Department of Computer Science

Proposed Syllabus for

B.Sc. (Honors) Computer Science

2013

BSC (Honors) – 2013

Semester I

Paper Code	Paper	Hours per Week	Marks	Credits
CSC131	Computer Fundamentals and Programming using C	04	100	03
CSC151	C Programming Lab	02	50	01
Total		**06**	**150**	**04**

Semester II

Paper Code	Paper	Hours per Week	Marks	Credits
CSC231	Data structures and Operating Systems	04	100	03
CSC251	Data structures Lab	02	50	01
Total		**06**	**150**	**04**

Semester III

Paper Code	Paper	Hours per Week	Marks	Credits
CSC331	Data Base Management System and Software Engineering	04	100	03
CSC351	Oracle Lab	02	50	01
Total		**06**	**150**	**04**

Semester IV

Paper Code	Paper	Hours per Week	Marks	Credits
CSC431	Object Oriented Programming using JAVA	04	100	03
CSC451	JAVA Programming Lab	02	50	01
Total		**06**	**150**	**04**

Semester V

Paper Code	Paper	Hours per Week	Marks	Credits
CSH531	Multimedia Applications	04	100	03
CSH532	Data Warehousing	04	100	03
CSH533	Design and Analysis of Algorithms	04	100	03
Elective I (Choose any one)				
CSH541A	WEB Technology	04	100	03
CSH541B	Python Programming			
CSH551	Multimedia Lab	04	100	02
CSH552	RDBMS Project Lab	08	200	04
CSH553A	WEB Technology Lab	04	100	02
CSH553B	Python Programming Lab			
Total		**32**	**800**	**20**

Semester VI

Paper Code	Paper	Hours per Week	Marks	Credits
CSH631	UNIX Programming	04	100	03
CSH632	Mobile Applications	04	100	03
CSH633	Computer Networks	04	100	03
Elective I (Choose any one)				
CSH641A	System Software	04	100	03
CSH641B	Computer Architecture			
CSH651	UNIX Programming Lab	04	100	02
CSH652	Mobile Applications Lab	04	100	02
CSH653	Project Lab	08	200	04
Total		**32**	**800**	**20**

CSC131 Digital Computer Fundamentals and Programming using C

Total teaching Hours/Semester: 60   *No of Lecture Hours/Week: 04*

I   COMPUTER FUNDAMENTALS

Objective
This Subject provides the basic concepts and the functional aspects of a computer. It is starts with the introduction to computers, working principle of a computer and the programming languages. It focuses on different number systems and its applications in computers. The Boolean algebra and different types gate networks, which gives a clear input to the students to understand the basic concepts of flip-flops, combinational logic, registers and counters.

Learning outcome
- To know the working principle of a computer.
- Ability to use Boolean algebra in different number systems and performing computations.
- Simplification of Boolean algebraic expressions.
- Ability to design efficient combinational and sequential logic circuits and implementation.

Unit I.   **(07)**
Introduction to Computers
Characteristics of Computers, Block Diagram of a Digital Computer
Number systems
Different number systems and their conversions (Decimal, Binary, Octal and Hexadecimal) Binary arithmetic - Addition, subtraction, multiplication and division of binary numbers, 1's and 2's complement, Floating point numbers, Coding – BCD, Gray, ASCII and EBCDIC.

Unit II.   **(06)**
Boolean Algebra
Boolean operations and expressions, Laws and rules of boolean algebra, Demorgan's Theorem, Boolean expressions, Simplification of Boolean expression.

Unit III.   **(06)**
Logic Gates
AND gate, OR gate, NOT gate , NAND gate , NOR gate , X-OR gate , X-NOR gate, The universal property of NAND gate and NOR gate,Karnaugh map (SOP).

Unit IV.   **(05)**
Combinational logic
Adders (Half and Full), Decoder, Encoder, Multiplexer , De-Multiplexer (Introductory Concepts Only).

Unit V.   **(06)**
Flip-flops
Flip-flops- SR flip-flop, D flip-flop, JK flip-flop, Positive edge triggered flip flops, Master slave JK flip-flop, Introduction to Registers and Counters.

Text Book:
[1]   Floyd, Thomas L, *Digital Computer Fundamentals*, 10th Edition, Pearson International, 2009.

Reference Books:
[1] Malvino, Paul Albert, Leach, Donald P. Gautam Saha, *Digital Principles And Applications*, 7th Edition, TMH, 2010.
[2]   Bartee, Thomas C, *Digital Computer Fundamentals,* 6th Edition, TMH, 2010.

II   PROGRAMMING USING C

Objective
 To study about algorithms, flowcharts and programs, to solve problems through logical thinking and to learn programming using C.
Learning outcome
   - To clearly understand the logic of the problem
   - To analyze the given problem and write the algorithm, flowchart
   - To write structured C programs

Unit I.   (05)
Introduction To Programming
Types of Programming, Languages, Structured Programming, Algorithms and Flowcharts with Examples.

Unit II.   (07)
Introduction To C
History of C- Character set - Structure of a C program – Data types, constants, variables and keywords. Expressions – Statements – Operators – Arithmetic, Unary, Relational and logical, Assignment, Conditional. Library functions. Data Input and output – Single character input, getchar, getch,– Single character output - putchar, Formatted I/O (scanf, printf), gets and puts function.

Unit III.   (08)
Control Structures And Arrays
Branching: if, if … else , if…else ladder, switch. Looping: while, do…while, for, nested control structures, break, continue statement, goto statement.
Arrays: definition, processing, types - One and Two dimensional arrays. String, string operations (strlen(),strrev(),strcpy(),strcat(),strcmp()strstr()), arrays of strings.

Unit IV.   (05)
Functions
Functions: Definition, Accessing and prototyping, types of functions, passing arguments to functions, recursion, passing arrays to functions.

Unit V.   (05)
Pointers And Structure
Pointers: Fundamentals, Declaration, Operations on Pointers, Passing pointers to a function.
Structures : Definition, Processing a structure.

Text Book:
[1] Byron Gottfried, Jitender Chhabra, *Programming with C, 3rd* Edition, Tata McGraw-Hill, 2010.

Reference Books:
[1] Balagurusamy E, *Programming in ANSI C,* 4th Edition, Tata-McGraw-Hill, 2007.
[2] Deitel H M and Deitel P J, *C - How to Program*, 7th Edition, Prentice-Hall, 2012.
[3] Yashavant P. Kanetkar, *Let Us C*, 12th Edition, BPB Publications, 2012.
[4].Susant K Rout, *Cimple,* Tata-McGraw-Hill Publishing Company Ltd., 2008.

CSC151 C Programming Lab

Total Hours/Semester: 30   *No of Hours/Week: 02*

Guidelines:

- Follow Coding standards
- The output of the programs should be neatly formatted
- The programs should be user friendly and interactive
- Use comments wherever necessary

1. Print the size of all the data types with its modifiers supported by C and its range.

2. Convert Fahrenheit to Celsius.

3. Find the roots of a quadratic equation.

4. Accept three numbers and find the largest and second largest among them.

5. Count the number of vowels in a line of text.

6. Accept two numbers and perform various arithmetic operations (+, -, *, /) based on the symbol entered.

7. Print all prime numbers between any 2 given limits.

8. Print all the Armstrong numbers between any 2 given limits.

9. Check whether a string is a Palindrome.

10. Check whether a given matrix is an Identity matrix or not.

11. Perform matrix addition and subtraction.

12. Print first 'n' elements of the Fibonacci series.

13. Find the factorial of a number using recursion.

QUESTION PAPER PATTERN

Two questions will be selected by the examiners. Students have to write and execute both the programs.

CSC231 Data structures and Operating Systems

Total teaching Hours/Semester: 60   *No of Lecture Hours/Week: 04*

I   DATA STRUCTURES

Objective
Data Structure is considered as one of the fundamental paper towards a more comprehensive understanding of programming and application development. Student is expected to work towards a sound theoretical understanding of Data Structures.

Learning Outcome
- Understand the need for Data Structures when building application
- Ability to calculate and measure efficiency of code
- Improve programming skills

Unit I.   **(05)**
Analysis of Algorithms: Introduction, What to Count and Consider, Rates of Growth, Sequential Search Analysis.
Arrays: Introduction, Array Operations, Merging Arrays, 2D Arrays, Matrix Operations.

Unit II.   **(07)**
Linked List: Introduction, Pointers, Insertion, Deletion, Searching, Double Linked List Representations.

Unit III.   **(06)**
Stack & Queue: Introduction, Stack Operations using Pointers, Infix to Prefix, Queue Operations using array.

Unit IV.   **(05)**
Trees: Introduction, Binary Trees, Properties of Binary Trees, Binary Tree Representations, Binary Tree Traversals.

Unit V.   **(07)**
Searching and Sorting: Introduction, Linear Search, Bubble Sort, Binary Search, Insertion Sort, Selection Sort.

Text Book:
[1] Yashwant Kanetkar, *Data Structures Through C,* BPB Publication, 2010.
Reference Books:
[1] Horowitz Sahni Anderson-Freed, *Fundamental of Data Structures in C,* Universities Press, Reprint 2009.
[2] Seymour Lipschultz: *Data Structures,* Schaum series TMH, 2010.

II   OPERATING SYSTEMS

Objective
 To acquire the fundamental knowledge of the operating system architecture and components
 and to know the various operations performed by the operating system.

Learning outcome
Upon completion of the course students will be able to:
- Understand the basic working process of an operating system.
- Understand the importance of process and scheduling.
- Understand the issues in synchronization and memory management.

Unit I.   (05)
Introduction and System Structures
Operating system definition, computer system organization, and architecture, structure and operations, process, memory and storage management.

Unit II.   (08)
Process Management
Process concepts, scheduling and operations on processes. Process Scheduling: Basic concepts, scheduling criteria, scheduling algorithms, Synchronization: Background, critical section problems

Unit III.   (05)
Deadlock
Deadlock: System model, deadlock characterization, methods for handling deadlock, deadlock prevention, avoidance and detection.

Unit IV.   (06)
Memory Management
Memory Management Strategies: Background, swapping, Memory allocation, Paging, Structure of the page table.

Unit V.   (06)
File system
File system: File system structure, directory structure, allocation methods and free-space management. Disk structure, disk scheduling and management.

Text Book:
[1] A. Silberschatz, P.B. Galvin and G. Gagne, *Operating System Concepts,* 8th Edition, New
     Delhi: Wiley India, 2011.
Reference Books:
[1] Stalling William, *Operating Systems: Internals and Design Principles*, 7th Edition, Prentice
     Hall, 2011.
[2] Dietel et al, *Operating Systems,* 3rd Edition, Pearson Education, 2004.
[3] A.S. Tanenbaum, *Modern Operating Systems*, 3rd Edition, Prentice Hall, 2007.

CSC251 Data structures Lab

Total Hours/Semester: 30   *No of Hours/Week: 02*

1. Inserting an element into one dimensional array

2. Deletion of an element in a one dimensional array

3. Implementation of Linear Search

4. Implementation of Binary Search

5. Creation of a linked list and inserting elements into it.

6. Deletion from a linked list.

7. Implementation of different operations on a stack using linked list

8. Implementation of different operations on a queue using an array

9. Implementation of insertion sort

10. Implementation of selection sort.

QUESTION PAPER PATTERN

Two questions will be selected by the examiners. Students have to write and execute both the programs.

CSC331 Database Management System and Software Engineering

Total teaching Hours/Semester: 60   *No of Lecture Hours/Week: 04*

I   DATABASE MANAGEMENT SYSTEM

Objective
To provide strong foundation of database concepts and to introduce students to application development in DBMS.

Learning outcome
- Understanding the fundamentals of RDBMS.
- Building foundation for implementation through project work.

Unit I.   (07)
Introduction
Data, Database, Database management system, Characteristics of the database approach, Role of Database administrators, Role of Database Designers, End Users, Advantages of Using a DBMS and When not to use a DBMS.
DBMS Architecture
Data Models – Categories of data models, Schemas, Instances, and Database states. DBMS Architecture and Data Independence – The Three schema architecture, Data independence. DBMS Languages and Interfaces. Classifications of Database Management Systems.

Unit II.   (06)
Data Modelling Using Entity-Relationship Model
Using High Level Conceptual Data Models for Database Design, Example Database applications. Entity types, Entity Sets, Attributes and Keys. Relationships, Relationship types, Roles and Structural constraints. Weak Entity Types and Drawing E- R Diagrams

Unit III.   (05)
Database Design
Functional dependencies and Normalization for Relational Databases - Normalization concepts, first, second, third normal forms

Unit IV.   (07)
SQL
SQL data definition and data types, specifying constraints in SQL, schema change statements, Basic queries, INSERT, DELETE and UPDATE statements in SQL, Views – Concept of a view in SQL.

Unit V.   **(05)**
Transaction Processing Concepts and Concurrency Control Techniques
Transaction and System concepts – Desirable properties of Transactions – Schedules and Recoverability. Lock-Based Protocols – Locks, Granting of Locks, and Two phase locking protocol.

Text Book:
[1] Elmasri & Navathe, *Fundamentals of Database Systems,* 5th Edition, Addison – Wesley, 2007.

Reference Books:
[1] O`neil Patric and O`neil Elizabeth, *Database Principles Programming and Performance,* 2nd Edition, Margon Kaufmann Publishers Inc., 2001.
[2] Silberschatz, Korth, Sudarshan, *Database System Concepts*, 5th Edition, McGraw Hill, 2006.

II   SOFTWARE ENGINEERING

Objective

To make the students aware of different software engineering principles. To prepare the students to develop the skills necessary to handle software projects. To make the students aware of the importance of software engineering principles in designing software projects.

Learning Outcome

On completion of the course the student shall

- Understand the importance of the stages in the software life cycle.
- Understand the various process models.
- Understand the importance of Software testing.

Unit I.   (05)
Software and Software Engineering
Nature of software- Defining software, Software Application Domains, Legacy Software-, Software Engineering, The software process, Software Engineering practice – The essence of Practice, General Principles -, Software Myths.

Unit II.   (06)
Process models
A generic process model – Defining a framework activity, identifying a Task Set, Process Patterns -, Process Assessment and improvement, Prescriptive Process Models – The waterfall Model, Incremental Model, Evolutionary Process Model, Concurrent Models-, A Final Word on Evolutionary Processes.

Unit III.   (06)
Understanding Requirements
Requirements Engineering, Establishing the groundwork – Identifying Stakeholders, Recognizing multiple viewpoints, Working toward Collaboration, Asking the first questions-, Eliciting requirements – Collaborative requirement gathering, Quality function Deployment, Usage Scenario Elicitation Work Products - , Developing use cases, building the requirements model – Elements of the requirements Model, Analysis pattern -, Negotiating requirements, validating requirements.

Unit IV.   (06)
Design Concepts
The design within the context of Software Engineering, The design process – Software quality guidelines and attributes, The evolution of software design -, Design concepts – Abstraction, Architecture, Patterns, Separation of concerns, Modularity, information hiding, Functional Independence, refinement, Aspects, Refactoring, Object Oriented design concepts Design classes. The design Model – Data Design elements, Architectural Design elements, Interface Design Elements, Component-Level Design elements, Deployment level Design elements.

Unit V.   **(07)**
Software Testing
A Strategic approach to testing- Verification and Validation, Organizing for software testing, software testing strategy, Criteria for completion of testing-, Test strategies for conventional software – Unit testing, Integration testing- , Test strategies for Object Oriented software-Unit testing in the OO Context, Integration testing in the OO Context-, Validation testing, White-box testing, Basic path testing- Flow Graph Notation, Independent program paths, Deriving test cases, Graph matrices-, control structure testing – Condition testing, Data flow testing, loop testing-, Black-box testing-Graph-based testing methods, Equivalence partitioning, boundary value analysis.

Text Book:
[1] Pressman S Roger, *Software Engineering A Practitioner's Approach*, McGraw Hill
    International Editions, 7th edition, 2010.

Reference Books:
[1] Sommerville, Ian, *Software Engineering*, Addison Wesley, 9th Edition, 2010.
[2] Rumbaugh, James, *Object Oriented Modeling and design*, Pearson Education, New Delhi,
    2005.

CSC351 Oracle Lab

Total Hours/Semester: 30   *No of Hours/Week: 02*

Objective
The primary objective of this course is to understand relational database concepts and design by using SQL.
Learning outcome
Upon successful completion of the course students will be able to
  • Design and implement programming logic for a relational database.
  • Manipulate data stored in an Oracle DBMS using Oracle SQL.

1. SQL*Plus and SQL   **(04)**
   a. Introduction
   b. Logging on to SQL*Plus and Leaving SQL*Plus
   c. Choosing and Describing Tables
   d. Elements of the SQL Query
   e. Editing SQL Statements
   f. The System Dummy Table
   g. Selecting Columns
   h. Duplicate Information (DISTINCT)
   i. Sorting Information

2. SQL Functions   **(04)**
   a. The Concatenation Operator
   b. Column Aliases
   c. String Functions
   d. Arithmetic Functions
   e. Date Functions

3. Advanced SQL Functions   **(02)**
   a. Select with Minus, Union and Intersect
   b. Handling NULL

4. Filtering Data Using Where   **(02)**
   a. Where Operators
   b. Where with Keywords and Logical Operators

5. Group By and Group By Functions   **(02)**
   a. Group Function Examples
   b. Group Function with Having

6. Data Definition Language (DDL)   **(04)**
   a. Create, Drop Alter Keywords
   b. Tables
   c. Column
   d. Views
   e. Object
   f. Alter table

7. Data Manipulation Language (DML)   **(02)**
   a. Insert
   b. Update
   c. Delete

8. Integrity Constraints   **(02)**
   a. Types of Constraint
   b. Referential Integrity
   c. Defining Constraints

9. Retrieving Data from Multiple Tables   **(04)**
   a. Joining Tables (Equi-Joins, Non-Equi-Joins)
   b. Aliases for Table Names

10. Sub-Queries   **(04)**
    a. Basic Sub queries
    b. Multiple Column Sub queries
    c. Sub queries with Having

CSC431 Object Oriented Programming Using JAVA

Total teaching Hours/Semester: 60   *No of Lecture Hours/Week: 04*

Objective
The objective of this subject to expose to the students the introduction to OOPs and advantages of object oriented programming. The concepts of OOPs make it easy to represent real world entities.

Learning outcome
- The students are introduced to basics of object oriented programming.
- The students are exposed to have the competence in the use of Java Programming language in the development of small to medium sized application programs that demonstrate professionally acceptable coding.

Unit I.   **(12)**
Introduction to OOPs
Problems in Procedure Oriented Approach-Features of Object Oriented Programming System-Class / Object-Encapsulation-Abstraction-Inheritance - Polymorphism. Benefits of OOPs, Applications of OOPs.
Introduction to Java
History-Features of Java-Java and the Internet-How Java differs from C and C++-Java Environment-Structure of Java Program-Java Virtual Machine-Data Types-Constants-Variables-Declaration of variables-Giving values to variables-Scope of variables-Symbolic constants-Literals. Operators-Arithmetic, Boolean logical, Relational and Bitwise operators-Operator Precedence.

Unit II.   **(12)**
Classes and String Handling
Classes and Objects - General form of a class-Declaring objects-Accessing class members-Constructors-Parameterized constructors-Overloading constructors-Defining methods-Overloading methods-Returning a value-Recursion-Introducing Access Control-Understanding static-Introducing Final-Garbage collection-finalize() method-this keyword.
Array
Introduction to Arrays, One Dimensional Arrays, Creation of Arrays, Array Intialization, Multidimensional Arrays, arrayname.length, Command Line Arguments.

Unit III.   **(12)**
Inheritance And Exception Handling
Basics-Member Access and Inheritance- Super class variable referring to a sub class-Applications of keyword super- Creating a Multilevel Hierarchy-Order of calling constructors-Method Overriding-Dynamic method dispatch-Abstract classes-Using final with Inheritance. Defining an Interface - Implementing interfaces-Variables in interfaces-Extending interfaces.

Exception

Introduction-Types of errors –Exception-Uncaught Exceptions - try and catch - Multiple catch - Nested Try - throw, throws and finally-Built-in Exceptions-User Defined Exception.

Unit IV.
Multithreading And Io Package   **(12)**
Multithreaded Programming – Creating Threads-Life cycle of a Thread- Thread Priorities-Synchronization-Inter thread communication-Deadlocks of thread-Avoiding deadlocks in a program –Daemon Threads-Application of Threads.
Io Package
I/O Basics-Streams-Byte Streams-Input Stream classes-Output Stream Classes, Character Streams-Reader Stream classes-Writer Stream classes.

Unit V.   **(12)**
Applets
Applet basics-How Applet differs from Applications-Applet Architecture-Applet life cycle-Applet display methods-Repaint-Status window-passing parameters to applets-getDocumentBase() and getCodeBase()-AppletContext and showDocument().
Event Handling-Event handling mechanisms-Delegation Event Model-Event classes-Sources of events-Event listener interfaces-Handling mouse and keyboard events-Adapter classes-Inner classes.

Text Book:
[1] Schildt Herbert, *Java: The Complete Reference*, 8th Edition*,* Tata McGraw- Hill, 2011.

Reference Books:
[1] E. Balagurusamy, *Programming with JAVA a Primer*, 4th Edition, Tata McGraw-Hill
    Publishing Company Limited, Delhi, 2010.
[2] Dr.Rao, Nageswara , *Core Java: An Integrated Approach*, Kongent Solutions Inc, 2009.

CSC451 JAVA Programming Lab

Total Hours/Semester: 30   *No of Hours/Week: 02*

Guidelines
- The output of the programs should be neatly formatted.
- Use meaningful variable names

1. Demonstration the use of methods of Math class.

2. Implementation of various methods of String class

3. Implementation of the concept of array

4. Demonstrate the concept of command line arguments

5. Demonstrate inheritance

6. Implementation of super and this keyword

7. Demonstration of static variables and methods

8. Implementation of interface.

9. Implementation of Abstract class

10. Implementation of Exception Handling

11. Implementation of multithreading by extending Thread class

12. Demonstration of FileInputStream and FileOutput Stream Classes

13. Creation of applets and passing parameters to applets

14. Implementation of Mouse and Keyboard events in an applet.

QUESTION PAPER PATTERN

Two questions will be selected by the examiners. Students have to write and execute both the programs.

CSH531 Multimedia Applications

Total teaching Hours/Semester: 60   *No of Lecture Hours/Week: 04*

Objective
This course teaches students how to fully utilize the latest image editing tool and animation tool to create and manipulate images and animation. In this course students learn to paint and retouch images, use layers, support video, work with vector tools, manage digital assets, work with RAW camera files, manage color, and prepare images for output to the web, Prepare 2D animation, interactive presentation, slideshows and Games and Enables the student to think creatively.

Learning outcome
At the end of the course students will

- Through with Design and color Concepts
- Prepare Images for Print and Web, Create Layouts for web pages, Paper Adverts, CD covers, Posters, brochure Package Designing, take necessary steps to fix colour issues. Converting B/W photo to color.
- Creating Interactive slide Shows, Interactive high graphic Presentations and 2D animation.

Unit I.   **(08)**
Concept of Graphics and Multimedia
Graphics, Vector Graphics, Raster Graphics, Multimedia Definition, Applications of Multimedia, Advantages of Multimedia, Different types of Media's -Text, Graphics, Image, Animation, Audio, Video, Special Effects, compositing, Morphing, Virtual Reality and its applications.
Color theory, typography and File formats- Color Theory, Color Modes, Concept of Digital Images, Image compression, Different File formats (BMP, JPEG, GIF, TIFF, PDF, PSD, WBMP …etc) Image for Print and WEB Resolutions, Monitor Configurations, Graphic Card.

Unit II.   **(12)**
Getting to Know the work area
Staring to work with Adobe Photoshop, Using tools, option bar and other panels, undoing actions in Photoshop, customizing the workspace, finding resources for using Photoshop.
Strategy for retouching, resolution and Image size, straightening and cropping the image, replacing the colors in image, Brush tools and options, adjusting saturation using Sponge tool, repairing with Clone Stamp, spot healing brush tool, applying content-aware patch, unsharp mask, saving image for color printing.
About selecting and selection tools, Quick selection tool, Moving selected area, manipulating selections, Magic wand, lasso tools, cropping and erasing with selection, saving and retrival of selection.

Unit III.   **(12)**
Correcting and Enhancing Digital Photographs
About layers, layers panel , Rearranging layers, applying gradient to a layer, applying a layer stylet, adding an adjustment layer, updating layer effects, adding a border, flattening and saving files.

About camera raw, processing files in camera raw, correcting digital photographs in Photoshop, correcting image distortion, adding depth of field.
Working with Masks and channels, creating mask, refining mask, creating quick mask, puppet warp, working with channels. About Type, Creating clipping mask from type, creating type on a path, warping point type, designing paragraphs of type.

Unit IV.   (12)
Advanced compositing and Vector Drawing Techniques
Paths and Pen tool, paths with artwork, creating vector objects for background, working with defined custom shapes, importing smart objects, applying filters, hand-coloring selections on a layer, applying smart filters, drop shadow and a border, color schemes , automating multistep task, stitching panorama, finishing image for Print and web.
Editing Video, creating video projects, animating text with key frames, creating effects, transitions, adding and muting audio, rendering video.Painting with mixer brush, Selecting brush settings, mixing colors, creating custom brush preset, mixing color with a photograph.
Creating slices, exporting HTML and images, using zoomify feature, creating a web gallery. Preparing for PRINT – specifying color Management settings, proofing an image, identifying out-of-gamut colors, saving as CMYK EPS files, printing.

Unit V.   (08)
Introduction to Flash
Getting to know the workspace, working with library, understanding timeline, using property inspector, using panel, using tools panel, previewing your movie, modifying with content and stage, saving and publishing.
Working with Graphics-Understanding strokes and fills, creating shapes, making selections, editing shapes, using gradients and bitmap fills, making patters and decorations, creating curves, creating and editing text.
Creating and editing symbols-Importing illustrator files, about symbols, creating symbols, importing Photoshop files, editing and managing symbols, changing the size and position of instance, understanding blend effects, applying filters and special effects.

Unit VI.   (08)
Adding Animation and Interactivity
About animation, understanding project file, animating position, changing the pacing and timing, All types of tweening, animating filters, animating transformations, swapping tween targets, nested animations, motion blur, easing, animating 3D motion, testing movie.
Creating buttons, understanding action Script 3.0, preparing the timeline, adding stop action, creating event handlers for buttons, creating destination key frames, home button with code snippets, code snippets option, and animated buttons.
Adding sound and video content to flash project , encoding video files , working with text tools , hyper links. Publishing flash documents, understanding publishing, publishing desktop application, publishing for a mobile device.

Text Books:
[1] Lisa DanaeDayley, Brad Dayley, *Adobe Photoshop CS6 Bible*, John Wiley & sons, 2012.
[2] *Adobe Flash Professional CS6*, Adobe Classroom in a Book- The official training workbook from Adobe Systems, Adobe Press, 2012.

Reference Books:
[1] Lesa Snider, *Photoshop CS6: The Missing Manual*, O'Reilly Media Inc, 1st Edition, 2012.
[2] Adobe Photoshop CS6, Adobe Classroom in a Book- The official training workbook from Adobe Systems, Adobe Press, 2012.
[3] Chris Grover, Flash CS6: The Missin*g Manual,* O'Reilly Media Inc, 1st Edition, 2012.

CSH532 Data Warehousing

Total teaching Hours/Semester: 60   *No of Lecture Hours/Week: 04*

Objective
To Main objective is to give an overview of Data Warehousing system.
Learning outcome
Upon successful completion of the course, The students are able to understand to do the following:

- Discuss the role of data warehousing and enterprise intelligence in industry and government.
- Summaries the dominant data warehousing architectures and their support for quality attributes

Unit II.   **(10)**
Introduction to Data Warehousing: Data warehousing, framework of the Data warehouse, Data warehouse Options, developing data warehouses, goals.
Data Warehouse design consideration and Dimension Modeling: defining Dimensional Model, Granularity of Facts, additivity of facts, Functional Dependency of the Data.

Unit III.   **(10)**
Implementing changing Dimensions: Types of slowly changing dimensions, Implementing Rapidly changing Dimensions, Multi-use Dimensions.
Designing the Data Warehouse-common steps: Identifying the source, The ETL Process.

Unit IV.   **(10)**
Data Quality: Introduction to Data Quality, Magnitude of the Problem, Techniques and solution, components of a good Data quality solution, Data quality and CMM levels.
Metadata Management: Categorizing Metadata, Metadata value proposition, strategies, Metadata Management in practice, Meta data management reference architecture.

Unit V.   **(10)**
Data warehousing and OLAP: Defining OLAP, value of Multidimensional Data, OLAP Terminology, Understanding Multidimensional Data, Multidimensional Architectures, Multidimensional views of relational data, Physical Multidimensional Databases, Data Explosion and Integrated Relational OLAP.

Unit VI.   **(10)**
CRM and Data Warehousing: Introduction to CRM, Corporate philosophy, conceptual customer Model, Implications of Multi-channel/Multiple Customer touch point environment, implications of exponential growth of customer data, customer centric data warehouse architecture.

Unit VII. (10)
Introduction to Data Mining: Data mining- characterization and discrimination, mining frequent patterns, associations and correlations, classification and prediction , cluster analysis, outlier analysis, evolution analysis, interesting patterns , data mining task primitives, major issues in data mining.

Text Books:
[1] Soumendra Mohanty, *Data warehousing-Design, development and Best Practices,* 4th reprint, Tata-McGraw Hill, 2012.
[2] Han Jiawei and Micheline Kamber, *Data mining concepts and techniques,* 3rd Edition, Morgan Kaufmann Publishers, 2011.

Reference Books:
[1] Berson Alex, Stephen J Smith, *Data Warehousing, Data Mining &OLAP*, Tata-McGraw-hill Edition, 2008.
[2] Kimball Ralph, Reeves, Ross, Thronthwaite, *The Data warehouse lifecycle toolkit,* Wiley India   Edition, 2006.

CSH533 Design and Analysis of Algorithms

Total teaching Hours/Semester: 60   *No of Lecture Hours/Week: 04*

Objective
To introduce the classic algorithms in various domains and to study the different techniques for designing efficient algorithms.

Learning outcome
Upon successful completion of the course student will be able to
- Design efficient algorithms using the various approaches for real world problems.
- Analyze the running time of algorithms for problems in various domains.
- Apply the algorithms and design techniques to solve problems.

Unit I.   **(10)**
Introduction
A simple example of design using insertion sort, pseudo code for insertion sort and analysis of time complexity. Performance Analysis – Space complexity and Time complexity (posteriori testing, and a priori approach), Asymptotic Notations (O, Ω, Θ). Polynomial Vs. Exponential Algorithms. Average, Best and Worst case complexity.

Unit II.   **(12)**
Divide And Conquer Algorithms
 Introduction to Divide and Conquer Algorithms - Finding the Maximum and Minimum, Quick sort (Derivation of Average case analysis and Worst case analysis), Binary Search (Derivation of Average case analysis), and Strassen's Matrix Multiplication

Unit III.   **(12)**
Greedy Algorithms
Introduction to Greedy Algorithms - Fractional Knapsack problem, minimum cost spanning trees, Kruskal's and Prim's Algorithms, Optimal Merge patterns and Single-Source Shortest Paths

Unit IV.   **(08)**
Dynamic Programming
Definition - All-pairs shortest paths, Traveling salesman problem, optimal parameterization for product of sequence of matrices and Multistage graphs

Unit IV.   **(10)**
Back Tracking And Randomized Algorithms
Introduction - N-Queens, Sum of Subsets, Graph coloring, Hamiltonian Cycles problem using Back tracking algorithms. Random Number Generators and Primality Testing using randomized algorithms

Unit IV.   **(08)**
Graphs And Heaps And Lower Bound Theory
Definitions – Adjacency Matrix, Adjacency Lists. Breadth First Search and Traversal, Depth First Search and Traversal. Priority Queues using Heap and Design of Heap sort using Heap.

Text Book:
[1] Horowitz, Sahni, Rajasekaran, *Fundamentals of Computer Algorithms*, Universities Press Pvt Ltd, 2008.

Reference Books:
[1] Donald E. Knuth, *The Art of Computer Programming Volume 3 Sorting and Searching*, 1st Edition, Pearson Education, Addison-Wesley, 2011.
[2] Gav Pai, *Data structures and Algorithms*, Tata McGraw Hill, 2008.

CSH541A WEB Technology

Total teaching Hours/Semester: 60   *No of Lecture Hours/Week: 04*

Objective
Enable students to program for the World Wide Web using HTML, JavaScript, PHP and MySQL. Expose the students to the idea of Content Management Systems with JOOMLA.

Learning outcome
Upon successful completion of the course, the students would be able to
- Create static HTML pages
- Validate HTML FORM data using Javascript at the client side.
- Create dynamic web pages using PHP and MySQL
- Build a website using CMS.

Unit I.   **(10)**
HTML
Introduction to Hyper Text Markup Language, Key components of HTML document, HTML elements, Headers, Linking, Images, Unordered Lists, Nested and ordered Lists,

Unit II.   **(08)**
Tables And Forms
HTML Tables and Formatting, HTML Forms, Internal Linking, Creating and Using Image Maps, frames.

Unit III.   **(10)**
JavaScript
Browser and Document object , scripts and HTML Document, variables, expressions, Data type conversions, decisions and loops , control structure, windows Document object, forms and form handling elements, scripting, event handling.

Unit IV.   **(10)**
PHP
Essential PHP, Creating a simple PHP program, operators and Flow Control, strings and arrays, creating functions.

Unit V.   **(10)**
PHP & MySQL
Connecting to MySQL from PHP, PHP MySQL Connectivity, Creating Databases and Tables with PHP.

Unit VI.   **(12)**
JOOMLA
Installation, Administration interface, Global Configuration, User Manager, Media Manager, Site Management Tools, Site Map. Sections, Categories, Articles, and Menus. Creating and configuring menus. Installing and configuring Templates, Modules, Components, Plug-ins.

Text Books:

[1] Steven M. Schafer, *HTML, XHTML and CSS Bible*, 5th Edition, Wiley-Eastern Publishing Inc., 2011.

[2] Mercer W.Dave, Allan Kent, Steven D Nowichi, David Mercer, Dan Squier, Wankyer Choi *Beginning PHP5,* Wiley – Dreamtech India Pvt. Ltd., 2006.

[3] Jen Kramer „*Joomla! Start to Finish: How to Plan, Execute, and Maintain Your Web Site*, Wily-Eastern Publishing Inc., 2010.

Reference Books:

[1] Danny Goodman, *JavaScript Bible*, 3rd Edition, Wiley-Eastern Private Ltd., 2010.

[2] RicShreves, *Joomla! Bible*, Wiley Publishing Inc., 2010.

CSH541B Python Programming

Total teaching Hours/Semester: 60   *No of Lecture Hours/Week: 04*

Objective
Learn to program and programming paradigms brought in by Python with a focus on File Handling and Regular Expressions

Learning Outcome
- Able to walk through the algorithm
- Improve programming skills
- Appreciate Python Programming Paradigm
- Hands on Regular Expression
- Write to file handling scripts

Unit I.   **(10)**
Introduction and overview
Introduction, What is Python, Origin, Comparison, Comments, Operators, Variables and Assignment, Numbers, Strings, Lists and Tuples, Dictionaries, if Statement, while Loop, for Loop and the range() Built-in Function, Files and the open() Built-in Function, Errors and Exceptions, Functions, Classes, Modules.

Syntax and Style
Statements and Syntax, Variable Assignment, Identifiers, Basic Style Guidelines, Memory Management, Python Application Examples.

Unit II.   **(10)**
Python Objects
Python Objects, Standard Types, Other Built-in Types, Internal Types, Standard Type Operators, Standard Type Built-in Functions, Categorizing the Standard Types, Unsupported Types.

Numbers and Strings
Introduction to Numbers, Integers, Floating Point Real Numbers, Complex Numbers, Operators, Built-in Functions.
Sequences: Strings, Lists, and Tuples, Sequences, Strings, Strings and Operators, String-only Operators, Built-in Functions, String Built-in Methods, Special Features of Strings.

Unit III.   **(10)**
Lists
Operators, Built-in Functions, List Type Built-in Methods, Special Features of Lists, Tuples, Tuple Operators and Built-in Functions, Special Features of Tuples.

Dictionaries
Introduction to Dictionaries, Operators, Built-in Functions, Built-in Methods, Dictionary Keys.

Conditionals and Loops
if statement, else Statement, elif Statement, while Statement, for Statement, break Statement, continue Statement, pass Statement, else Statement.

Unit IV.   **(10)**
Files and Input/output
File Objects, File Built-in Function, File Built-in Methods, File Built-in Attributes, Standard Files, Command-line Arguments, File System, File Execution, Persistent Storage Modules.
Programming Exercise: Create Comma Separate Files (CSV), Load CSV files into internal Data Structure, Write script to work like a SELECT statement from a CSV file.

Unit V.   **(09)**
Regular Expressions
Introduction/Motivation, Special Symbols and Characters for REs, REs and Python.
Programming Exercise: Check for data error in CSV files: Numeric Check, Alphanumeric Check, Email Check, Date Check.

Unit VI.   **(11)**
Errors and Exceptions
Exceptions, Exceptions in Python, Detecting and Handling Exceptions, Exceptions as Strings, Raising Exceptions, Assertions, Standard Exceptions.
Functions
What Are Functions?, Calling Functions, Creating Functions, Passing Functions, Formal Arguments, Positional Arguments, Default Arguments, Why Default Arguments?, Default Function Object Argument Example, Variable-length Arguments, Non-keyword Variable Arguments (Tuple), Keyword Variable Arguments (Dictionary).

Text Book:
[1] Chun, J Wesley, *Core Python Programming, 2nd Edition,* Pearson, 2007, Reprint 2010.

Reference Books:
[1] Barry, Paul, *Head First Python,* 2nd Edition, O Rielly, 2010.
[2] Lutz, Mark, *Learning Python,* 4th Edition, O Rielly, 2009.

CSH551 Multimedia Lab

Total Hours/Semester: 60   *No of Hours/Week: 04*

1. Creating Business card with the given requirement (using text , filters and picture)

 2. Create logo, Letterhead, envelop design for a given company

3. Create an Invitation for Music concert Program

4. Creating Poster for a given company or program with all the required information

5. Creating Brochure for a new product which will be launched in the market soon

6. Create an advertisement for a company (use filters, layer effects, text effects, color gradient)

7. Retouch the old photo (convert a black & white photo into color)

8. Design a webpage with all the details given, with buttons, images and effects

9. Create a simple animation to simulate the traffic signal

10. Create non interactive and interactive slide show

11. Create an advertisement for mobile device

12. Create an interactive quiz program

13. Create an interactive game using action script

14. Create an interactive photo show album

15. Create an E-learning content (learning Movies, interactive learning for kids)

16. Create simulated program of calculator

QUESTION PAPER PATTERN

Two questions will be selected by the examiners. Students have to write and execute both the programs.

CSH552 RDBMS Project Lab

Total Hours/Semester: 120   *No of Hours/Week: 08*

The mini project is to introduce the student to the methodology for solving a problem and preparing a report using the steps of software engineering. Student should take a Database mini project in group of not more than 4 members and submit a report.

Guidelines:
1. The students should carry out the mini project during the allotted lab hours in the computer lab.
2. The project should be a database-oriented project, involving suitable front end and back end tools available in the college.
3. The guide can award internal marks by evaluating the performance of the students during the course of the project work.
4. The format of the project report shall be instructed by the faculty in-charge.

CSH553A WEB Technology Lab

Total Hours/Semester: 60   *No of Hours/Week: 04*

1. Create a HTML page that will have the following:
   Headers, Linking and Images.

2. Create a HTML page that will have the following:
   Frames, Unordered Lists, Nested and ordered Lists

3. Create a HTML page that will have the following:
   Tables and FORMS

4. FORM validation using Javascript

5. Event handing using Javascript

6. Displaying HTML content using PHP

7. FORM processing using PHP

8. FORM validation using PHP

9. Storing data in MYSQL using PHP

10. Retrieving data from MYSQL using PHP

11. Drill down reports using PHP

12. Build a sample site using JOOMLA by modifying a Template and integrating PHP scripts.

QUESTION PAPER PATTERN

Two questions will be selected by the examiners. Students have to write and execute both the programs.

CSH553B Python Programming Lab

Total Hours/Semester: 60   *No of Hours/Week: 04*

1. Create a calculator program

2. Explore String functions

3. Implement sequential search

4. Implement Selection sort.

5. Implement Stack.

6. Implement Queue.

7. Creating a CSV File based on user input

8. Reading a CSV File already created and check for a specific pattern

9. Implement SQL Select type program for a CSV file, should support where clause, select particular columns

10. Regular Expression to validate string, number, date and email

11. Create a mini project to explore an advanced topic in Python **(10 Hours)**

QUESTION PAPER PATTERN

Two questions will be selected by the examiners. Students have to write and execute both the programs.

CSH631 Computer Networks

Total teaching Hours/Semester: 60　　　　　　*No of Lecture Hours/Week: 04*

Objective
To study about network components, topologies and network models and the various network protocols and algorithms.

Learning outcome
Students will gain knowledge about
  - Networks
  - internal components its functionality.

Unit I.　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　(10)
Physical Layer
Data communications : components – Network criteria – physical structures – network models – categories of networks –interconnection of networks – inter network Protocols and standards : protocols-standards-standards organizations- internet standards Network models: Layered tasks – OSI model – layers in the OSI model – TCP/IP protocol suite.

Unit II.　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　(10)
Digital Transmission
Digital to digital conversion: Line coding – line coding schemes – block coding - analog to digital conversion – PCM - transmission modes: serial transmission – parallel transmission Analog Transmission: Digital to analog conversion: FSK-ASK-PSK Analog to Analog conversion: Amplitude modulation – Frequency modulation – phase modulation Multiplexing: Frequency division multiplexing – Time division multiplexing – Transmission Media Guided media: Twisted pair cable – coaxial cable – fiber optic cable Unguided media: radio waves - micro waves – infrared.

Unit III.　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　(10)
Data Link Layer
Error correction and detection: Introduction- block coding-linear block cods-cyclic codes-checksum. Data link Control: protocols-simplest protocol- stop and wait protocol- stop and wait automatic repeat request-go back n automatic repeat request-selective repeat-automatic repeat request-piggybacking. Multiple Access: Random access-Aloha-CSMA-CSMA/CD-CSMA/CA Controlled access: reservation, polling, token passing. Channelization: FDMA,TDMA, CDMA.

Unit IV.　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　(10)
Wired LANs
Ethernet: IEEE standards, standard Ethernet- fast Ethernet. Wireless Lans: IEEE 802.11-arhitecture-MAC sub layer addressing mechanism, physical layer-Bluetooth: architecture-Bluetooth layers-radio layer-baseband layer-L2CAP-other upper layers.
Network Layer: AddressingIPV4 addresses - IPV6 Addresses Internet Protocol: IPv4 –
IPv6 Address mapping protocols: ARP – RARP.

Unit V.   **(10)**
Routing Protocols
Unicast routing protocols: distance vector routing, Link State routing Multicast Routing protocols (Any two) Transport Layer: Process to process delivery – UDP – TCP Congestion control and QOS: Data traffic – congestion – congestion control – quality of service – techniques to improve quality of service.

Unit VI.   **(10)**
Application layer& Network Security
DNS: Name space – domain name space – distribution of name space Electronic mail Architecture – FILE transfer: FTP WWW and HTTP: Architecture – web documents – HTTP Network Security: Introduction - definitions – two categories - symmetric key cryptography – traditional ciphers – asymmetric key cryptography.

Text Book:
[1] Behrouz A Forouzan, *Data communication and networking*, McGraw-Hill,5th Edition, 2012.

Reference Book:
[1] Andrew S Tanenbaum, *Computer Networks*, PHI publications, 5th Edition, 2011.

CSH632 UNIX Programming

Total teaching Hours/Semester: 60   *No of Lecture Hours/Week: 04*

Objective
UNIX is the multi user and multi tasking operating system. After learning the concepts of an operating system, it is appropriate to learn how UNIX implements these concepts. The subject is introduced with the features and basic architecture of UNIX. The file system, process management and memory management are discussed which makes the students to understand the internals of UNIX.

Learning outcome
Upon successful completion of the course students can
- Use line and screen text editors with regular expressions.

- Explain UNIX file system including advanced file processing and practice pipelining and IO redirecting.

- Be familiar with reading and using man page information as well as other standard reference materials.

- Explain process concepts and cooperating processes and write complex shell scripts.

- Clearly and accurately explain design decisions in written program documentation.

- Use UNIX as a programming platform.

Unit I.   **(09)**
Introduction To Unix
Evolution of UNIX – UNIX System Structure – Features of Unix - Operating System Services - Unix Kernel - Locating Commands, Internal and External Commands, Flexibility of Command Usage, man: Browsing and Manual Pages On-line, Understanding the man Documentation.
General Purpose Utilities: cal, date, echo, printf, echo, bc, script, passwd, who, uname, tty, stty. Introduction to system calls.

Unit II.   **(11)**
The Unix File System
The File, I-nodes – Structure of a regular file. Directories – Conversion of a path name to an Inode – Super Block – Inode Assignment to a new file – Allocation of disk Blocks .
The HOME Variable: The Home Directory,   Directory related commands: pwd, mkdir, cd, rmdir. Absolute and relative path names.The UNIX File System.
File manipulation commands: cat, cp, rm, mv, more, The lp Subsystem: Printing a File, file, wc, Words and Characters, od, The spell and ispell, cmp, comm, diff. File compression commands: gzip, gunzip, tar, zip, unzip.
Basic file attributes: The –d Option: Listing Directory Attributes, File Ownership, File Permissions, chmod: Changing File Permissions, Directory Permissions, Changing File Ownership. Hard links, Symbolic Links, ln, umask, find.

Unit III.   (09)
Unix Process Management
Process Basics, Process States and Transitions, ps: Process Status, System Processes (-e or –a), Mechanism of Process Creation, Internal and External Commands, Running Jobs in Background, nice: Job Execution With Low Priority, Killing Processes with Signals, Job Control, at and batch: Execute Later, cron: Running Jobs Periodically, time: Timing Processes. PID & PPID.

Unit IV.   (09)
Filters And Communication
Simple Filters - The Sample Database, pr: Paginating Files, head: Displaying the Beginning of a File, tail: Displaying the End of a File, cut: Slitting a File Vertically, paste: Pasting Files, sort: Ordering a File, uniq: Locate Repeated and Non repeated Lines, tr: Translating Characters, An Example: Displaying a Word-count List.
Filters Using Regular Expressions
grep: Searching for a Pattern, Basic Regular Expressions (BRE) – An Introduction, Extended Regular Expressions (ERE) and egrep.
Communication
Communicating with Other Users: Who, Mail, Wall, Send, Mesg, Ftp.

Unit V.   (11)
Unix Shell Environment and The vi Editor
Types of shells – Features of Bourne, C and Korn shells. The Shell's Interpretive Cycle, Pattern Matching – The Wild-cards, Escaping and Quoting, Redirection: The Three Standard Files, /dev/null and /dev/tty: Two Special Files, Pipes, tee: Creating a Tee, Command Substitution, Shell Variables.
Review of vi Operations – Different Modes – Cursor Movement Commands - Edit Commands - Saving and Exiting - Accessing Multiple Files - Interacting with Unix - Macros - Miscellaneous Commands - Alphabetical List of Keys - Customizing vi.

Unit VI.   (11)
Unix Shell Programming
Shell variables - Shell Keywords - Positional parameters - Passing command line arguments.
Arithmetic in shell scripts - Read and Echo - Control Structures - if-then-fi - if-then-else-fi - Nested if - Case control structure – Loops - while-until –for - break and continue. Shell meta characters - Exporting variables - User defined Functions.

Text Books:
[1] Sumitabha Das, *UNIX Concepts and Applications*, 4th Edition, New Delhi: Tata McGraw Hill, 2010.
[2] Yashavant P Kanetkar, *Unix Shell Programming*, New Delhi: BPB Publications, 2008.

Reference Books:
[1] Maurice J Bach.*The Design of Unix Operating System.* New Delhi: Prentice Hall of India Pvt. Ltd, 2008.
[2] Paul Love, Joe Merlino, Craig Zimmerman et al., *Beginning UNIX*, NewDelhi: Wiley Publishing, Inc, 2005.

CSH633 Mobile Applications

Total teaching Hours/Semester: 60   *No of Lecture Hours/Week: 04*

Objective

This course focuses on developing applications for modern smartphone operating systems. Most of the course is dedicated to Google's Android and Microsoft's Windows. Rapid application development techniques are covered, as well as setup of the development environment, real-world testing, and deployment.

Learning outcome

Upon completion of this course, Students will be able to:

- Build your own Android and Windows apps.
- Explain the differences between Android , Windows and other mobile development environments.
- Secure, tune, package and deploy Applications

Unit I.   **(08)**

Overview

A little background about mobile technologies, Different mobile technologies – Android, Windows, IOS, Black Berry, series 40, Bada, Benefits and drawbacks of Smartphone programming, Overview of Android, How it all got started, Why Android different and important, Android Stack overview, Linux kernel, native libraries, App framework, Apps, SDK overview, platforms, tools, versions. Creating and setting up custom Android emulator.

Unit II.   **(12)**

Get Started with Android

Install the android SDK, Install base tools, install SDKs and Add-ons, Install apache Ant, Emulator, and Device. Get know Eclipse, Build , install and Run the Application in your Emulator or Device, Project Structure.

Designing User interface

Designing by declaration, creating the opening screen, using alternate resources, implementing an about box, applying a theme, adding a menu, adding settings, debugging with log messages, debugging with debugger.

Unit III.   **(12)**

Exploring 2D graphics and Multimedia

Learning the basics, adding Graphics to existing apps, handling input, learn to change the final improvements, Playing audio, Playing Video, Adding sound to existing app,

Storing local Data

Reading/writing local data, Accessing the Internal File system, Accessing SD card.

Unit IV.   **(12)**

Location and Sensing

SMS Messaging , Displaying MAPS Location Data, Monitoring and Tracking a Location.

Introducing SQLite, In and Out of SQLite, Hello Database, Data Binding, using content provider, implementing content provider.

Preparing and Publishing

Preparing app for publishing, Deploying APK files, uploading in Market.

Unit V.   **(08)**
Introduction to Windows Phone Programming

Windows 8 GUI development, windows 8 software Development tools, .Net 4.5 features

Windows Phone platform overview, Multitasking windows, interacting from background, local data, working with sensors, tools – phone emulator, debugging and performance, whats new in windows phone 8, app-to-app communication.

Unit VI.   **(08)**
More on Windows phone

Lock screen background, Lock screen badges, Tiles, tiles templates, Tiles update, , Final touch before deploying and testing in emulators, Monetizing the App, in-app purchase.

Text Books:

[1] Grant Allen, *Beginning Android 4*, Apress, 2012.

[2] Wei-Meng Lee, *Beginning android 4 application Development*, John Wiley & sons, Inc, 2012.

[3] Charles Petzol, *Programming Windows,* Microsoft Press,6th Edition, 2012.

Reference Books:

[1] Ed Burnette, *Hello, Android: Introducing Google's Mobile Development Platform,* Pragmatic Bookshelf, 2009.

[2] Jerome (J.F) DiMarzio, *Android - A programmer's Guide*, Tata Mc-graw Hill, 2010.

[3] Charles Petzold, *Programming Windows Phone*, Microsoft Press, 2010.

CSH641A System Software

Total teaching Hours/Semester: 60   *No of Lecture Hours/Week: 04*

Objective
To provide basic knowledge of various system software to get deeper understanding of actual working of a computer system.

Learning Outcome
* Understanding basics of system software.
* Understanding design approaches for developing system software.

Unit I.   **(10)**
Introduction
System Software, Evolution of Components of a Programming System, General Machine Structure - Memory, Registers, Data and Instructions. Machine Language - No Looping, Address modification using instruction as Data and Index registers, Looping. Assembly Language Program using Literals and pseudo -ops.

Unit II.   **(10)**
Assembler
General design procedure, Design of Assembler- Statement of Problem, Data Structures, Format of Databases, Algorithm (2-pass assembler) in brief with flowchart.

Unit III.   **(10)**
Macro Language and the Macro Processor
Macro instructions, Features of Macro facility ,Macro instruction argument, Conditional Macro expansions, Macro call within Macro, Implementation.

Unit IV.   **(10)**
Loaders and Linkers
Loader Schemes- Compile and Go Loader, General Loader scheme, Absolute Loaders, Subroutine Linkages, Relocating Loaders, Direct-Linking Loaders Binders, Linking loaders, Overlays, Dynamic Binders, Design of an Absolute Loader, Design of direct linking loader.

Unit V.   **(08)**
Compilers
Statement of problem – Problem No1 - Recognizing basic elements, Problem No 2 - Recognizing syntactic units and interpreting meaning, Intermediate form , Problem no 3 – storage allocation, problem no 4 – code generation, General model of compiler, General model of compiler.

Unit VI.   **(12)**
Phases of Compilers
Different phases- Lexical Phase, Syntax Phase, Interpretation Phase, Optimization Phase, Storage Assignment Phase, Code Generation Phase and ,Assembly phase. Passes of a Compiler with flow chart.

Text Book:
[1] Donovan, John J, *Systems Programming,* Tata McGraw-Hill, Reprint 2001.

Reference Book:
[1] A.V.Aho and J.D. Ullman, *Compilers-Principles, Techniques and Tools*, Addison Wesley, 3rd Indian Reprint, 2006.

CSH641B Computer Architecture

Total teaching Hours/Semester: 60   *No of Lecture Hours/Week: 04*

Objective
The subject deals with basic model of a computer with the concepts on processor design in which data representation and instruction basics are discussed, the basics of design of an ALU, the concepts of memory design including characteristics and features of different types of memories are included.

Learning outcome
At the end of the course students will be able to
- Understand how computer hardware has evolved to meet the needs of multi- processing systems and explain and demonstrate the concept of top down approach.
- Make a summary on basic operation and state the significant components in computer.
- Explain and demonstrate the basic concept of computer organization and computer architecture.

.
Unit I.   **(09)**
Introduction
Basic Model of a Computer, Computer Components, Register transfer and Micro operations Register Transfer Language ,Register Transfer, Bus and Memory Transfers, Arithmetic Micro operations, Logic Micro operations , Shift Micro operations, Arithmetic Logic and Shift Unit.

Unit II.   **(09)**
Basic Computer organization and design
Instruction codes, Computer registers, Computer Instruction, Timing and control, Instruction cycle, Memory reference instructions, Input output and Interrupt, Design of basic computer, Design of Accumulator logic.

Unit III.   **(11)**
Central Processing Unit
Introduction, General Register Organization, Stacks organizations-Register stack, Memory stack, Reverse polish notation , Instruction formats- Three address instruction, two address instruction, one address instruction, zero address instruction, Addressing modes, Data transfer and manipulation- Data transfer instructions, Data manipulation instructions . Program control.

Unit IV.   **(11)**
Computer Arithmetic
Introduction ,Addition and Subtraction – Addition and subtraction with signed magnitude data, addition and subtraction with signed 2's complement data ,Multiplication Algorithms-Signed magnitude,Booth multiplication algorithm, array multiplier ,Division Algorithms- signed magnitude algorithm, Decimal Arithmetic, Decimal Arithmetic Operation- Addition, Subtraction, Multiplication, Division.

Unit V.   **(10)**
Input Output Organization
Peripheral Device, Input Output Interface – I/O bus and interface modules , I/O versus memory bus, Asynchronous data transfer, Modes of transfer – programmed I/O , Interrupt initiated I/O, Priority Interrupt- Daisy chaining priority , parallel priority interrupt, Priority encoder, Interrupt cycle, Direct Memory Access – DMA controller and DMA transfer.

Unit VI.   **(10)**
Memory Organization
Memory Technology, Random Access Memory, (Organization, Semiconductor RAM, RAM Design, Memory Interleaving), Serial Access Memories (Access Methods, Memory Organization, Magnetic Surface Recording, Magnetic Disk Memories, Optical Memories), Memory Systems, Multilevel Memories (General Characteristics, Virtual Memory, Locality of Reference, Address Translation, Base Addressing), Cache, (Main Features, Cache Organization, Cache operation), Associative Memory.

Text Book:
[1] Mano M Morris, *Computer System Architecture,* PHI, 3rd Edition, 2008.

Reference Books:
 [1] Stalling, Williams,*Computer Organization and Architecture*, 7th Edition, 2010.
 [2] Hayes, John. P, *Computer Architecture and Organization*, 3rd Edition, McGraw-Hill, 2002.

CSH651 UNIX Programming Lab

Total Hours/Semester: 60   *No of Hours/Week: 04*

SECTION – A

1. Write a shell program using command line arguments.

2. Write a shell program for string manipulation.

3. Write a shell program to demonstrate manipulation of file contents.
   a. Comparing file contents
   b. Conversion of case of characters in a file using command line arguments

4. Write a Shell Script to demonstrate directory related commands.

5. Write a Shell Script to demonstrate various compression commands.

6. Write Shell Scripts to demonstrate recursive functions.

7. Write a shell program to demonstrate process related commands.

8. Write a shell program to demonstrate filter commands.

9. Write a Shell Script to demonstrate communication and date related commands.

10. Write a Shell Script to demonstrate 2-D arrays.

SECTION – B

11. File handling system.
    a. create a file
    b. copy the file
    c. move the file
    d. delete the file
    e. exit

12. Write a menu based program to permit or remove read/write/execute permission of a file.

13. Write a shell program to calculate the net salary of an employee in a particular month.
    Note: Store 3 levels of Designations and corresponding Basic Pay, TA, DA and HRA in a
    file. Input the designation and calculate the net salary according to the designation.

14. Students marks sheet processing.

15. Write a shell program which will
    a. ask the user to enter a filename
    b. check if the file is ordinary file and is readable
    c. display the file if the file is ordinary and readable
    d. display an error message if the file is not ordinary and/or not readable

QUESTION PAPER PATTERN

Two questions will be selected by the examiners. Students have to write and execute both the programs.

CSH652 Mobile Applications Lab

Total Hours/Semester: 60   *No of Hours/Week: 04*

1. Creating "Hello world" Application.

2. Creating an Application that displays message based on the screen orientation.

3. Create an application that displays custom designed Opening Screen .

4. Create menu in Application.

5. Play an audio, based on the user event.

6. Read/ write the Local data.

7. Display Map based on the Current location.

8. Create / Read / Write data with database (SQLite).

9. Hello world – windows app.

10. Create a Tiles-based app.

11. Design a Lock Screen in the existing app.

12. Learn to deploy both android and windows Applications.

QUESTION PAPER PATTERN

Two questions will be selected by the examiners. Students have to write and execute both the programs.

CSH653 Project Lab

Total Hours/Semester: 120　　　　　　　　　　　　　　*No of Hours/Week: 08*

The main project is to introduce the student to the methodology for solving a problem and preparing a report using the steps of software engineering. Student should take up a project of any domain and submit a report.

Guidelines:
1. The students should carry out the project during the allotted lab hours in the computer lab.
2. The project could be of any domain.
3. The guide can award internal marks by evaluating the performance of the students during the course of the project work.
4. The format of the project report shall be instructed by the faculty in-charge.