, Image Switcher, Grid View, Options Menu, Context Menu, Clock View, Web view

Reference Books:

1.Android Programming: The Big Nerd Ranch Guide (Big Nerd Ranch Guides) By: Bill Philips & Brian Hardy

2.Android Design Patterns: Interaction design solutions for developers by Greg Nudelman

3.Android User Interface Design: Turning Ideas and Sketches into Beautifully Designed Apps By: Ian G. Clifton

4. Android Recipes: A Problem-Solution Approach By: Dave Smith & Jeff Friesen

5. Hello, Android: Introducing Google's Mobile Development Platform (Pragmatic Programmers) By: Ed Burnette

Student Activity:

1. Create a mobile APP for your college

2. Create a mobile APP for any rural application

Android basics lab

Exercise 1

- *Developing Simple Applications for Android*

Exercise 2

- *Creating Applications with Multiple Activities and a Simple Menu using ListView*

Exercise 3

- *Creating Activities For Menu Items and Parsing XML Files*

Exercise 4

- *Writing Multi-Threaded Applications*

Exercise 5

- *Using WebView and Using the Network*

Exercise 6

- *Using Audio Functions in Android*

Exercise 7

- *Graphics Support in Android*

Exercise 8

- *Preferences and Content Providers*

Exercise 9

- *Location Services and Google Maps in Android*
- *Data Storage*

Exercise 10

- *Simulating Sensors*

BCA III Year V Semester
Elective -2 : **Principles of Animation**

Course Objectives

Students can expect to learn basic principles and relevant techniques for comprehending the underlying

Course Outcome

- Understanding of the key principles of animation.
- Understanding of the concept of timing for animation and its application as a means of communication.
- Ability to creatively manipulate frame time as a means of emphasizing and actualizing action and expressing an idea.

UNIT-I

What is Animation: Its definition, early examples of Animation. History of Animation: Stop Motion Photo Animation, Zoetrope, Thaumatrope, Cell and Paper Animation, early Disney's Cell Animation Processes

UNIT-II
Types of Animation: Cell Animation, Stop Motion Animation, Computer Animation, 2-D Animation, 3-D Animation. Skills for an Animation Artist: Visual and creative development of an Artist , importance of observation with minute details, efficiency to draw gestures, facial expressions, good listener, hard work and patience, creative and innovative.

 UNIT-III
Basic Principles of Animation: Illusion of Life, straight action and pose to pose Timing, Exaggeration, Drama and Psychological Effect, Fade in and Fade out, Squash and Stretch, Anticipation, staging, follow through and overlapping action, Arcs, Solid Drawing ,Appeal, slow in and slow out, Secondary Action.

UNIT-IV
Various Terms: Animation Drawings/Cels, Rough Drawings , Clean ups, Color reference drawings, Layout, Model Sheet, Key Drawings and in Betweens, Master Background, Concept Piece, Character drawing , Story Board.

References:

1.The complete animation course by Chris Patmore -Baron's Educational Series.(New York)

2. Animation Unleashed by Ellen Bessen, Michael Weise Productions,2008(U.S.A)

3.The Animator's Survival Kit by Richard Williams, Arrar Straus & Giroux Pub.(U.S.A

Student Activity:

1. Develop a simple animated short film

2. Develop a simple animated short film with back ground music

Computer Animation lab

ADVANCED 2D ANIMATION

1: Action scripting
Using actions to control a timeline - Using frame labels - Creating button symbols - Creating animated buttons using movie clips – Movie Clip Controls – Browser / network.

2: Advanced Animation Methods
Creating movies playing within movies (movie clips and .swf) - Controlling multiple timelines (movies) through action scripting - Critique storyboards.

3: Streamlining Files for Use on the Web, Publishing Files to the Internet & Pre loaders
Pre loaders - Controlling sound with script - Exploring types of output - Work on final project in class - Importing video - Publishing demo (video) reels on web - Publishing and exporting files - Trouble shooting sites.

Suggested books for Reading:
Flash books

- The Illusion of Life: Disney Animation by Frank Thomas, Ollie Johnston (Contributor), Collie Johnston.
- Adobe Flash CS3
- The Animator's Survival Kit: A Manual of Methods, Principles, and Formulas for Classical, Computer, Games, Stop Motion, and Internet Animators by Richard Williams

BCA III Year V Semester

Elective -2 : **SOFTWARE TESTING METHODOLOGIES**

Course Objectives

The Objective of this course is to enable a clear understanding and knowledge of the foundations, techniques, and tools in the area of software testing and its practice in the industry. The course will prepare students to be leaders in software testing. Whether you are a developer or a tester, you must test software**.** We can learn strengths and weaknesses of a variety of software testing techniques.

Course Outcomes

After completion of this course the student will be able to plan, develop, and execute an automated test plan.

UNIT-I

Introduction: Purpose of testing, Dichotomies, model for testing, consequences of bugs, taxonomy of Bugs.

Flow Graphs and Path testing: Basics concepts of path testing, predicates, path predicates and Achievable paths, path sensitizing, path instrumentation, application of path testing.

UNIT-II

Transaction Flow Testing: Transaction flow, transaction flow testing techniques.

Dataflow testing: Basics of dataflow testing, strategies in dataflow testing, application of dataflow testing.

UNIT-III

Domain Testing: domains and paths, Nice & ugly domains, domain testing domains and interfaces Testing, domain and interface testing, domains and testability.

UNIT-IV

Paths, Path products and Regular Expressions: Path products & path expression, reduction procedure, Applications, regular expressions & flow anomaly detection.

Logic Based Testing: Overview, decision tables, path expressions kv charts, specifications.

UNIT-V

State, State Graphs and Transition testing: State graphs, good & bad state graphs state testing, Testability tips.

Graph Matrices and Application: Motivational overview, matrix of graph, relations, power of a matrix, Node reduction algorithm, building tools. (Student should be given an exposure to a tool like J Meter or Win runner.)

Reference Books

1. Software Testing techniques –Baris Beizerm Dreamtech, Second edition.

2. Software Testing Tools – Dr. K.V.K.K. Prasad, Dreamtech.
3. Software Testing Principles and Practices by Naresh Chauhan, Oxford University Press
3. The craft of software testing – Brain Matrick, Pearson Education.
4. Software Testing Techniques – SPD (Oreille)
5. Software Testing in the Real World-Edward Kit, Pearson.
6. Effective methods of Software Testing, Peery, John Wiley.
7. Art of Software Testing – Meyers, John Wiley.

Student Activity:

1. Prepare a chart for guidelines for data security in your organization

2.Test the performance of any software that is used by your organization under maximum load

Testing Tools Lab

1. Introduction to win runner testing tool
2. Recording test in context sensitive & analog mode
3. Synchronizing test.
4. Checking gui objects
5. Checking bitmap objects.
6. Programming test with tsl
7. Creating data driven test
8. Maintaining test script
9. Batch test
10. Project (creating test report)

BCA III Year VI Semester

E-commerce

Course Objectives

1. To develop an understanding of scope of E-Commerce.
2. To develop an understanding of electronic market and market place.
3. To develop an understanding of business models.
4. To develop an understanding of legal issues, threats of E-Commerce.

Course Outcomes
1. Students would be able to analyze the concept of electronic market and market place.
2. Students would be able to understand the business models.
3. Students would be able to understand the business standards.
4. Students would be able to understand the legal and security issues.

Unit-I
Overview of Electronic Commerce: Main Activities of E-Commerce, Broad Goals of E-Commerce, E-Commerce technical Components, Functions of E-Commerce, Prospectus of E-commerce, Lessons from E-commerce Evolution, Scope of E-commerce.
Unit-II

 E-commerce Technical Architecture, E- Commerce Strategies, E-commerce Essentials, E-commerce applications, Foundation of E-commerce, Growth of E-Commerce, Advantages of E-Commerce, Disadvantages of E-commerce, progress of E-commerce in India.

Unit-III

Driving the E-commerce Revolution. E-commerce Activities, Matrix of E-commerce models, B2C, B2B, B2B Boom, E-commerce opportunity Frame work, Developing an E-commerce Strategy, International E-commerce, International Strategy Development, Dotcom Companies.

Unit-IV
Electronic Market:-Online Shopping, Online Purchasing, Electronic Market, Three models of Electronic Market, Markets category, International Marketing, one-to –one Marketing, Permission Marketing, pull and push technologies, B2B Hubs, B2B market places, B2B exchange.

Unit-V
Electronic Business: Electronic Business applications Emerging applications, Electronic Business Architecture, AMR Model for Electronic Business, Evolution of Electronic Business

Application, Dotcom companies, The Indian scenario for E-Business, electronic business implementations, B2B E-commerce, B2C E-commerce, B2B Market Place..

References:

1. E-Commerce Concepts. Models, Strategies C.S.V Murthy, Himalaya Publishing House

2..The Complete E-Commerce Book: Design, Build & Maintain a Successful Web-based Business by Janice Reynolds

3..E-Commerce: Fundamentals and Applications by Henry Chan, Raymond Lee, Tharam Dillon, Elizabeth Chang November 2001
Student Activity:
1. Study the activities of any E-Commerce website and give suggestions to improve their business
2. Prepare your own E-commerce business site

BCA III Year VI Semester
Open Source Software

Course Objective

This course provides an overview of the historical and modern context and operation of free and open source software (FOSS) communities and associated software projects. The practical objective of the course is to teach students how they can begin to participate in a FOSS project in order to contribute to and improve aspects of the software that they feel are wrong. Students will learn some important FOSS tools and techniques for contributing to projects and how to set up their own FOSS projects.

Course Outcomes

Ability to install and run open-source operating systems. Ability to gather information about Free and Open Source Software projects from software releases and from sites on the internet. Ability to build and modify one or more Free and Open Source Software packages. Ability to use a version control system and to interface with version control systems used by development communities. Ability to contribute software to and interact with Free and Open Source Software development projects.

UNIT-I

Introduction to Open sources – Need of Open Sources – Advantages of Open Sources– Application of Open Sources.

UNIT-II

Open source operating systems: LINUX: Introduction – General Overview – Kernel Mode and user mode. Process – Advanced Concepts – Scheduling – Personalities – Cloning – Signals

UNIT-III

OPEN SOURCE DATABASE: MySQL: Introduction – Setting up account – Starting, terminating and writing your own SQL programs – Record selection Technology – Working with strings – Date and Time– Sorting Query Results

UNIT-IV

OPEN SOURCE PROGRAMMING LANGUAGES : PHP: Introduction – Programming in web environment – variables – constants – data types – operators – Statements – Functions – Arrays – OOP – String Manipulation and regular expression.

UNIT-V

PERL : Perl backgrounder – Perl overview – Perl parsing rules – Variables and Data – Statements and Control structures – Subroutines, Packages, and Modules- Working with Files – Data Manipulation.

REFERENCE BOOKS:

1. Rasmus Lerdorf and Levin Tatroe, "Programming PHP", O'Reilly, 2002

2. Wesley J. Chun, "Core Phython Programming", Prentice Hall, 2001

3. Martin C. Brown, "Perl: The Complete Reference", 2nd Edition, Tata McGrawHill Publishing Company Limited, Indian Reprint 2009.

4. Steven Holzner, "PHP: The Complete Reference", 2nd Edition, Tata McGrawHill Publishing Company Limited, Indian Reprint 2009.

5. Vikram Vaswani, "MYSQL: The Complete Reference", 2nd Edition, Tata McGraw -Hill Publishing Company Limited, Indian Reprint 2009

Student Activity:

1. Suggest list of open source softwares for the commercial software you come across

BCA III Year VI Semester

Cloud Computing

Objectives:

1. Discuss, with confidence, what is cloud computing and what are key security and control considerations within cloud computing environments.
2. Identify various cloud services.
3. Assess cloud characteristics and service attributes, for compliance with enterprise objectives.
4. Explain the four primary cloud category "types".
5. Evaluate various cloud delivery models.
6. Contrast the risks and benefits of implementing cloud computing.
7. Specify security threat exposure within a cloud computing infrastructure.
8. Recognize steps and processes used to perform an audit assessment of a cloud computing environment.

Course Outcome:

1)Explain the core concepts of the cloud computing paradigm: how and why this paradigm shift came about, the characteristics, advantages and challenges brought about by the various models and services in cloud computing

2)Apply the fundamental concepts in datacenters to understand the tradeoffs in power, efficiency and cost

3)Discuss system virtualization and outline its role in enabling the cloud computing system model

4)Illustrate the fundamental concepts of cloud storage and demonstrate their use in storage systems such as Amazon S3 and HDFS

5)Analyze various cloud programming models and apply them to solve problems on the cloud

Unit 1

Cloud Computing Overview – Origins of Cloud computing – Cloud components - Essential characteristics – On-demand self-service , Broad network access , Location independent resource pooling , Rapid elasticity , Measured service

Unit II

. Cloud scenarios – Benefits: scalability , simplicity , vendors ,security.

Limitations – Sensitive information - Application development – Security concerns - privacy concern with a third party - security level of third party - security benefits

Regularity issues: Government policies

Unit III

Cloud architecture: Cloud delivery model – SPI framework , SPI evolution , SPI vs. traditional IT Model

Software as a Service (SaaS): SaaS service providers – Google App Engine, Salesforce.com and google platfrom – Benefits – Operational benefits - Economic benefits – Evaluating SaaS

Platform as a Service (PaaS): PaaS service providers – Right Scale – Salesforce.com – Rackspace –

Force.com – Services and Benefits

Unit IV

Infrastructure as a Service (IaaS): IaaS service providers – Amazon EC2 , GoGrid – Microsoft soft implementation and support – Amazon EC service level agreement – Recent developments – Benefits

Cloud deployment model : Public clouds – Private clouds – Community clouds - Hybrid clouds - Advantages of Cloud computing

Unit V

Virtualization : Virtualization and cloud computing - Need of virtualization – cost , administration , fast deployment , reduce infrastructure cost - limitations
Types of hardware virtualization: Full virtualization - partial virtualization - para virtualization
Desktop virtualization: Software virtualization – Memory virtualization - Storage virtualization – Data virtualization – Network virtualization

Microsoft Implementation: Microsoft Hyper V – Vmware features and infrastructure – Virtual Box - Thin client

REFERENCES:

1. Cloud computing a practical approach - Anthony T.Velte , Toby J. Velte Robert Elsenpeter TATA McGraw- Hill , New Delhi – 2010
2. Cloud Computing: Web-Based Applications That Change the Way You Work and Collaborate Online - Michael Miller - Que 2008

Student Activity:

1.Prepare a list of companies that provide different cloud services

2, Create your own cloud using a local server

BCA III Year VI Semester

Elective- I

Hadoop & R Language

Course Objectives

- Apply Data Mining and understand Decision Trees and Random Forests
- Master the concepts of Hadoop 2.7 framework and its deployment in a cluster environment
- Learn to write complex MapReduce programs
- Perform Data Analytics using Pig & Hive
- Acquire in-depth understanding of Hadoop Ecosystem including Flume, Apache Oozie workflow scheduler, etc.
- Master advance concepts of Hadoop 2.7 : Hbase, Zookeeper, and Sqoop
- Get hands-on experience in setting up different configurations of Hadoop cluster
- Work on real-life industry based projects using Hadoop 2.7

Course Outcomes

Hadoop and R Language will prepare you to perform analytics and build models for real world data science problems. It is the world's most powerful programming language for statistical computing and graphics making it a must know language for the aspiring Data Scientists. 'R' wins strongly on Statistical Capability, Graphical capability, Cost and rich set of packages.

UNIT I

Introduction to BIG'Data '&' Hadoop Introduction to MapReduce '&' HDFS

UNIT II

The Hadoop MapReduce API & Algorithms. How to get started writing programs with Hadoop's API. Programming methodologies and paradigms in Map Reduce Beyond basics: The flow; APIs; Creating Input Formats and Output Formats; Driver; Mapper; Reducer; Streaming

UNIT III

Introduction to The'Hadoop'Ecosystem'Components An introduction to components surrounding Hadoop, which complete the greater ecosystem of available, processing tools.

UNIT IV

R over view, basic syntax, data types, variable, operators, decision making, loops, functions

UNIT V

String, vectors, list, matrices, data frames, reshaping, packages, graphics.

References:

Hadoop: The Definitive Guide By: Tom White Hadoop in Practice (By: Alex Holmes)

Hadoop Operations (By: Eric Sammer) Instant MapReduce Patterns - Hadoop Essentials How-to (By: Srinath Perera)

An Introduction to R: A Programming Environment for Data Analysis and Graphics **Author(s)** William N Venables, David M Smith.

The Art of R Programming: A Tour of Statistical Software Design Author(s) Norman Matloff

<div align="center">

BCA III Year VI Semester

Elective- I

NETWORK PROGRAMMING

</div>

Course Objectives

1. To understand inter-process and inter-system communication
2. To understand socket programming in its entirety
3. To understand usage of TCP/UDP / Raw sockets
4. To understand how to build network applications

Course Outcomes

1. Analyze the security requirements of a networked programming environment and identify the issues to be solved;
2. come up with conceptual solutions to those issues;
3. implement a programming solution;
4. understand the key protocols that support the Internet;
5. be familiar with several common programming interfaces for network communication;
6. have a detailed knowledge of the TCP/UDP Sockets

UNIT I

INTRODUCTION : Overview of UNIX OS - Environment of a UNIX process - Process control – Process relationships Signals – Inter-process Communication- overview of TCP/IP protocols

UNIT II

ELEMENTARY TCP SOCKETS: Introduction to Socket Programming –Introduction to Sockets – Socket address Structures – Byte ordering functions – address conversion functions – Elementary TCP Sockets – socket, connect,
bind, listen, accept, read, write , close functions – Iterative Server – Concurrent Server.

UNIT III

APPLICATION DEVELOPMENT:TCP Echo Server – TCP Echo Client – Posix Signal handling – Server with multiple clients –boundary conditions: Server process Crashes, Server host Crashes, Server Crashes and reboots,Server Shutdown – I/O multiplexing – I/O Models – select function – shutdown function – TCP echo Server (with multiplexing) – poll function – TCP echo Client (with Multiplexing)

UNIT IV

SOCKET OPTIONS, ELEMENTARY UDP SOCKETS: Socket options – getsocket and setsocket functions – generic socket options – IP socketoptions –ICMP socket options – TCP socket options – Elementary UDP sockets – UDP echo Server – UDP echo Client – Multiplexing TCP and UDP sockets – Domain name system – gethostbyname function.

UNIT V

ADVANCED SOCKETS: Ipv4 and Ipv6 interoperability – threaded servers – thread creation and termination – TCP echo server using threads – Mutexes – condition variables – raw sockets – raw socket creation – raw socket output – raw socket input – ping program – trace route program.

REFERENCES:
1. W. Richard Stevens, B. Fenner, A.M. Rudoff, "Unix Network Programming – The Sockets Networking API", 3rd edition, Pearson, 2004.
2. W. Richard Stevens, S.A Rago, "Programming in the Unix environment", 2nd edition, Pearson,
2005.

<div align="center">

BCA III Year VI Semester

Elective- I

Cyber laws

</div>

Course Objectives: The objectives of this course is to:

1. Enable learners to understand, explore, and acquire a critical understanding of Cyber Law
2. Develop competencies for dealing with frauds and deceptions (confidence tricks, scams) and other cyber crimes for example, child pornography etc. that are taking place via the Internet.
3. Make learners conversant with the social and intellectual property issues emerging from 'Cyberspace'.
4. Explore the legal and policy developments in various countries to regulate Cyberspace;
5. Develop the understanding of relationship between commerce and cyberspace; and give learners in depth knowledge of Information Technology Act and legal frame work of Right to Privacy, Data Security and Data Protection.

Course outcomes

At the end of the course, students should be able to:

1. Critically evaluate ongoing developments in law relating to information technologies
2. Display an understanding of how these developments relate to one another.
3. Examine areas of doctrinal and political debate surrounding rules and theories;
4. Evaluate those rules and theories in terms of internal coherence and practical outcomes;
5. Draw on the analysis and evaluation contained in primary and secondary sources

Unit I

Introduction: Computers and its Impact in Society, Overview of Computer and Web Technology, Need for Cyber Law, *Cyber Jurisprudence* at International and Indian Level.

Unit II

Cyber Law- International Perspectives: UN &International Telecommunication Union (ITU)Initiatives, Council of Europe -Budapest Convention on Cybercrime, Asia-Pacific Economic Cooperation(APEC), Organization for Economic Co-operation and Development(OECD), World Bank, Commonwealth of Nations.

Unit III

Constitutional& Human Rights Issues in Cyberspace: Freedom of Speech and Expression in Cyberspace, Right to Access Cyberspace – Access to Internet, Right to Privacy, Right to Data Protection.

Unit IV

Cyber Crimes& Legal Framework: Cyber Crimes against Individuals, Institution and State, Hacking, Digital Forgery, Cyber Stalking/Harassment, Cyber Pornography, Identity Theft &Fraud, Cyber terrorism, Cyber Defamation, Different offences under IT Act, 2000.

Unit V

Cyber Torts: Cyber Defamation, Different Types of Civil Wrong sunder the IT Act, 2000, Intellectual Property Issues in Cyber Space, Interface with Copyright Law, Interface with Patent Law, Trade marks & Domain Names Related issues

Reference Books

1. ChrisReed&John Angel,*Computer Law*, OUP,NewYork,(2007).

2. JusticeYatindraSingh,*Cyber Laws*, UniversalLaw PublishingCo, New Delhi, (2012).

3. VermaK,MittalRaman,*LegalDimensionsofCyberSpace*,IndianLawInstitute,NewDelhi,(2004)

4. Jonthan Rosenoer,*Cyber Law*, Springer,New York, (1997).

5. SudhirNaib,TheInformationTechnologyAct,2005:AHandbook,OUP,NewYork,(2011)

6. S.R.Bhansali, *Information TechnologyAct,2000*,UniversityBookHousePvt.Ltd.,Jaipur (2003).

7. Vasu Deva,*Cyber Crimes and Law Enforcement*,CommonwealthPublishers,New Delhi,(2003).

Elective- II

Advanced Android

Course Objective

The objective is to help learners to create applications using Google's Android open-source platform. The course explains what Android is and how it compares to other mobile environments, the setup of the Android™ Eclipse-based development tools, the Android SDK, all essential features, as well as the advanced capabilities and APIs such as background services, accelerometers, graphics, and GPS

Course Outcomes

1. Build your own Android apps
2. Explain the differences between Android and other mobile development environments
3. Understand how Android™ applications work, their life cycle, manifest, Intents, and using external resources
4. Design and develop useful Android applications with compelling user interfaces by using, extending, and creating your own layouts and Views and using Menus.
5. Take advantage of Android's APIs for data storage, retrieval, user preferences, files, databases, and content providers
6. Tap into location-based services, geo-coder, compass sensors, and create rich map-based applications
7. Utilize the power of background services, threads, and notifications.
8. Use Android's communication APIs for SMS, telephony, network management, and internet resources (HTTP).
9. Secure, tune, package, and deploy Android applications

Unit-I

Data Persistence: User Preferences, Persisting Data to Files, Using SQLite Databases

Unit-II
Messaging: SMS Messaging, Sending E-mail

Unit-III

Location-Based Services: Displaying Maps, Getting Location Data, Monitoring a Location, Building a Location Tracker

Unit-IV

Android Services: Create your Own Service, Communication between Services and Activity, Binding Activities to Services, Threading.

Unit-V

Exception Handling in Android: Handling Errors, Handling Exceptions Using Try, Catch and Finally
Publishing Android Application: Prepare for Publishing; Deploy APK Files, Publishing on the Android Market

Reference Books:

1.Android Programming: The Big Nerd Ranch Guide (Big Nerd Ranch Guides) By: Bill Philips
& Brian Hardy

2.Android Design Patterns: Interaction design solutions for developers by Greg Nudelman

3.Android User Interface Design: Turning Ideas and Sketches into Beautifully Designed Apps
By: Ian G. Clifton

4. Android Recipes: A Problem-Solution Approach By: Dave Smith & Jeff Friesen

5. Hello, Android: Introducing Google's Mobile Development Platform (Pragmatic
Programmers) By: Ed Burnette

6. Beginning Android Games By: Mario Zechner

7.Programming Android By: Zigurd Mednieks, Laird Dornin, G. Blake Meike & Masumi
Nakamura

BCA III Year VI Semester

Elective- II

Design of Video Games

Course Objectives

1. Discuss and define the terms and principles of game design and development.
2. Select and evaluate programming and scripting languages to develop particular games.
3. Define the structure and duties of the game development team.
4. Practice animation production and creation tools.
5. Apply the mathematics used in game design.
6. Apply the physics needed to design computer games.
7. Apply artificial intelligence to developing computer games.

Course Outcomes

After completing this course, students will be able to understand all game development problems and issues, such as story creation, selection of programming language, mathematical analysis, physical analysis, graphics, multimedia, artificial intelligence, and others.

UNIT I

History of video games, game genres, The games industry, Theory of funativity: what is fun?

UNIT II

Game design teams and processes, Level design, Modeling

UNIT III

Human-computer interaction (HCI) & interface design, Computer graphics, collision detection, lighting, and animation

UNIT IV

Game scripting and programming, Game data structures and algorithms

UNIT V

Artificial intelligence, Play testing

Reference Books

1. Introduction to Game Development Edited by: Steve Rabin ISBN: 1- 58450-377-7 Charles River Media, May 2005.

2.Game Development Essentials: An Introduction. 3rd Edition

3. *A Theory of Fun for Game Design*, by Koster

4. <u>*Understanding Comics: The Invisible Art*</u>, by McCloud

BCA III Year VI Semester

Elective- II

Advanced software Testing

Course Objectives

The Objective of this course is to learn and apply basic skills needed to create and automate the test plan of a software project., to know how to plan, develop, and execute an automated test plan. Students should learn testing concepts, Test planning, Creating a test plan in Test Director , Breaking the test plan into manageable components , Designing test cases and test steps, Analyzing the test plan, Developing Win Runner automated test scripts, Creating a script through recording, Synchronizing the test, Adding verification of GUI objects, bitmaps and text, and Managing the GUI map

Course Outcomes:

1 To be able to apply various test processes and continuous quality improvement

2 To be able to define the types of errors and fault models

3 To be able to use methods of test generation from requirements

4 To be able to use UML.

5 To be able to Test generation from FSM models

Unit-I

Basic Aspects of Software Testing: Testing in the Software Life Cycle, Product Paradigms, Metrics and Measurement

Unit-II

Testing Processes: Processes in General, Test Planning and Control, Test Analysis and Design, Test Implementation and Execution, Evaluating Exit Criteria and Reporting, Test Closure

Unit-III

Test Management: Business Value of Testing, Test Management Documentation, Test Estimation, Test Progress Monitoring and Control, Testing and Risk,

Unit-IV

Test Techniques: Specification-Based Techniques, Structure-Based Techniques, Defect-Based Techniques, Experience-Based Testing Techniques, Static Analysis, Dynamic Analysis, Choosing Testing Techniques,

Unit-V

Testing of Software Characteristics: Quality Attributes for Test Analysts,. Quality Attributes for Technical Test Analysts.

References:
1. Guide to Advanced Software Testing by Anne Mette Jonassen Hass

PROJECT & VIVA-VOCE

The objective of the project is to motivate them to work in emerging/latest technologies, help the students to develop ability, to apply theoretical and practical tools/techniques to solve real life problems related to industry, academic institutions and research laboratories.

The project is of 2 hours/week for one (semester VI) semester duration and a student is expected to do planning, analyzing, designing, coding, and implementing the project. The initiation of project should be with the project proposal. The synopsis approval will be given by the project guides.

The project proposal should include the following:
- Title

- Objectives

- Input and output

- Details of modules and process logic

- Limitations of the project

- Tools/platforms, Languages to be used

- Scope of future application

The Project work should be either an individual one or a group of not more than three members and submit a project report at the end of the semester. The students shall defend their dissertation in front of experts during viva-voce examinations.