COMPUTER CONCEPTS AND ‘C’ PROGRAMMING
(4:0:0)

Sub code : CS111/121  CIE : 50% Marks
Hrs/week : 04 Hrs   SEE : 50% Marks
SEE Hrs : 03 Hrs   Max. Marks : 100

Course Outcomes:
On successful completion of the course, the students will be able to

1. Discuss various terminologies and basic interaction methods in the computers
2. Classify basic methods of storing and processing data and discuss the role of operating systems
3. Get familiarize with the concepts of networks and Internet; Basic concepts of flowcharts & algorithms.
4. Cite examples on using operators, decision making and Branching in C.
5. Illustrate the use of arrays & structures of C language
6. Demonstrate the use of user defined functions, pointers & file management of C.

PART – A

UNIT – 1

Introducing Computer Systems:
The Computer defined, Computers for individual users, Computers for organizations, The parts of a computer system, The information processing cycle, Essential computer hardware.

Interacting with Computer:

7 Hrs
UNIT – 2
Processing Data:
Transforming Data into Information: How computers represent data, How computers process data, Factors affecting processing speed.
Storing Data: Magnetic Storage Devices- How data is stored, How data is organized on a disk, Diskettes-Floppy, Hard disk, Optical Storage Devices-CD-ROM, DVD-ROM, Solid-State storage devices- Flash memory and smart card.

Using Operating Systems:

UNIT – 3
Networks and the Internet:
Networking basics - The uses of a network, Common types of networks, Introduction to Internet, Internet’s major services, WWW, E-mail.

Algorithms and Flowcharts:
Algorithms, Flowcharts, Writing algorithms and drawing flowcharts for simple exercises.

Introduction to C:

Constants, Variables, and Data types:
Structure of C, Characters set, C tokens, Keywords and Identifiers, Constants, Variables, Data types, Declaration of variables.

Operators and Expressions:
Arithmetic operators, Relational operators, Logical operators, Assignment operators.

9 Hrs

10 Hrs
PART - B

UNIT – 4

Operators and Expressions:
Increment and decrement operators, conditional operator, Bitwise operators, Special operators, Arithmetic expressions. Evaluation of expressions. Precedence of Arithmetic operators. Type conversions in expressions, Operator precedence and associativity.

Managing Input and Output Operations:
Reading a character, Writing a character, Formatted Input, Formatted Output.

Decision making and Branching:
Decision making with if statement, Simple if statement, The if...else statement, Nesting of if...else statements, The else ... if ladder, The switch statement, the ?: operator, The Go to statement

UNIT – 5

Decision making and Looping:
The while statement, the do statement, the for statement, Jumps in Loops.

Arrays:
One-dimensional Arrays, Declaration of one-dimensional Arrays, Initialization of one-dimensional Arrays, Two-dimensional Arrays, Initializing two-dimensional Arrays

Structures:
Defining a Structure, Declaring Structure Variables, Accessing Structure Variables, Structure Initialization.

UNIT – 6

User-defined Functions:
Need for User-defined Functions, A multi-function Program, Elements of User-defined Functions, Definition of Functions, Return Values and their Types, Function Calls, Function
Declaration, Parameter Passing technique—call by value and call by reference, Category of Functions.

**Pointers:**
Understanding Pointers, Accessing the Address of a Variable, Declaring Pointer Variables, Initialization of Pointer Variables, Accessing a variable through a Pointer.

**File Management in C:**
Introduction, Defining and Opening a File, Closing a File & Input/output operations on Files.

6 Hrs

**TEXT BOOKS:**


**REFERENCE BOOKS:**

COMPUTER PROGRAMMING LABORATORY (0:0:3)

Sub code : CS112/122  Max. Marks : 50
Hrs/week : 03

Course Outcome:

1. Creation and manipulation of word processing documents.
2. Creation of Pie charts and bar charts
3. Creation of power point presentations
4. Using of simple C programs
5. Using features like functions, matrices and pointers in C programs

PART – A

1. Create a document using a suitable word processing package, with at least three paragraphs and perform the following operations:
   1. Set left margin 1" and right margin 0.75"
   2. Centre the heading and make it bold. Increase the font size
   3. Underline the specified words in the document and change them to italics
   4. Conduct spell check and correct them suitably
   5. Demonstrate use of numbering and bullets
   6. Exchange paragraphs 2 and 3 using cut and paste facility
   7. Put suitable headers and footers
   8. Count the number of words and lines and demonstrate the use of drawing tools
   9. Include suitable logo/emblem/symbol
   10. Include the a table

2. You have a monthly income of Rs.11000. Your monthly expenses are Rent-Rs.4000, Food-Rs.3000, Electricity-Rs.40, Phone-Rs. 60, and Cable TV-Rs. 30. Make out a clear spreadsheet with the Monthly Income, the Monthly Expenses listed and summed, the remainder (what’s left over each month) calculated, and the amount left over per day (assuming
30 days in a month). Draw a pie chart to describe the monthly expenses.

3. Generate pay slip for different categories of employees (example: manager, assistant manager, supervisor, clerk etc) using spreadsheet which includes Basicpay, DA, HRA and other allowances along with deductions like PF, Professional Tax, and other deductions. DA and HRA are calculated in terms of percentage which keeps varying. Use formula while performing calculations. Each column and row should display total. Sort the name list in ascending order. Draw the bar chart to describe the DA percentage for different types of employees.

4. Create a PowerPoint presentation consisting of
   Master slide generation,
   Minimum 8 slides
   Information about your college
   Include animation, Hyperlinks, Insert Header and Footer, Insert files, images, tables and Charts. Slide show features.

5. Write a C program to find and output all the roots of a given quadratic equation, for non-zero coefficients. (Using if...else statement)

6. Write a C program to simulate a simple calculator that performs arithmetic operations like addition, subtraction, multiplication, and division only on integers. Error message should be reported, if any attempt is made to divide by zero. (Using switch statement)

7. Write a C program to reverse a given six digit integer number and check whether it is a palindrome or not. Output the given number with suitable message. (Using While statement)

8. Write a C program to generate and print first ‘N’ Fibonacci numbers. (Using do-While statement)

9. Write a C program to find the GCD and LCM of two integers and output the results along with the given integers. Use Euclid’s algorithm. (Using looping constructs)

10. Write a C program to find whether a given number is prime or not. Output the given number with suitable message. (Using for statement)
PART – B

11. Write a C program to input N real numbers in ascending order into a single dimension array. Conduct a binary search for a given key integer number and report success or failure in the form of a suitable message.

12. Write a C program to input N integer numbers into a single dimension array. Sort them in ascending order using bubble sort technique. Print both the given array and the sorted array with suitable headings.

13. Write a C program to read two matrices A(M x N) and B(P x Q) and perform addition. Output the input matrices and the resultant matrix with suitable headings and format. (Using two dimension arrays where array size M, N, P, Q ≤ 3)

14. Write C user defined functions:
   1. To input N integer numbers into a single dimension array.
   2. To conduct a linear search.

Using these functions, write a C program to accept the N integer numbers & given key integer number and conduct a linear search. Report success or failure in the form of a suitable message.

15. Write C user defined functions:
   - To input N integer numbers into a single dimension array.
   - To sort the integer numbers in descending order using selection sort technique.
   - To print the single dimension array elements.

Using these functions, write a C program to input N integer numbers into a single dimension array, sort them in descending order, and print both the given array & the sorted array with suitable headings.

16. Write C user defined functions:
   To input N real numbers into a single dimension array.
   Compute their mean.
   Compute their variance
   Compute their standard deviation.
Using these functions, write a C program to input N real numbers into a single dimension array, and compute their mean, variance & standard deviation. Output the computed results with suitable headings.

17. Write a C program to read a matrix A(M x N) and to find the following using user defined functions:
   - Sum of the elements of the specified row
   - Sum of the elements of the specified column
   - Sum of all the elements of the matrix

Output the computed results with suitable headings.

18. Write C user defined functions:
   1. To read the elements of a given matrix of size M x N
   2. To print the elements of a given matrix of size M x N
   3. To compute the product of two matrices

Using these functions, write a C program to read two matrices A (M x N) and B (P x Q) and compute the product of A and B after checking compatibility for multiplication. Output the input matrices and the resultant matrix with suitable headings and format. (Using two dimension arrays where array size M, N, P, Q ≤ 3)

19. Write a C program to create an integer variable, Assign value to the variable, create a pointer to this variable and print the value of the variable and address of the variable using pointer.

20. Write a C program to create a structure called Employee with members Name, Job, Salary. Create a structure variable. Accept the input values for the structure members at run time. Suitably display the same.

21. Write a C program to create a C file called student.rec and store information about a student, in terms of his NAME, USN, BRANCH, SEMESTER.

REFERENCE BOOK:

53
FUNCTIONAL ENGLISH
(MANDATORY LEARNING COURSE)

Introduction Grammar
- Importance of Languages
- Parts of Speech, Usage of Preposition and Article, Punctuation
- 5 Hrs

Tenses & Degrees of Comparison
- 3 Hrs

Transformation of Sentences
- Active-Passive, Affirmative-Negative, Exclamatory-Assertive, Interrogative-Assertive, Kinds of sentences
- 5 Hrs

Direct-Indirect Speech
- 5 Hrs

Vocabulary Usage
- Homonyms, Correcting Spelling, One-word equivalents
- 7 Hrs

Precis Writing

Essay/Report Writing

Letter Writing
- Personal, Official, Applications
- 5 Hrs

Idioms & Phrases
- Meaning & Usage in sentences
- 5 Hrs

Comprehension
- Of an unseen passage
- 2 Hrs

Elaboration
- Expansion of ideas, proverbs
- 2 Hrs

Presentation
- Preparation of materials and presentation – step
- 3 Hrs

TEXT BOOKS:


REFERENCE BOOKS:

1. English Rank Scorer, G. Sankaran, Addone Publishing group, Thiruvanantapuram, Kerala

2. English Grammar, Wren & Martin


DATA STRUCTURES WITH ‘C’ (4:0:0)

Sub code : CS 0404
Hrs/week : 04
SEE Hrs : 03 Hrs

CIE : 50% Marks
SEE : 50% Marks
Max. Marks : 100

Course Outcome

On successful completion of the course the students will be able to

1. Analyze and implement the concepts using the pointers and to make use of the dynamic memory allocations.

2. Illustrate the data structures like stack and queues using programming.

3. Design the linked list for the various applications and also to discriminate the different types of the linked list.

4. Apply the concepts of recursions in solving a problem

5. Create the trees, BST and Huffman codes for the different applications using them.

6. Compare the efficiency of different searching and searching algorithms.

PART – A

UNIT 1

Review of pointers and structures of C: Concepts, Pointer variables, Accessing variables through pointers, Pointer declaration and definition, Initialization of pointer variables, Pointers and functions, Pointer to pointers, Compatibility, Lvalue and Rvalue, Arrays and pointers, Pointer arithmetic and arrays, Passing an array to a function, Understanding complex declarations, Memory allocation function, Array of pointers. 8 Hrs

UNIT 2

Stacks: Basic stack operations, Linked List implementation of stacks, Stack algorithm-Creation of Stack, Destruction of stack, Push, Pop, Constraint checking-empty stack, full stack, stack ADT, Application of stack – reversing data, decimal to binary conversion, parsing, Infix to Postfix transformation, Evaluation of postfix
transformation, backtracking, recursion using stack.

**Queues**: Queue operations, Queue Linked List, Queue Algorithms, Queue ADT, Queuing Theory, Queue Applications, categorizing data, queue simulation.  

**UNIT 3**

**Linked Lists**: Basic operations - Insertion, Deletion, Retrieval, Traversal, Implementation, Data structure, Algorithms, List ADT, applications – Application functions, Testing Insert and delete logic, Circularly Linked List, Doubly Linked List, Multilinked Lists.

**9 Hrs**

**PART – B**

**UNIT 4**

**Recursion**: Factorial, GCD and Fibonacci Number Computation, Prefix to Postfix conversion, More Examples on Linked List Manipulation.

**8 Hrs**

**UNIT 5**


**9 Hrs**

**UNIT 6:**

**Searching**: Sequential Search, Binary searches, Hashing-Hash Functions, Collisions, Collision Resolution, Separate Chaining, and Double Hashing.

**Sorting**: Insertion Sort, Merge Sort, Quick sort, Heap sort.  

**9 Hrs**

**TEXT BOOK:**

Data Structures: A Pseudocode Approach with C, Richard F. Gilberg & Behrouz A. Forouzan, Thomson Learning (Indian edition), 2nd Edition, 2005 (For Units 2, 3, 4, 5, 6, Chapters 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 12, 13)

**REFERENCE BOOKS:**


OBJECT ORIENTED PROGRAMMING WITH C++
(4:0:0)

Sub code : CS 0405  
CIE : 50% Marks  
Hrs/week : 04  
SEE : 50% Marks  
SEE Hrs : 03 Hrs  
Max. Marks : 100  

Course Outcome

On successful completion of the course the students will be able to

1. Provide the basic knowledge about classes, syntax, semantics, and terminology of C++.

2. Understand types of constructors, destructor, friend functions, static member functions and copy control.

3. Gain knowledge about function and operator overloading.

4. Learn the concepts of various types of inheritance and its uses.

5. Apply generic programming technique using function and class templates.

6. Handle different types of errors by using exception handling techniques.

PART – A

UNIT 1

Classes:
Class Definitions and Declarations: Class Definitions, Data Abstraction and Encapsulation, More on Class Definitions, Class Declarations versus Definitions, Class Objects, The Implicit this Pointer, Class Scope: Name Lookup in Class Scope, References, The new and delete Expressions, Argument Passing: Non-Reference Parameters, Reference Parameters, Function Declarations: Default Arguments, Local Objects: Automatic Objects, Static Local Objects, Inline Functions, Class Member Functions: Defining the Body of a Member Function, Defining a Member Function outside the Class.  

8 Hrs
UNIT 2

Initialization:
- Default Arguments and Restrictors
- The Constructor/Initializer
- Implicit Class-Type Conversions
- Explicit Initialization of Class Members
- Friends
- Static Class Members: Static Member Functions, Static Data Members

Copy Control:
- The Copy Constructor: The Synthesized Copy Constructor
- Defining Our Own Copy Constructor
- Preventing Copies
- The Assignment Operator
- The Destructor

UNIT 3

Function Overloading:
- Overloaded Functions: Overloading and Scope, Function Matching and Argument Conversions, The Three Steps in Overload Resolution, Argument-Type Conversions, Pointers to Functions

Overload Operations and Conversions:
- Defining an Overloaded Operator: Overloaded Operator Design
- Input and Output Operators: Overloading the Output Operator <<, Overloading the Input Operator >>
- Arithmetic and Relational Operators: Equality Operators, Relational Operators, Assignment Operators, Subscript Operator, Member Access Operators, Increment and Decrement Operators
- Conversions and Class Types: Why Conversions Are Useful, Conversion Operators, Argument Matching and Conversions, Overload Resolution and Class Arguments, Overloading, Conversions, and Operators

10 Hrs

PART B

UNIT 4

Object Oriented Programming:
- OOP-An Overview
- Defining Base and Derived Classes: Defining a Base Class, protected Members, Derived Classes, virtual and Other Member Functions, Public, Private and Protected Inheritance, Friendship and Inheritance, Inheritance and Static Members, Conversions and Inheritance
- Derived-to-Base Conversions, Conversions from Base to Derived, Constructors and

22
UNIT 5

Templates and Generic Programming:
Template Definitions: Defining a Function Template, Defining a Class Template, Template Parameters, Template Type Parameters, Nontype Template Parameters, Writing Generic Programs, Instantiation: Template Argument Deduction, Function-Template Explicit Arguments, Template-Compilation Models, Class Template Members: Class-Template Member Functions, Template Arguments for Nontype Parameters, Friend Declarations in Class Templates, Member Templates, Static Members of Class Templates.

UNIT 6

Tools and Special Techniques:
Exception Handling: Throwing an Exception of Class Type, Stack Unwinding, Catching an Exception, Rethrow, The Catch-All Handler, Function Try Blocks and Constructors, Exception Class Hierarchies, Automatic Resource De-allocation.

Multiple and Virtual Inheritance: Multiple Inheritance, Conversions and Multiple Base Classes, Copy Control for Multiple Derived Classes, Class Scope under Multiple Inheritance, Virtual Inheritance, Virtual Base Class Declaration, Special Initialization Semantics.

TEXT BOOK


REFERENCE BOOKS


DATA STRUCTURES LAB (0:0:3)

Sub code : CS 0102
Hrs/week : 03
SEE Hrs : 03 Hrs

CIE : 50% Marks
SEE : 50% Marks
Max. Marks : 100

Course Objective: Implementation of Data structures like Stacks, Queues, Linked List, and Trees using C.
OBJECTION ORIENTED PROGRAMMING LAB (0:0:3)

<table>
<thead>
<tr>
<th>Sub code</th>
<th>CIE</th>
<th>SEE</th>
<th>Max. Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS 0103</td>
<td>50% Marks</td>
<td>50% Marks</td>
<td>100</td>
</tr>
<tr>
<td>Hrs/week 03</td>
<td>SEE Hrs 03 Hrs</td>
<td>Max. Marks 100</td>
<td></td>
</tr>
</tbody>
</table>

**Course Objective:** Implementation of basic concepts of Object oriented Programming such as Classes, Objects, data abstractions and Encapsulation, Constructors, operator overloading, virtual functions, polymorphism, Templates, Exception handling techniques using C++.
OPERATING SYSTEMS (3:0:2)

Sub code : CS 0413
Hrs/week : 05
SEE Hrs : 03 Hrs
CIE : 50% Marks
SEE : 50% Marks
Max. Marks : 100

Course Outcome

On successful completion of the course the students will be able to

1. Demonstrate the difference between processes and threads. Knowledge about the issues of scheduling of user-level processes/threads.

2. Analyze the issues and use of locks, semaphores and monitors for Synchronizing multithreaded systems and implement them in multithreaded programs.

3. Knowledge about the issues and design of memory management.

4. Design and analyze the virtual memory management.

5. Explain the concepts of file system and implementation.

6. Master the concepts of deadlock in operating systems and how they can be managed/avoided.

PART – A

UNIT 1

INTRODUCTION, PROCESSES AND THREADS
WHAT IS AN OPERATING SYSTEM? : The Operating system as an extended machine, The Operating system as a resource manager.

PROCESSES: The process model, Process creation, Process termination, Process hierarchies, Process states. THREADS: The thread model, Thread usage, implementing threads in user space, Implementing threads in the kernel, Hybrid implementations, Scheduler activations. SCHEDULING: Introduction to scheduling, Scheduling in interactive systems, Thread scheduling. 9 Hrs
UNIT 2
INTERPROCESS COMMUNICATION
Race conditions, Critical regions, Mutual exclusion with busy waiting, Sleep and wakeup, Semaphores, Mutexes, Monitors, Message passing. CLASSICAL IPC PROBLEMS: The dining philosophers problem, The readers and writers problem.

MEMORY MANAGEMENT
BASIC MEMORY MANAGEMENT: Monoprogramming without swapping or paging, Multiprogramming with fixed partitions, Modeling multiprogramming, Analysis of multiprogramming system performance, Relocation and protection. 8 Hrs

UNIT 3
MEMORY MANAGEMENT, PAGE REPLACEMENT ALGORITHMS & DESIGN ISSUES
SWAPPING: Memory management with bitmaps, Memory management with linked lists. VIRTUAL MEMORY: Paging, Page tables, TLBs—Translation Look aside Buffers, Inverted page tables.

PAGE REPLACEMENT ALGORITHMS: The optimal page replacement algorithm, The not recently used page replacement algorithm, The first-in first-out, The second chance page replacement algorithm, The clock page replacement algorithm, The least recently used, Simulating LRU in software. MODELING PAGE REPLACEMENT ALGORITHMS: Belady’s anomaly, Stack algorithms. 9 Hrs

PART – B

UNIT 4
DESIGN ISSUES FOR PAGING SYSTEMS
Local versus Global allocation policies, Load control, Page size, Separate instruction and data spaces, Shared pages, Cleaning policy, Virtual memory interface. IMPLEMENTATION ISSUES: Operating system involvement with paging, Page fault handling, Instruction backup, Locking pages in memory, Backing store, Separation of policy and mechanism. SEGMENTATION: Implementation of pure segmentation, Segmentation with paging. 8 Hrs

12

Scanned by CamScanner
UNIT 5

INPUT/OUTPUT

I/O SOFTWARE LAYERS: Interrupt handlers, Device drivers, Device-Independent I/O software.

FILE SYSTEMS


9 Hrs

UNIT 6

DEADLOCKS

INTRODUCTION TO DEADLOCKS: Conditions for deadlock, Deadlock modeling. THE OSTRICH ALGORITHM, DEADLOCK DETECTION AND RECOVERY: Deadlock detection with one resource of each type, Deadlock detection with multiple resource of each type, Recovery from deadlock, DEADLOCK AVOIDANCE: Resource trajectories, Safe and Unsafe States, The Banker’s algorithm for a single resource, The banker’s algorithm for multiple resources, DEADLOCK PREVENTION: Attacking the mutual exclusion condition, Attacking the Hold and Wait condition, Attacking the No Preemption condition, Attacking the Circular Wait condition.

9 Hrs

TEXT BOOKS:


2. Operating system - Design and Implementation 2/E, Tanenbaum, A.S., Woodhull Albert S Prentice Hall, India

REFERENCE BOOKS:


DATABASE MANAGEMENT SYSTEMS (3:0:2)

Sub code : CS 0414
Hrs/week : 05
SEE Hrs : 03 Hrs

CIE : 50% Marks
SEE : 50% Marks
Max. Marks : 100

Course Outcome

On successful completion of the course the students will be able to

1. Choose the available different database models for a given application
2. Design the database using Entity Relationship model and normalization
3. Apply the SQL for the query to fetch the existing data from the tables and to operate on transactions
4. Create triggers, constraints and objects in MYSQL
5. Demonstrate the use of Indexes and Views
6. Detect the failures and recovery from the different types of failures using logs

PART – A

UNIT 1

Introduction to Databases: Data stored in other ways, Different Database Models, Interfacing with a Database, The Mechanics, Disk versus Main-Memory.
An Example Database: Functions, Requirements Reality Check.
Relational Databases(with a MySQL Flavor): MySQL Database System, Database Organization, Creating and using the Database, Steps in Designing a Database. 8 Hrs

UNIT 2

Manipulating the Database: Example Tables, Relational Algebra.
Database Design: The Entity-Relationship Model, Ad Hoc Database Design, Normalized Tables. 9 Hrs
UNIT 3
SQL: Basics, Data Definition Language, Data Manipulation, Language, stored Routines, Orders, Invoices & Reports, Formatting and SQL. Transactions: Informally Speaking, Transactions in SQL, Transaction Semantics, Serializability, Locks, SQL Isolation Levels, Transactions in MySQL

PART – B

UNIT 4
Constraints: Constraints in SQL, Constraint Check Time. Triggers: Triggers in SQL. Objects: Objects in SQL, Referencing Objects, Comparing Objects.

UNIT 5
Indexes: How does an Index Work?, Types of Indexes, Different Types of Indexes, Hash Indexes, Some Thoughts on Indexing, Indexes in MySQL, Views: Two Implementations of Views, Views in MySQL, Examples.

UNIT 6
Spatial Databases: MySQL, Example Logs and Recovery: Failure Types, Logs, Log Details, Some Log Characteristics, Database Recovery, Database Recovery Example, MySQL.

TEXT BOOK:

REFERENCE BOOKS:
SOFTWARE ENGINEERING (4:0:0)

Sub code : CS 0417  
Hrs/week : 04     
SEE Hrs : 03 Hrs  

CIE : 50% Marks    
SEE : 50% Marks    
Max. Marks : 100

Course Outcome

On successful completion of the course the students will be able to

1. Acquire Knowledge of fundamental concepts of software engineering, software engineering process, process assessments and product. Key abstraction of waterfall model, Incremental process model and many other process models.

2. Give a versatile knowledge and with an example in analyzing the design aspects of agile view process and system engineering in business product and system modeling.

3. Identify the key steps in Gathering details while requirement engineering.

4. Design with the context of Software Engineering from architectural modeling design to component modeling design.

5. Introduce test strategic approaches, explain black box testing and white box testing.

6. Explain the methods of managing people and the product with different skill sets.

PART – A

UNIT 1


UNIT 2


UNIT 3


PART – B

UNIT 4


Modelling Component-Level Design: Component, Designing Class Based Components, Conducting Component-Level Design.

UNIT 5
UNIT 6


Risk Management: Risk Management, Software risk, Risk Identification, Risk Projection, Risk Refinement, Risk Mitigation, Monitoring & Management. 9 Hrs

TEXT BOOK:


REFERENCE BOOKS:


COMPUTER NETWORKS (4:0:0)

Sub code : CS 0418
Hrs/week : 04
SEE Hrs : 03 Hrs

Course Outcome

On successful completion of the course the students will be able to

1. Identify of the different layers in the networking software along with their functionality.

2. Explain the functions of the Networking layer.

3. Acquire knowledge about working of different Routing algorithms.

4. Identify functions of the transport layer such as providing both connection oriented (TCP) and connectionless (UDP) communication from host to host. Identify the need to control congestion and ways of doing it.

5. Identify the need to store information on WWW and the different kinds of document formats needed.

6. Identify the prominent application layer protocols making use of the transport layer services to provide quality utility tools along with their working.

PART-A

UNIT 1

Network Layer-1: Switching; Datagram Networks, Virtual circuit Networks; Logical addressing: IPv4 addresses, IPv6 addresses; Internet Protocol: Internetworking, IPv4, IPv6, transition from IPv4 to IPv6. 9 Hrs

UNIT 2

Network Layer-2: Address Mapping: ARP, RARP, BOOTP, DHCP; ICMP, Delivery: Direct verses Indirect delivery; Forwarding: Forwarding Techniques, Forwarding Process and Routing Table. 8 Hrs
UNIT 3


UNIT 4


UNIT 5

Application Layer: Domain Name System: Name Space, Domain Name Space, Distribution of Name space, DNS in the Internet, Resolution, DNS messages, Types of Records, Registrars, Dynamic DNS, Encapsulation; Electronic Mail; WWW and HTTP: Architecture, Web Documents, HTTP; File Transfer: File Transfer Protocol, Anonymous FTP.  

UNIT 6


TEXT BOOKS:

DATABASE LABORATORY (0:0:3)

Sub code : CS 0108  
Hrs/week : 03  
SEE Hrs : 03 Hrs

CIE : 50% Marks  
SEE : 50% Marks  
Max. Marks : 100

Course Objective: Development of Databases using MySQL for real world scenarios in a project based approach.
Computer Networking laboratory (0:0:3)

Sub code : CS 0109
Hrs/week : 03
SEE Hrs : 03 Hrs

CIE : 50% Marks
SEE : 50% Marks
Max. Marks : 100

Course Objective: Development of Networks using NS2 or equivalent simulators in world scenarios in a project based approach.
Cloud Computing (4:0:0)

Sub code : CS 0437
Hrs/week : 04
SEE Hrs : 03 Hrs
CIE : 50% Marks
SEE : 50% Marks
Max. Marks : 100

Course Outcome

On successful completion of the course the students will be able to

1. Analyze what is cloud computing and what is not.

2. Develop an insight into the basic cloud architecture and cloud services

3. Use the concepts of abstraction and virtualization to plan a cloud environment

4. Explain clear insight into the working of a cloud, with special reference to google cloud services.

5. Take decisions on cloud deployment with special reference to cloud security issues

6. Acquire the working knowledge of Service Oriented Architectures and Cloud based storages.

PART - A

UNIT 1 : Introduction


UNIT 2 : Understanding Cloud Architecture

UNIT 3: Understanding Services, Applications, Abstraction and Virtualization

Defining Infrastructure as Service (IaaS), Defining Platform as a Service (PaaS), Defining Software as a Service (SaaS), Defining Identity as a Service (IDaaS), Defining Compliance as a Service (CaaS), Using Virtualization Technologies, Load Balancing and Virtualization, Understanding Hypervisors, Understanding Machine Imaging, Porting Applications. 10 Hrs

PART – B

UNIT 4: Using Google Web Services and Managing the Cloud

Exploring Google Applications, Surveying the Google Application Portfolio: Indexed Search, The dark Web, Aggregation and disintermediation, Productivity applications and services, Enterprise offerings, Google Analytics, Google Translate, Exploring the Google Toolkit, Working with Google App Engine, Administrating the Clouds, Cloud Management Products, Emerging Cloud Management Standards. 8 Hrs

UNIT 5: Understanding Cloud Security


UNIT 6: Understanding Service Oriented Architecture and Moving Applications to the Cloud

Introduction to C# programming
And .Net Concepts (4:0:0)

Sub code : CS 0426
Hrs/week : 04
SEE Hrs : 03 Hrs
CIE : 50% Marks
SEE : 50% Marks
Max. Marks : 100

Course Outcome

On successful completion of the course the students will be able to

1. Explain the basics of .Net platform and the role of base class libraries, role of common intermediate language and namespaces and also to Define and deploy the different command line compiler options.

2. Discuss the fundamentals of c# and to build the basic c# program using different constructs.

3. Review the basic pillars of object oriented programming concepts and apply the exception handling technique to handle different types of errors.

4. Provide the knowledge about basics of object lifetime and to define the use of interfaces and collections.

5. Explain the callback interfaces, delegates, advanced C# keywords and its implementation.

6. Build and deploy local and shared assemblies.

PART-A

UNIT-1

THE PHILOSOPHY OF .NET: Understanding the Previous State of Affairs, The .NET Solution, The Building Block of the .NET Platform (CLR,CTS, and CLS), The Role of the .NET Base Class Libraries, What C# Brings to the Table, An Overview of .NET Binaries (aka Assemblies ), the Role of the Common Intermediate Language , The Role of .NET Type Metadata, The Role of the Assembly Manifest, Compiling CIL to Platform –Specific Instructions, Understanding the Common Type System, Intrinsic CTS Data Types, Understanding the Common Languages Specification,
Embedded systems (3:0:2)

Sub code : CS 0428
Hrs/week : 04
SEE Hrs : 03 Hrs

CIE : 50% Marks
SEE : 50% Marks
Max. Marks : 100

Course Outcome

On successful completion of the course the students will be able to

1. Describe the fundamentals of Embedded Systems

2. Develop necessary skills to understand and design an embedded system application.

3. Compare the advantages of the design of an embedded system application using an RTOS

4. Design the Hardware along with the choice of the RTOS for the application at hand.

5. Model the requirements of an application as a set of tasks

6. Relate the tasks with proper inter task communication primitives like semaphores and message queues

PART – A

UNIT 1

Custom single-purpose processor design; RT-level custom single-purpose processor design; Optimizing custom single-purpose processors: Optimizing the original program, optimizing the FSMD, Optimizing the data-path, optimizing the FSM. 8 Hrs

UNIT 2

Timers, counters, and watchdog timers.
State machine models: Introduction; An introductory example, A basic state machine model: finite-state machines (FSM); Finite-state machines with data path model (FSMD); Using state machines: Describing a system as a state machine, Comparing state machine and sequential program models, Capturing a state machine model in a sequential programming language; Hierarchical/Concurrent state machine model (HCFSM) and the State charts language; Program state machine model (PSM); The
role of an appropriate model and language. 9 Hrs

UNIT 3

Concurrent process models: Concurrent processes: Process create and terminate, Process suspend and resume, Process join; Communication among processes: Shared memory, Message passing; Synchronization among processes: Condition variables, Monitors.

Interrupts: Interrupt Basics; The Shared Data Problem; Interrupt latency. 9 Hrs

PART – B

UNIT 4


Introduction to RTOS: Tasks and Task States, Tasks and Data, Semaphores and shared data. 8 Hrs

UNIT 5

Operating Systems Services: Message Queues, Mailboxes, and Pipes; Timer Functions; Events; Memory Management, Interrupt Routines in an RTOS environment. Basic Design Using an RTOS: Overview, Principles, An Example, Encapsulating semaphores and Queues, Hard Real-Time Scheduling Considerations, Saving Power. 9 Hrs

UNIT 6

Embedded Software Development Tools: Host and Target Machines, Linker/Locator for Embedded Software, getting embedded software into the Target System

Debugging Techniques: Testing on host machine, Instruction set simulators, macros and tools.

An example system: What the program does, Environment in which the program operates. 9 Hrs
WEB TECHNOLOGIES AND APPLICATIONS (3:0:2)

Sub code : CS 0420
Hrs/week : 05
SEE Hrs : 03 Hrs

CIE : 50% Marks
SEE : 50% Marks
Max. Marks: 100

Course Outcome

On successful completion of the course the students will be able to

1. Explain why Java is important in the Internet, how Java is different from other language, cite examples on using data types, variables, operators, control statements. Write programs using arrays and vectors

2. Compare and demonstrate the different category of classes, Recognize group of classes with different visibility control to create package; Prediction of errors and handling them using exceptional handling;

3. Analyze working of I/O operations, compare functions of string and stringbuffer class

4. Prepare web pages using HTML/XHTML and CSS.

5. Create dynamic web pages using CGI script. Illustrate how to create database using MYSQL and link them with perl script.

6. Demonstrate how to embed php into HTML and how to link php with MYSQL to create dynamic pages

PART - A

UNIT 1

The Java Language – 1: The Genesis of Java: Java’s Lineage, The creation of Java, Why Java is important to the Internet? Java’s Magic, Java features; An overview of Java Programming: Object oriented programming, simple java program, Lexical issues; Data types, Variables and Arrays: Simple types, Literals, Variables, Type conversions and Casting, Arrays, Operators, Control Statements, Introduction to Java Classes: Classes, Object, Methods, Constructors, The this keyword, Garbage collection, The Finalize() Method, A Stack Class.
Implementation on above important concepts. 7 Hrs

UNIT 2

The Java Language – 2: Methods and Classes; Inheritance; Inheritance Basic, Using Super, Creating Multilevel hierarchy, When Constructors are Called, Method Overriding, Dynamic Method Dispatch, Using Abstract Classes, Using Final, the Object Class; Java Packages and Interfaces; Exception Handling; Multithreading Programming.

Implementation on above important concepts. 7 Hrs

UNIT 3


Implementation on above important concepts. 7 Hrs

PART-B

UNIT 4

XHTML, XML and CSS: XHTML: Basic syntax; Standard XHTML document structure; Basic text markup, Images; Hypertext Links; Lists; Tables; Forms; Frames; CSS: Introduction; Levels of style sheets; Style specification formats; Selector forms; Property value forms; Font properties; List properties; Color; Alignment of text; The Box model; Background images; The and tags; Conflict resolution. XML: Introduction; Syntax; Document structure; Document Type definitions; Namespaces; XML schemas; Displaying raw XML documents; Displaying XML documents with CSS.

Implementation on above important concepts. 7 Hrs

UNIT 5

CGI and MYSQL: Introduction to Apache Web Server- Introduction, Starting, stopping and Restarting Apache, Apache Log Files, MYSQL: Introduction, Tutorial, DBI, Table Joins, Loading and Dumping a Database. CGI - Introduction, Apache
Configuration, A Simple CGI Program, what can go wrong in the program? CGI.pm: Introduction, CGI.pm HTML Shortcuts, Information received by the CGI program, Form Widget Methods, Security consideration, A note on die().

Implementation on above important concepts. 7 Hrs

UNIT 6

PHP and MYSQL: PHP Introduction, Embedding PHP into HTML, Configuration, Quick examples, Language Syntax, Built-IN PHP Functions, PHP and MySQL.

Implementation on above important concepts. 7 Hrs

Note: Implementation of the concepts taught in the theory classes will be done in laboratory sessions.

TEXT BOOKS:


3. Open Source Web Development with Lamp, James Lee and Brent Ware, Pearson Education, 2009

REFERENCE BOOKS;


CRYPTOGRAPHY (3:0:2)

Sub code  : CS 0421       CIE  : 50% Marks
Hrs/week : 05            SEE  : 50% Marks
SEE Hrs  : 03 Hrs        Max. Marks: 100

Course Outcome

On successful completion of the course the students will be able to

1. Explain the foundations of Cryptography and network security. Critical evaluation of the risks and threats to networked computers. Demonstration of the various mechanisms to protect data and their limitations.

2. Demonstrate detailed knowledge of the role of encryption to protect data using DES.

3. Demonstrate advanced encryption technique like AES.

4. Familiar with cryptographic techniques for secure (confidential) communication of two parties over an insecure (public) channel using public key algorithms.

5. Verify the integrity of the messages transmitted via an insecure channel and unique identification of the originator of any message using MAC.

6. Explain the different key management and digital signature systems.

PART – A

UNIT 1

Introduction to Symmetric-Key Encipherment: Security Goals, Cryptographic Attacks, Services and mechanism, Techniques.

Traditional Symmetric-Key Ciphers: Introduction, Substitution Ciphers, Transposition Ciphers, Stream and Block Ciphers. 9 Hrs

UNIT 2

Data Encryption Standard (DES): Introduction, DES Structure (overview only), Security of DES, Multiple DES-Conventional Encryption Algorithms. 9 Hrs

UNIT 3


Encipherment Using Modern Symmetric-Key Ciphers: Use of Modern Block Ciphers (overview only), Use of Stream Ciphers (overview only) 8 Hrs

PART – B

UNIT 4

Asymmetric-Key Cryptography: RSA Cryptosystems, Elliptic curve crypto systems (overview only), ElGamal Cryptosystem. 9 Hrs

UNIT 5

Message Integrity and Message Authentication
Message Integrity, Random Oracle Model, Message Authentication.

Digital Signature: Comparison, Process, Services, Attacks of Digital Signature, Digital Signature Schemes, Variations and Applications. 9 Hrs

UNIT 6


Key Management: Symmetric-Key Distribution, Kerberos, Symmetric-key Agreement, Public- Key Distribution. 8 Hrs

TEXT BOOK

Students are expected to go through the recent developments in the area of information technology / Computer Science & Engineering and identify a problem to be solved. For the so identified problem, the requirements need to be formulated thoroughly.

In this phase of project, the students will formulate the problem – perform analysis and check the feasibility of solution in terms of time, cost and operations. The students are expected to submit a report at the end of the project phase - I.

This work is carried out in a batch. The batch size is limited to a maximum of 4 students.
MANAGEMENT AND ENTREPRENEURSHIP (4:0:0)

Sub code : CS 0424
Hrs/week : 04
SEE Hrs : 03 Hrs

CIE : 50% Marks
SEE : 50% Marks
Max. Marks : 100

Course Outcome

On successful completion of the course the students will be able to

1. Provide the information about management as science , art, administration and as a key approach in doing a assigned work.
2. Explains planning and organizing, students learn its importance and hierarchy, Departmentation, committees, centralization and decentralization.
3. Explain the importance of staffing directing and controlling , meaning and steps in controlling.
4. Identifying who is an entrepreneur , his functions and types and identifying based on geography.
5. Provide knowledge of small scale industry and institutional support.
6. Prepare Project Selection; Project criteria Report; Need and Significance of Report;

PART – A

UNIT 1

Management: Introduction - Meaning - nature and characteristics of Management, Scope and functional areas of management - Management as a science, art or profession Management & Administration - Roles of Management, Levels of Management, Development of Management Thought - early management approaches - Modern management approaches. 7 Hrs
SEMINAR (0:0:2)

Sub code : CS 0110  
Hrs/week : 02

Seminar should be given by an individual student based on topics chosen from the emerging areas and technologies of Computer science & engineering. References from journals such as IEEE, ACM etc., shall be used.

PROJECT WORK – II (0:0:12)

Sub code : CS 0601  
Hrs/week : 12

Students have to carry out the project identified in project work – I. in this phase students have to develop the Design modules – algorithmic design, / Hardware design / GUI design etc. and implement the same. The system so developed needs to be tested. The results of testing should be documented and reported.

The outcome of this phase is to submit a project report and demonstrate the work.
COMPUTER CONCEPTS AND 'C' PROGRAMMING
(4:0:0)

Sub code : CS0401
Hrs/week : 04
SEE Hrs : 03 Hours

CIE : 50% Marks
SEE : 50% Marks
Max. Marks : 100

Course Outcomes:

On successful completion of the course, the students will be able to

1. Identify the various terminologies and basic interaction
   methods in the computers

2. Distinguish basic methods of storing and processing data
   and also the role of operating systems and networking.

3. Analyse problems and convert them to flowcharts &
   algorithms.

4. Write C programs using operators, decision making and
   branching.

5. Write C programs using arrays & structures.

6. Write C programs using user defined functions, pointers &
   File management.

UNIT – 1

Introducing Computer Systems:

The Computer defined, Computers for individual users, Computers
for organizations, The parts of a computer system, The information
processing cycle, Essential computer hardware.

Interacting with Computer:

The Keyboard – The Standard Keyboard Layout, The Mouse,
Inputting data in other ways – Devices for the hand, Optical Input
Devices and Audiovisual Input Devices. Video and Sound –
Monitors-CRT, Flat-Panel monitors, Printing – Commonly used

SLE: Identification of latest printers and its features.

9 Hrs
UNIT – 2

Processing Data:

Transforming Data into Information: How computers represent data?, How computers process data?, Factors affecting processing speed.

Storing Data: Magnetic Storage Devices- How data is stored?, How data is organized on a disk?, Diskettes-Floppy, Hard disk, Optical Storage Devices-CD-ROM, DVD-ROM, Solid-State storage devices- Flash memory and smart card.

Using Operating Systems:


Networks and the Internet:

Networking basics - The uses of a network, Common types of networks, Introduction to Internet, Internet's major services, WWW, E-mail.

SLE: Importance of Android OS.

UNIT – 3

Algorithms and Flowcharts:

Algorithms, Flowcharts, Writing algorithms and drawing flowcharts for simple exercises.

Introduction to C

Constants, Variables, and Data types:

Structure of C, Characters set, C tokens, Keywords and Identifiers, Constants, Variables, Data types, Declaration of variables.

Operators and Expressions:

Arithmetic operators, Relational operators, Logical operators, Assignment operators.
UNIT - 4

Operators and Expressions:
Increment and decrement operators, conditional operator, Bitwise operators, Special operators, Arithmetic expressions, Evaluation of conversions in expressions, Operator precedence and associativity.

Managing Input and Output Operations:
Reading a character, Writing a character, Formatted Input, Formatted Output.

Decision making and Branching:
Decision making with if statement, Simple if statement, The if...else statement, Nesting of if...else statements, The else ... if ladder, The switch statement, the?: operator, The Goto statement

SLE: Writing Set of Programs on above concept.

UNIT - 5

Decision making and Looping:
The while statement, the do statement, the for statement, Jumps in Loops.

Arrays:
One-dimensional Arrays, Declaration of one-dimensional Arrays, Initialization of one-dimensional Arrays, Two-dimensional Arrays, Initializing two-dimensional Arrays

Structures:
Defining a Structure, Declaring Structure Variables, Accessing Structure Variables, Structure Initialization.

SLE: List applications of arrays and structures. Writing Programs on above concept.
UNIT – 6

User-defined Functions:

Need for User-defined Functions, A multi-function Program, Elements of User-defined Functions, Definition of Functions, Return Values and their Types, Function Calls, Function Declaration, Parameter Passing technique- call by value and call by reference, Category of Functions.

Pointers:

Understanding Pointers, Accessing the Address of a Variable, Declaring Pointer Variables, Initialization of Pointer Variables, Accessing a variable through a Pointer.

File Management in C:

Introduction, Defining and Opening a File, Closing File & Input/output operations on Files.

SLE: Programs on I/O operations on files, applications of pointers.

TEXT BOOK:


REFERENCE BOOK:

1. Introduction to Computer Science, ITL Education Solutions Ltd.
4. The C programming language, Brian w. Kernighan, Dennis Ritchie, 2nd edition
Course Outcome

1. Creation and manipulation of word processing documents.
2. Creation of Pie charts and bar charts
3. Creation of power point presentations
4. Using of simple C programs
5. Using features like functions, matrices and pointers in C programs

PART – A

1. Create a document using a suitable word processing package, with at least three paragraphs and perform the following operations:
   1. Set left margin 1" and right margin 0.75"
   2. Centre the heading and make it bold. Increase the font size
   3. Underline the specified words in the document and change them to italics
   4. Conduct spell check and correct them suitably
   5. Demonstrate use of numbering and bullets
   6. Exchange paragraphs 2 and 3 using cut and paste facility
   7. Put suitable headers and footers
   8. Count the number of words and lines and demonstrate the use of drawing tools
   9. Include suitable logo/emblem/symbol
   10. Include a table

2. You have a monthly income of Rs.11000. Your monthly expenses are Rent-Rs.4000, Food-Rs.3000, Electricity-Rs.40, Phone-Rs.60, and Cable TV-Rs.30. Make out a clear
spreadsheet with the Monthly Income, the Monthly Expenses listed and summed, the remainder (what’s left over each month) calculated, and the amount left over per day (assuming 30 days in a month). Draw a pie chart to describe the monthly expenses.

3. Generate pay slip for different categories of employees (example: manager, assistant manager, supervisor, clerk etc) using spreadsheet which includes Basicpay, DA, HRA and other allowances along with deductions like PF, Professional Tax, and other deductions. DA and HRA are calculated in terms of percentage which keeps varying. Use formula while performing calculations. Each column and row should display total. Sort the name list in ascending order. Draw the bar chart to describe the DA percentage for different types of employees.

4. Create a PowerPoint presentation consisting of
   Master slide generation
   Minimum 8 slides
   Information about your college
   Include animation, Hyperlinks, Insert Header and Footer, Insert files, images, tables and Charts. Slide show features.

5. Write a C program to find and output all the roots of a given quadratic equation, for non-zero Coefficients. (Using if...else statement)

6. Write a C program to simulate a simple calculator that performs arithmetic operations like addition, subtraction, multiplication, and division only on integers. Error message should be reported, if any attempt is made to divide by zero. (Using switch statement)

7. Write a C program to reverse a given six digit integer number and check whether it is a palindrome or not. Output the given number with suitable message. (Using While statement)

8. Write a C program to generate and print first ‘N’ Fibonacci numbers. (Using do-While statement)

9. Write a C program to find the GCD and LCM of two integers and output the results along with the given integers. Use Euclid’s algorithm. (Using looping constructs)
PART - B

11. Write a C program to input N real numbers in ascending order into a single dimension array. Conduct a binary search for a given key integer number and report success or failure in the form of a suitable message.

12. Write a C program to input N integer numbers into a single dimension array. Sort them in ascending order using bubble sort technique. Print both the given array and the sorted array with suitable headings.

13. Write a C program to read two matrices A(M x N) and B(P x Q) and perform addition. Output the input matrices and the resultant matrix with suitable headings and format. (Using two dimension arrays where array size M, N, P, Q ≤ 3)

14. Write C user defined functions:
   1. To input N integer numbers into a single dimension array.
   2. To conduct a linear search.

   Using these functions, write a C program to accept the N integer numbers & given key integer number and conduct a linear search. Report success or failure in the form of a suitable message.

15. Write C user defined functions:
   1. To input N integer numbers into a single dimension array.
   2. To sort the integer numbers in descending order using selection sort technique.
   3. To print the single dimension array elements.

   Using these functions, write a C program to input N integer numbers into a single dimension array, sort them in descending order, and print both the given array & the sorted array with suitable headings.

16. Write C user defined functions:
   1. To input N real numbers into a single dimension array.
   2. Compute their mean.
3. Compute their variance
4. Compute their standard deviation.

Using these functions, write a C program to input N real numbers into a single dimension array, and compute their mean, variance & standard deviation. Output the computed results with suitable headings.

17. Write a C program to read a matrix \( A(M \times N) \) and to find the following using user defined functions:
   1. Sum of the elements of the specified row
   2. Sum of the elements of the specified column
   3. Sum of all the elements of the matrix

   Output the computed results with suitable headings.

18. Write C user defined functions:
    1. To read the elements of a given matrix of size \( M \times N \)
    2. To print the elements of a given matrix of size \( M \times N \)
    3. To compute the product of two matrices

   Using these functions, write a C program to read two matrices \( A(M \times N) \) and \( B(P \times Q) \) and compute the product of \( A \) and \( B \) after checking compatibility for multiplication. Output the input matrices and the resultant matrix with suitable headings and format. (Using two dimension arrays where array size \( M, N, P, Q \leq 3 \))

19. Write a C program to create an integer variable, Assign value to the variable, create a pointer to this variable and print the value of the variable and address of the variable using pointer.

20. Write a C program to create a structure called Employee with members Name, Job, Salary.

   Create a structure variable. Accept the input values for the structure members at run time. Suitably display the same.

21. Write a C program to create a C file called student.rec and store information about a student, in terms of his NAME, USN, BRANCH, SEMESTER.

REFERENCE BOOK:

FUNCTIONAL ENGLISH
(MANDATORY LEARNING COURSE)

<table>
<thead>
<tr>
<th>Topic</th>
<th>Content</th>
<th>Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction</td>
<td>Importance of Languages, Parts of Speech, Preposition and Article, Punctuation</td>
<td>5 Hrs</td>
</tr>
<tr>
<td>Grammar</td>
<td>Tenses &amp; Degrees of Comparison, Active-Passive, Affirmative-Negative, Exclamatory-Assertive, Interrogative-Assertive, Kinds of sentences</td>
<td>3 Hrs</td>
</tr>
<tr>
<td>Transformation of Sentences</td>
<td>Direct-Indirect Speech, Homonyms, Correcting Spelling, One-word equivalents</td>
<td>5 Hrs</td>
</tr>
<tr>
<td>Vocabulary Usage</td>
<td>Precis Writing, Essay/Report Writing, Personal, Official, Applications</td>
<td>7 Hrs</td>
</tr>
<tr>
<td>Idioms &amp; Phrases</td>
<td>Idioms &amp; Phrases, Meaning &amp; Usage in sentences</td>
<td>5 Hrs</td>
</tr>
<tr>
<td>Comprehension</td>
<td>Idioms &amp; Phrases, Expansion of unseen passage, Of an unseen passage</td>
<td>2 Hrs</td>
</tr>
<tr>
<td>Elaboration</td>
<td>Idioms &amp; Phrases, Expansion of unseen passage, Of an unseen passage</td>
<td>2 Hrs</td>
</tr>
<tr>
<td>Presentation</td>
<td>Idioms &amp; Phrases, Preparation of ideas, proverbs</td>
<td>3 Hrs</td>
</tr>
</tbody>
</table>

TEXT BOOKS:

REFERENCE BOOKS:
1. English Rank Scorer, G. Sankaran, Addone Publishing group, Thiruvananantapuram, Kerala
2. English Grammar, Wren & Martin
DATA STRUCTURES WITH ‘C’ (4:0:0)

Sub code : CSO404                  CIE : 50% Marks
Hrs/week : 04                       SEE : 50% Marks
SEE Hrs  : 03 Hours                 Max. Marks: 100

Prerequisite: Programming in ‘C’

Course Outcomes

On Successful completion of the course, the students will be able to:

To understand the basic operations to be performed using pointers and dynamic memory allocations.

1. Implementation of Abstract Data Types using an Arrays and recursion.

2. Design and implement various operations of different types of Queues.

3. Understand the basic operations of linked list and its implement using dynamic variables.

4. Understand the operations and appreciate the applications of Binary trees.

5. Understand different sorting and searching methods.

UNIT – 1

**Pointer:** Understanding pointers, Pointer variables, Accessing address of a variable, Initialization of pointer variables, Accessing variables through pointers, Chain of pointers, Pointer Arithmetic and arrays, Array of pointers, Pointers as function argument, Function returning pointers, Pointers to function, Pointers and structures. **Dynamic memory allocation:** malloc, calloc, free.

**SLE:** Pointer expressions, realloc.

9 hours

UNIT – 2

**Abstract Data Types:** Introduction. **Stack:** Definition and examples, Representing stacks in C, An example: Infix, Postfix.
Recursion: Recursive definition and processes, Recursion in C: Factorial in C, Binary search in C. Writing recursive programs.

SLE: Prefix, recursive chains

UNIT – 3

Queue: The Queue and its representation, Circular Queue, Priority Queue

SLE: Applications of Queues.

UNIT – 4

Linked Lists: Inserting and Removing nodes from a list, Implementation of stacks, getnode and freenode operations, Linked implementation of queues, Linked list as a data structure, Example of list operations, Header nodes, List in C, Array implementation of lists, Limitations of array implementation, Allocating and freeing dynamic variables, Queues as lists in C, Examples of list operations in C, Circular lists, Stack as a circular lists, Double linked list.

SLE: Linked lists using dynamic variables, Primitive operations on circular lists, Queue as a circular list.

UNIT – 5


SLE: An Example: The Huffman algorithm, C Representation of trees, Tree traversal.

Unit 6

Sorting: Binary tree Sort, Heap sort, Insertion Sort, Shell Sort and Merge Sort.
**Searching:** Sequential Search, Binary Searches, Hashing-Hash Functions, Collisons, Collision resolution.

**SLE:** Radix Sorts, Separate Chaining.

**9 Hours**

**TEXT BOOK:**


**REFERENCE BOOKS:**


OBJECT ORIENTED PROGRAMMING WITH C++
(4:0:0)

Sub code : CS0405  
Hrs / week : 04  
SEE Hrs : 3 Hrs  
CIE : 50 %  
SEE : 50 %  
Max. Marks : 100

Prerequisite: C++

Course Outcome

On successful completion of the course the students will be able to

1. Provide the basic knowledge about classes, syntax, semantics, and terminology of C++.
2. Understand types of constructors, destructor, friend functions, static member functions and copy control.
3. Gain knowledge about function and operator overloading.
4. Learn the concepts of various types of inheritance and its uses.
5. Apply generic programming technique using function and class templates.
6. Handle different types of errors by using exception handling techniques.

UNIT 1

Classes:

Class Definitions and Declarations: Class Definitions, Data Abstraction and Encapsulation, More on Class Definitions, Class Declarations verses Definitions, Class Objects, The Implicit this Pointer, Class Scope: Name Lookup in Class Scope, References, The new and delete Expressions, Argument Passing: Non-Reference Parameters, Reference Parameters, Function Declarations: Default Arguments, Local Objects: Automatic Objects, Static Local Objects, Inline Functions, Class Member Functions: Defining the Body of a Member Function, Defining a Member Function outside the Class.

SLE: Data Abstraction and Encapsulation

8 Hours
UNIT 2

Initialization
Constructors: The Constructor Initializer, Default Arguments and Constructors, The Default Constructor, Implicit Class-Type Conversions, Explicit Initialization of Class Members, Friends, Static Class Members: Static Member Functions, Static Data Members.

Copy Control:

SLE: Defining Our Own Copy Constructor

8 Hours

UNIT 3

Function Overloading:
Overloaded Functions: Overloading and Scope, Function Matching and Argument Conversions, The Three Steps in Overload Resolution, Argument-Type Conversions, Pointers to Functions.

Overload Operations and Conversions:

SLE: Subscript Operator, Member Access Operators

10 Hours

UNIT 4

Object Oriented Programming:
OOP-An Overview. Defining Base and Derived Classes: Defining a
Base Class, protected Members, Derived Classes, virtual and Other Member Functions, Public, Private and Protected Inheritance, Friendship and Inheritance, Inheritance and Static Members, Conversions and Inheritance: Derived-to-Base Conversions, Conversions from Base to Derived, Constructors and Copy Control: Base-Class Constructors and Copy Control, Derived-Class Constructors, Copy Control and Inheritance, Virtual Destructors, Virtuals in Constructors and Destructors, Class Scope under Inheritance: Name Lookup Happens at Compile Time, Name Collisions and Inheritance, Scope and Member Functions, Virtual Functions and Scope, Pure Virtual Functions.

SLE: Friendship and Inheritance

10 Hours

UNIT 5

Templates and Generic Programming:

Template Definitions: Defining a Function Template, Defining a Class Template, Template Parameters, Template Type Parameters, Nontype Template Parameters, Writing Generic Programs, Instantiation: Template Argument Deduction, Function-Template Explicit Arguments, Template Compilation Models, Class Template Members: Class-Template Member Functions, Template Arguments for Nontype Parameters, Friend Declarations in Class Templates, Member Templates, Static Members of Class Templates.

SLE: Friend Declarations in Class Templates

8 Hours

UNIT 6

Tools and Special Techniques:

Exception Handling: Throwing an Exception of Class Type, Stack Unwinding, Catching an Exception, Rethrow, The Catch-All Handler, Function Try Blocks and Constructors, Exception Class Hierarchies, Automatic Resource De-allocation.

Multiple and Virtual Inheritance: Multiple Inheritance, Conversions and Multiple Base Classes, Copy Control for Multiply Derived Classes, Class Scope under Multiple Inheritance, Virtual Inheritance, Virtual Base Class Declaration, Special Initialization Semantics.
SLE: The Catch-All Handler

8 Hours

TEXT BOOK


REFERENCE BOOKS


DATA STRUCTURES LAB (0:0:3)

Sub code: CS0102

Hrs/week: 03

Course Outcome

On successful completion of the course the students will be able to

1. Implement Data structures like Stacks, Queues, Linked List, and Trees using C.

2. Implement the basic search and sort algorithms.

3. Appropriate use of a particular data structure and algorithm to solve a problem
OBJECT ORIENTED PROGRAMMING WITH C++ LAB (0:0:3)

Sub code: CS0103
Hrs/week: 03

Course Outcome

On successful completion of the course the students will be able to

1. Understand object-oriented concepts and how they are supported by C++ and implementation issues related to object-oriented techniques.

2. Demonstrate the ability to analyze, use, and create functions, classes, to overload operators, and use inheritance and Pointers when creating or using classes and create templates.

3. Demonstrate the ability to understand and use Exception handling and file handling mechanism.

4. Design and write programs that make appropriate use of advanced object-oriented facilities common to many object-oriented languages such as classes, message passing, overloading and inheritance.
OPERATING SYSTEMS (3:0:2)

Sub code : CS0413
Hrs / week : 05
SEE Hrs : 3 Hrs

CIE : 50 %
SEE : 50 %
Max. Marks : 100

Course Outcome

On successful completion of the course the students will be able to

1. Demonstrate the difference between processes and threads. Knowledge about the issues of scheduling of user-level processes/threads.
2. Analyze the issues and use of locks, semaphores and monitors for synchronizing multithreaded systems and implement them in multithreaded programs.
3. Knowledge about the issues and design of memory management.
4. Design and analyze the virtual memory management.
5. Explain the concepts of file system and implementation.
6. Master the concepts of deadlock in operating systems and how they can be managed/avoided.

UNIT 1

INTRODUCTION, PROCESSES AND THREADS

WHAT IS AN OPERATING SYSTEM? : The Operating system as an extended machine, The Operating system as a resource manager.

PROCESSES: The process model, Process creation, Process termination, Process hierarchies, Process states. THREADS: The thread model, Thread usage, implementing threads in user space, implementing threads in the kernel, Hybrid implementations, Scheduler activations. SCHEDULING: Introduction to scheduling, Scheduling in interactive systems,

SLE: Thread scheduling.

9 Hours
UNIT 2

INTERPROCESS COMMUNICATION

Race conditions, Critical regions, Mutual exclusion with busy waiting, Sleep and wakeup, Semaphores, Mutexes, Monitors, Message passing. CLASSICAL IPC PROBLEMS: The dining philosophers problem, The readers and writers problem.

MEMORY MANAGEMENT

BASIC MEMORY MANAGEMENT: Monoprogramming without swapping or paging, Multiprogramming with fixed partitions, Modeling multiprogramming, Relocation and protection.

SLE: Analysis of multiprogramming system performance

8 Hours

UNIT 3

MEMORY MANAGEMENT, PAGE REPLACEMENT ALGORITHMS & DESIGN ISSUES

SWAPPING: Memory management with bitmaps, Memory management with linked lists. VIRTUAL MEMORY: Paging, Page tables, TLBs—Translation Look aside Buffers, Inverted page tables.


SLE: Belady’s anomaly

9 Hours

UNIT 4

DESIGN ISSUES FOR PAGING SYSTEMS

Local versus Global allocation policies, Load control, Page size, Separate instruction and data spaces, Shared pages, Cleaning policy, Virtual memory interface. IMPLEMENTATION ISSUES: Operating system involvement with paging, Page fault handling, Instruction backup, Backing store, Separation of policy and
mechanism. SEGMENTATION: Implementation of pure segmentation, Segmentation with paging.

SLE: Locking pages in memory.

UNIT 5

INPUT/OUTPUT

I/O SOFTWARE LAYERS: Interrupt handlers, Device drivers, Device-Independent I/O software.

FILE SYSTEMS


SLE: File system reliability

UNIT 6

DEADLOCKS

INTRODUCTION TO DEADLOCKS: Conditions for deadlock, Deadlock modeling. THE OSTRICH ALGORITHM, DEADLOCK DETECTION AND RECOVERY: Deadlock detection with one resource of each type, Deadlock detection with multiple resource of each type, Recovery from deadlock, DEADLOCK AVOIDANCE: Resource trajectories, Safe and Unsafe States, The Banker’s algorithm for a single resource, The banker’s algorithm for multiple resources, DEADLOCK PREVENTION: Attacking the mutual exclusion condition, Attacking the Hold and Wait condition, Attacking the No Preemption condition.

SLE: Attacking the Circular Wait condition

TEXT BOOKS:


**REFERENCE BOOKS:**

DATABASE MANAGEMENT SYSTEMS (3:0:2)

Sub code : CS0414
Hrs / week : 05
SEE Hrs : 03 Hrs

Prerequisite: File Systems

Course Outcome:
On successful completion of the course the students will be able to

1. Choose the available different database models for a given application
2. Design the database using Entity Relationship model and Normalization
3. Apply the SQL for the query to fetch the existing data from the tables and to operate on transactions
4. Create triggers, constraints and objects in MYSQL
5. Demonstrate the use of Indexes and Views
6. Detect the failures and recovery from the different types of failures using logs

UNIT 1

Introduction: Advantages of using a DBMS approaches, Definition of schema, data model and instances, Three-schema architecture and data independence, Different Database Models, Interfacing with a Database, The Mechanics, Disk versus Main-Memory.

An Example Database: The Everest Books Database

Relational Databases (with a MySQL Flavor): MySQL Database System, Database Organization, Creating and using the Database, SLE: Steps in Designing a Database.

9 Hours

UNIT 2

Manipulating the Database: Example Tables, Relational Algebra.
**Database Design 1:** *The Entity-Relationship Model* - Entity, Entity Sets, Attributes, Relationships, Constraints on Relationship types, Weak entity type

**Database Design 2:** Definition of Functional Dependencies, Normal Forms Based on Primary Keys - 1NF, 2NF, 3NF, BCNF

SLE: Notations for ER Diagrams

9 Hours

**UNIT 3**

SQL: Basics, Data Definition Language, Data Manipulation Language, Stored Routines, Formatting and SQL.

Transactions: Informally Speaking, Transactions in SQL, Transaction Semantics, Serializability, Locks, SQL Isolation Levels, Transactions in MySQL

SLE: Orders, Invoices & Reports

8 Hours

**UNIT 4**

Constraints: Constraints in SQL, Constraint Check Time.

Triggers: Triggers in SQL.

Objects: Objects in SQL, Referencing Objects

SLE: Comparing Objects.

9 Hours

**UNIT 5**

Indexes: How does an Index Work?, Types of Indexes, Different Types of Indexes, Hash Indexes, Indexes in MySQL,

Views: Two Implementations of Views, Views in MySQL, Examples.

SLE: Some Thoughts on Indexing.

9 Hours

**UNIT 6**

Spatial Databases: MySQL, Example
Logs and Recovery: Failure Types, Logs, Log Details, Some Log Characteristics, Database Recovery, Database Recovery Example, MySQL.

SLE: Spatial Databases: Example

TEXT BOOK:


   (1.1, 1.6, 2.2, 3.3, 3.4.3, 3.5, 10.2.1, 10.3, 10.5)

REFERENCE BOOKS:


SOFTWARE ENGINEERING (4:0:0)

Sub Code : CS0417
Hrs / Week : 04
SEE Hrs : 3 Hrs

Course Outcome

On successful completion of the course the students will be able to

1. Explain the fundamental concepts of software engineering Product and Process.
2. Provide Development guidelines and philosophy of Agile view of process and system engineering.
3. Identify the key steps in gathering requirements of a problem and building an Analysis Model.
4. Encompasses the set of principles, concepts and practices for the Design of high quality system.
5. Develop the Testing Skills for validating system functions.
6. Identify the effective software project management and estimation methods.

UNIT 1


SLE: The changing nature of Software.

9 Hours
UNIT 2


SLE: Computer Based Systems. 8 Hours

UNIT 3


SLE: Validating Requirements. 9 Hours

UNIT 4


SLE: The Design Model. 9 Hours

UNIT 5


SLE: Validating Testing. 8 Hours
UNIT 6


SLE: Software Risk.

9 Hours

TEXT BOOK


REFERENCE BOOKS


2. Software Engineering Theory and Practice - Shari Lawrence Pfleeger, Joanne M.


COMPUTER NETWORKS (4:0:0)

Sub code : CS0418  
Hrs/Week : 04  
SEE Hrs : 03  
CIE : 50% Marks  
SEE : 50% Marks  
Max Marks : 100

Prerequisite: Data Communications and Networking

Course Outcome

On successful completion of the course the students will be able to

1. Identify of the different layers in the networking software along with their functionality.
2. Explain the functions of the Networking layer.
3. Acquire knowledge about working of different Routing algorithms.
4. Identify functions of the transport layer such as providing both connection oriented (TCP) and connectionless (UDP) communication from host to host. Appreciation of the need to control congestion and ways of doing it.
5. Discuss the need to store information on WWW and the different kinds of document formats needed.
6. Acquire knowledge of the prominent application layer protocols making use of the transport layer services to provide quality utility tools along with their working.

UNIT 1


SLE: Transition from IPv4 to IPv6.

9 hours

UNIT 2

Network Layer-2: Address Mapping: ARP, RARP, BOOTP, ICMP, Delivery: Direct verses Indirect delivery; Forwarding: Forwarding Techniques, Forwarding Process and Routing Table.
UNIT 3


SLE: Path Vector Routing Algorithm

UNIT 4


SLE: Congestion control in TCP.

UNIT 5

Application Layer: Domain Name System: Name Space, Domain Name Space, Distribution of Name space, DNS in the Internet, Resolution, DNS messages, Types of Records, Registrars, Dynamic DNS, Encapsulation; Electronic Mail; WWW and HTTP: Architecture, Web Documents, HTTP; File Transfer: File Transfer Protocol.

SLE: Anonymous FTP.

UNIT 6

Real Time Protocols: RTP; RTP packet format, UDP port; RTCP; Voice over IP: SIP, H.323; Network Management: Network management system: Configuration, Fault, Performance, Security and Accounting Management; Simple Network Management.
Protocol (SNMP): Concept, Management Components, Structure of Management Information, Management Information Base (MIB), Lexicographic Ordering, SNMP, Messages.

SLE: UDP ports, Security.

9 Hours

TEXT BOOKS:


REFERENCE BOOKS:


DATABASE LABORATORY (0:0:3)

Sub code : CS0108  Hrs/week : 03

Course Outcome:

At the end of this course the student will be able to

1. Explain his/her knowledge of database technology, its importance, its architectures, and the central role Database technology plays in Information Systems.

2. Apply appropriate development methodologies of data analysis, design and use appropriate modeling techniques for databases.

3. Demonstrate query facilities to formulate queries and manipulate the database e.g. Structured Query Language (SQL)

4. Appreciate the issues underlying database implementation in any database

5. Concentrate on a methodology for good database design and practical experience in designing and implementing standalone database system.
COMPUTER NETWORKING LABORATORY (0:0:3)

Sub code: CS0109

Course Outcome:
At the end of this course the student will be able to
(a) (i) Implement the basic Concepts of Networking like
   (1) Usage of different devices,
   (2) Connecting systems.

(ii) Use any packet analyzer to determine the various
    Networking factors.

(iii) Prepare a write up about the CSLAB NETWORK Set-up,

(b) Develop Networks using NS2 or equivalent simulators in world
    scenarios in a project based approach.
**CLOUD COMPUTING**

Sub code : CS0437  
Hrs / week : 04  
SEE Hrs : 3 Hrs  

CIE : 50 %  
SEE : 50 %  
Max. Marks : 100

**Course Outcome**

On successful completion of the course the students will be able to

1. Analyze what is cloud computing and what is not.
2. Develop an insight into the basic cloud architecture and cloud services
3. Use the concepts of abstraction and virtualization to plan a cloud environment
4. Explain clear insight into the working of a cloud, with special reference to google cloud services.
5. Take decisions on cloud deployment with special reference to cloud security issues
6. Acquire the working knowledge of Service Oriented Architectures and Cloud based storages.

**UNIT 1**

**Introduction**


**SLE: Measuring Cloud Criteria**  
8 Hours
UNIT 2
Understanding Cloud Architecture


SLE: Virtual Appliances

UNIT 3
Understanding Services, Applications, Abstraction and Virtualization

Defining Infrastructure as Service (IaaS), Defining Platform as a Service (PaaS), Defining Software as a Service (SaaS), Defining Identity as a Service (IDaaS), Defining Compliance as a Service (CaaS), Using Virtualization Technologies, Load Balancing and Virtualization, Understanding Hypervisors, Understanding Machine Imaging, Porting Applications.

SLE: Load Balancing Techniques used in Cloud Computing

UNIT 4
Using Google Web Services and Managing the Cloud


SLE: Google Web Toolkit, Google App Engine

UNIT 5
Understanding Cloud Security

INTRODUCTION TO C# PROGRAMMING AND .NET CONCEPTS(4:0:0)

Sub code : CS0426
Hrs/week : 04
SEE Hrs : 03 Hours

CIE : 50% Marks
SEE : 50% Marks
Max. Marks : 100

Course Outcome

On successful completion of the course the students will be able to

1. Explain the basics of .Net platform and the role of base class libraries, role of common intermediate language and namespaces and also to Define and deploy the different command line compiler options.

2. Discuss the fundamentals of c# and to build the basic c# program using different constructs.

3. Review the basic pillars of object oriented programming concepts and apply the exception handling technique to handle different types of errors.

4. Provide the knowledge about basics of object lifetime and to define the use of interfaces and collections.

5. Explain the callback interfaces, delegates, advanced C# keywords and its implementation.

6. Build and deploy local and shared assemblies.

UNIT 1

BUILDING C# APPLICATIONS: The Role of the Command Line
Compiler (csc.exe), Building C# Application using csc.exe Working
with csc.exe Response Files, Generating Bug Reports, Remaining
C# Compiler Options, The Command Line Debugger (cordbg.exe)
Using the, Visual Studio .NET IDE, C# "Preprocessor" Directives.

SLE: Deploying the .NET Run-time, An Interesting Aside The
System, EnvironmentClass.

9 Hours

UNIT 2

C# LANGUAGE FUNDAMENTALS: The Anatomy of a Basic C#
Class, Creating objects: Constructor Basics, The Composition of a
C# Application, Default Assignment and Variable Scope, The C#
Member Initialization Syntax, Basic Input and Output with the
Console Class, Understanding Value Types and Reference Types,
The Master Node System.Object, The System Data Types (and C#
Aliases), Converting Between Value Types and Reference Types
Boxing and Unboxing, Defining Program Constants, C# Iteration
Constructs, C# Controls Flow Constructs, The Complete Set of C#
Operators, Defining Custom Class Methods, Understating Static
Methods, Methods Parameter Modifies, Array Manipulation in C#,
String Manipulation in C#, C# Enumerations, Defining Structures in
C#, Defining Custom Namespaces.

SLE: Programming using basic constructs of c#

9 Hours

UNIT 3

OBJECT-ORIENTED PROGRAMMING WITH C#: Forms Defining
of the C# Class, Definition the "Default Public Interface" of a Type,
Recapping the Pillars of OOP, The First Pillars: C#’s Encapsulation
Services, Pseudo-Encapsulation: Creating Read-Only Fields, The
Second Pillar: C#’s Inheritance Supports, keeping Family Secrets:
The "Protected" Keyword, Nested Type Definitions, The Third
Pillar: C#’s Polymorphic Support, Casting Between. EXCEPTIONS
: Ode to Errors, Bugs, and Exceptions, The Role of .NET Exception
Handling, the System, Exception Base Class, Throwing a Generic
Exception, Catching Exception, CLR System – Level Exception
(System, System Exception), Custom Application-Level Exception
(System, System Exception), Handling Multiple Exception, The
Family Block, the Last Chance Exception Dynamically Identifying Application – and System Level Exception Debugging System.

SLE: Exception Using VS.NET

UNIT 4


INTERFACES AND COLLECTIONS: Defining Interfaces Using C# Invoking Interface Members at the object Level, Exercising the Shapes Hierarchy, Understanding Explicit Interface Implementation, Interfaces As Polymorphic Agents, Building Interface Hierarchies, Implementing, Implementation, Interfaces Using VS .NET, understanding the IConvertible Interface, Building a Custom Enumerator (IEnumerarble and Enumerator), Building Cloneable objects (ICloneable), Building Comparable Objects (IComparable), Exploring the system. Collections Namespace, Building a Custom Container (Retrofitting the Cars Type).

SLE: Building custom Interfaces and collections using VS.NET

UNIT 5

CALLBACK INTERFACES, DELEGATES, AND EVENTS, ADVANCED TECHNIQUES: Understanding Callback Interfaces, Understanding the .NET Delegate Type, Members of System. Multicast Delegate, The Simplest Possible Delegate Example, Building More a Elaborate Delegate Example, Understanding Asynchronous Delegates, Understanding (andUsing)Events.

The Advances Keywords of C#, A Catalog of C# Keywords Building a Custom Indexer, A Variation of the Cars Indexer Internal Representation of Type Indexer. Using C# Indexer from VB .NET. Overloading operators, The Internal Representation of Overloading Operators, interacting with Overload Operator from Overloaded-Operator- Challenged Languages, Creating Custom Conversion Routines, Defining Implicit Conversion Routines, The Internal Representations of CustomsConversionRoutines.

8 Hours

9 Hours
UNIT 6


SLE: Building a custom assemblies.

TEXT BOOKS


REFERENCE BOOKS:

EMBEDDED SYSTEMS (4:0:0)

Sub code : CS0428  
Hrs / week : 05  
SEE Hrs : 03 Hours  
CIE : 50% Marks  
SEE : 50% Marks  
Max. Marks : 100

Course Outcome

On successful completion of the course the students will be able to

1. Describe the fundamentals of Embedded Systems.

2. Develop necessary skills to understand and design an embedded system application.

3. Identify the challenges of Concurrent Process and its solutions.

4. Compare the advantages of the software Architectures and design an embedded system application using an RTOS.

5. Relate the tasks with inter task communication primitives like, semaphores and message queues and design the hardware along with the choice of the RTOS for the application at hand.

6. Acquire the knowledge about the Embedded Software Development Tools and Model the requirements of an application as a set of tasks.

UNIT 1

Custom single-purpose processor design; RT-level custom single-purpose processor design; Optimizing custom single-purpose processors: optimizing the FSMD, Optimizing the data-path, optimizing the FSM.

SLE : Optimizing the Original Program.

8 Hours

UNIT 2

Timers, counters, and watchdog timers.

State machine models: Introduction; finite-state machines (FSM); Finite-state machines with data path model (FSMD); Using state
machines: Describing a system as a state machine, Comparing state machine and sequential program models, Capturing a state machine model in a sequential programming language; Hierarchical/Concurrent state machine model (HCFSM) and the State charts language; Program state machine model (PSM); The role of an appropriate model and language.

**SLE**: An introductory example, A basic state machine model.

**UNIT 3**

**Concurrent process models**: Concurrent processes: Process create and terminate, Process suspend and resume, Process join; Communication among processes: Shared memory, Message passing; Synchronization among processes: Condition variables, Monitors.

**Interrupts**: The Shared Data Problem; Interrupt latency.

**SLE**: Interrupt Basics.

**9 Hours**

**UNIT 4**

**Survey of Software Architecture**: Round Robin with Interrupts, Function Queue Scheduling Architecture; Real Time Operating System Architecture, Selecting architecture.

**Introduction to RTOS**: Tasks and Task States, Tasks and Data, Semaphores and shared data.

**SLE**: Round Robin.

**8 Hours**

**UNIT 5**


**SLE**: Saving Power.

**9 Hours**
UNIT 6
Embedded Software Development Tools: Host and Target embedded software into the Target System
Debugging Techniques: Testing on host machine, Instruction set simulators, macros and tools.
An example system: What the program does?
SLE: Environment in which the program operates.

TEXT BOOKS
1. Embedded System Design: A Unified Hardware/Software Introduction - Frank Vahid, Tony Givargis, John Wiley & Sons, Inc.2002 (Articles: 2.4, 2.5, 2.6; 3.8; 4.2)
2. An Embedded Software Primer - David E. Simon: Pearson Education, 1999. (Chapters: 4,5,6,7,8,9,10 and 11)

REFERENCE BOOKS
INTRODUCTION TO ANDROID PROGRAMMING
(4:0:0)

Sub code : CS0443
Hrs / week : 04
SEE Hrs : 4 Hours

CIE : 50 %
SEE : 50 %
Max. Marks : 100

Pre-requisite: DBMS, Java Programming

Course outcomes:

On successful completion of the course the students will be able to

1. Able to understand the basic versions, the evolution of Android as a Mobile OS, create simple apps and apply different styles.

2. Learn the User Interface of Android.

3. Manipulate the UI with different common elements.

4. Apply the knowledge to a real time Internet download app.

5. Able to learn how to persist data using preferences.

6. Apply the knowledge of DBMS to persist data.

Unit 1

GETTING STARTED WITH ANDROID PROGRAMMING:


ACTIVITIES, FRAGMENTS AND INTENTS:

Understanding Activities, Applying Styles and Themes to an Activity, Hiding the Activity Title, Displaying a Dialog Window, Displaying a Progress Dialog, Displaying a More Sophisticated Progress Dialog, Linking Activities Using Intents, Resolving Intent Filter, Collision, Returning Results from an Intent, Passing Data
Using an Intent Object, Fragments, Adding Fragments Dynamically, Life Cycle of a Fragment, Interactions between Fragments, Calling Built-In Applications Using Intents, Understanding the Intent Object, Using Intent Filters, Adding Categories, Displaying Notifications

SLE: Designing a Android Application

UNIT 2

GETTING TO KNOW THE ANDROID USER INTERFACE:

Understanding the Components of a Screen, Views and ViewGroups, LinearLayout, AbsoluteLayout, TableLayout, RelativeLayout, FrameLayout, ScrollView, Adapting to Display Orientation, Anchoring Views, Resizing and Repositioning, Managing Changes to Screen Orientation, Persisting State Information during Changes in Configuration, Detecting Orientation Changes, Controlling the Orientation of the Activity, Utilizing the Action Bar, Adding Action Items to the Action Bar, Customizing the Action Items and Application Icon, Creating the User Interface Programmatically, Listening for UI Notifications, Overriding Methods Defined in an Activity, Registering Events for Views


SLE: Designing User Interface

9 Hours

UNIT 3

DISPLAYING PICTURES AND MENUS WITH VIEWS: Using Image Views to Display Pictures, Gallery and ImageView Views, ImageSwitcher, GridView, Using Menus with Views, Creating the Helper Methods, Options Menu, Context Menu, Some Additional
UNIT 5

USING INTERNET RESOURCES:

Downloading and Parsing Internet Resources, Connecting to an Internet Resource, Parsing XML Using the XML Pull Parser, Creating an Earthquake Viewer, Using the Download Manager, Downloading Files, Customizing Download Manager Notifications, Specifying a Download Location, Cancelling and Removing Downloads, Querying the Download Manager, Using Internet Services, Connecting to Google App Engine, Best Practices for Downloading Data Without Draining the Battery

SLE: Using Internet Resources to develop Android Applications

UNIT 5

FILES, SAVING STATE, AND PREFERENCES:

UNIT 6

DATABASES AND CONTENT PROVIDERS: Introducing Android Databases, SQLite Databases, Content Providers, Introducing SQLite, Content Values and Cursors, Working with SQLite Databases, Introducing the SQLiteOpenHelper, Opening and Creating Databases Without the SQLite Open Helper, Android Database Design Considerations, Querying a Database, Extracting Values from a Cursor, Adding, Updating, and Removing Rows, Inserting Rows, Updating Rows, Deleting Rows, Creating Content Providers, Registering Content Providers, Publishing Your Content Provider’s URI Address, Creating the Content Provider’s Database, Implementing Content Provider Queries, Content Provider Transactions, Storing Files in a Content Provider, A Skeleton Content Provider Implementation, Using Content Providers, Introducing the Content Resolver, Querying Content Providers, Querying for Content Asynchronously Using the Cursor Loader, Introducing Loaders, Using the Cursor Loader, Adding, Deleting, and Updating Content, Inserting Content, Deleting Content, Updating Content, Accessing Files Stored in Content Providers, Creating a To-Do List Database and Content Provider

SLE: Usage of SQLite Databases

TEXT BOOKS

1. Wei-Meng Lee, Beginning Android 4 Application Development, Wrox, Wiley India Edition. (Chapters: 1, 2, 3, 4, 5)

2. Reto Meier, Professional Android 4 Application Development, Wrox, Wiley India Edition(Chapters: 6, 7, 8)

REFERENCE BOOKS:

**BIG-DATA ANALYTICS (4:0:0)**

<table>
<thead>
<tr>
<th>Sub code</th>
<th>CIE</th>
<th>SEE</th>
<th>Max. Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS0444</td>
<td>50 %</td>
<td>50 %</td>
<td>100</td>
</tr>
</tbody>
</table>

**Course Outcome**

On successful completion of the course the students will be able to

1. Analyze several key technologies used in manipulating, storing, and analyzing big data.
2. Acquire clear understanding of Hadoop.
3. Identify the tools that provide SQL-like access to unstructured data.
5. Manage Big Data and analyze Big Data.
6. Apply tools and techniques to analyze Big Data.

**UNIT 1**

**INTRODUCTION TO BIG DATA**


**SLE:** Big Data Analytics Applications

8 hrs

**UNIT 2**

**BIG DATA TECHNOLOGIES**

Hadoop’s Parallel World – Data discovery – Open source technology for Big Data Analytics – cloud and Big Data – Predictive Analytics – Mobile Business Intelligence and Big Data – Crowd Sourcing Analytics – Inter- and Trans-Firewall Analytics - Information Management.
UNIT 3
PROCESSING BIG DATA
Integrating disparate data stores - Mapping data to the programming framework - Connecting and extracting data from storage - Transforming data for preprocessing - Subdividing data in preparation for Hadoop Map Reduce.
SLE: Data Preparation for Map Reduce

UNIT 4
HADOOP MAPREDUCE
Employing Hadoop Map Reduce - Creating the components of Hadoop Mapreduce jobs - Distributing data processing across server farms - Executing Hadoop Map Reduce jobs - Monitoring the progress of job flows - The Building Blocks of Hadoop Map Reduce - Distinguishing Hadoop daemons - Investigating the Hadoop Distributed File System Selecting appropriate execution modes: local, pseudo-distributed, fully distributed.
SLE: Applications of Hadoop Mapreduce

UNIT 5
ADVANCED ANALYTICS PLATFORM
SLE: Big Data Convergence

UNIT 6
BIG DATA TOOLS AND TECHNIQUES
Installing and Running Pig - Comparison with Databases - Pig Latin - User-Define Functions - Data Processing Operators -

SLE: Installing and Running Hive

TEXTBOOKS


INTRODUCTION TO DATA MINING (2:0:0)

Sub code : CS0201  
CIE : 50% Marks

Hrs/week : 02  
SEE : 50% Marks

SEE Hrs : 02 Hours  
Max. Marks : 50

Course Outcome

On successful completion of the course the students will be able to

1. Understand Data Mining concepts and applications of Data Mining Applications.
2. Understand Data Warehouse Implementation.
3. Discuss Data Preprocessing Techniques.

UNIT 1

Introduction to Data Mining:

Motivation and importance, What is Data Mining, Relational Databases, Data Warehouses, Transactional Databases, Advanced Database Systems and Advanced Database Applications, Data Mining Functionalities, Interestingness of a pattern Classification of Data Mining Systems.

SLE: Major issues in Data Mining.  
9 Hours

UNIT 2

Data Warehouse and OLAP Technology for Data Mining

What is a Data Warehouse? Multi-Dimensional Data Model, Data Warehouse Architecture, Data Warehouse Implementation, Development of Data Cube Technology.

SLE: Data Warehousing to Data Mining  
9 Hours

UNIT 3

Data Preprocessing

Why Pre-process the Data? Data Cleaning, Data Integration and Transformation Data Reduction, Discretization.

SLE: Concept Hierarchy Generation  
8 Hours

104
TEXT BOOK

1. Data Mining Concepts and Techniques, Jiawei Han and Micheline Kamber, Morgan Kaufman Publications

REFERENCE BOOKS

1. Introduction to Data Mining, Adriaan, Addison Wesley Publication

2. Data Mining Techniques, A.K.Pujari, University Press
WEB TECHNOLOGIES AND APPLICATIONS (3:0:2)

Sub code: CS0420  
CIE: 50%

Hrs/week: 05  
SEE: 50%

SEE Hrs: 3 Hours  
Max. Marks: 100

Prerequisite: Object Oriented Programming Concepts

Course outcome

On successful completion of the course the students will be able to

1. Explain why Java is important in the internet, how Java is different from other language, cite examples on using data types, variables, operators, control statements. Write programs using arrays and vectors

2. Compare and demonstrate the different category of classes, Recognize group of classes with different visibility control to create package; Prediction of errors and handling them using exceptional handling;

3. Analyze working of I/O operations, compare functions of string and stringbuffer class

4. Prepare web pages using HTML/XHTML and CSS.

5. Creation of dynamic web pages using CGI script. Illustrate how to create database using MYSQL and link them with perl script.

6. Demonstrate how to embed php into HTML and how to link php with MYSQL to create dynamic pages

UNIT 1

The Java Language – 1: The Genesis of Java: Java’s Lineage, The creation of Java, Why Java is important to the Internet? Java’s Magic, Java features; An overview of Java Programming: Object oriented programming, simple java program, Lexical issues; Data types, Variables and Arrays: Simple types, Literals, Variables, Type conversions and Casting, Arrays, Operators, Control Statements, Introduction to Java Classes: Classes, Object, Methods, Constructors, The this keyword, Garbage collection, The Finalize() Method, A Stack Class.
SLE: Programs on Arrays in java
Implementation on above important concepts.

UNIT 2

The Java Language – 2: Methods and Classes; Inheritance; Inheritance Basic, Using Super, Creating Multilevel hierarchy, When Constructors are Called, Method Overriding, Dynamic Method Dispatch, Using Abstract Classes, Using Final, the Object Class; Java Packages and Interfaces; Exception Handling; Multithreading Programming.

SLE: Creation of User defined package
Implementation on above important concepts.

UNIT 3


SLE: Programs on Strings
Implementation on above important concepts.

UNIT 4

XHTML, XML and CSS: XHTML: Basic syntax; Standard XHTML document structure; Basic text markup, Images; Hypertext Links; Lists; Tables; Forms; Frames; CSS: Introduction; Levels of style sheets; Style specification formats; Selector forms; Property value forms; Font properties; List properties; Color; Alignment of text; The Box model; Background images; The and tags; Conflict resolution. XML: Introduction; Syntax; Document structure; Document Type definitions; Namespaces; XML schemas; Displaying raw XML documents; Displaying XML documents with CSS.
SLE: Creation of Web Pages
Implementation on above important concepts.

UNIT 5

CGI and MYSQL: Introduction to Apache Web Server. Introduction, Starting, stopping and Restarting Apache, Apache Log Files, MYSQL: Introduction, Tutorial, DBI, Table Joins, Loading and Dumping a Database. CGI - Introduction, Apache Configuration, A Simple CGI Program, what can go wrong in the program? CGI.pm: Introduction, CGI.pm HTML Shortcuts, Information received by the CGI program, Form Widget Methods, Security consideration, A note on die().

SLE: Programs on PERL
Implementation on above important concepts.

UNIT 6

PHP and MYSQL: PHP Introduction, Embedding PHP into HTML, Configuration, Quick examples, Language Syntax, Built-IN PHP Functions, PHP and MySQL.

SLE: Programs on PHP
Implementation on above important concepts.

Note: Implementation of the concepts taught in the theory classes will be done in laboratory sessions.

TEXT BOOKS:


3. Open Source Web Development with Lamp, James Lee and Brent Ware, Pearson Education, 2009
Sub code : CS0301
Course outcome
On successful completion of the course the students will be able to:

1. Formulate the problem.
2. Learn to perform analysis and check the feasibility of solution in terms of time, cost and operations.
3. Carry out Extensive Literature survey, identify software and Hardware requirements.
4. Describe costs and benefits and submit a preliminary plan.
MANAGEMENT AND ENTREPRENEURSHIP (4:0:0)

Sub Code: CS0424  CIE : 50%
Hrs/week: 04  SEE : 50%
See Hrs: 3 Hrs  Max Marks: 100

COURSE OUTCOME

On successful completion of the course the students will be able to

1. Provide the information about management as science, art, administration and as a key approach in doing a assigned work.

2. Explain planning and organizing, with planning students learn its importance and hierarchy and with organizing students learn about departmentation, committees, centralization and decentralization.

3. Explain the importance of staffing directing and controlling , meaning and steps in controlling.

4. Identifying who is an entrepreneur, his functions and types and identifying based on geography.

5. Provide knowledge of small scale industry and institutional support

6. Prepare project Project Selection; Project Report; Need and Significance of Report;

UNIT 1

Management: Introduction - Meaning - nature and characteristics of Management, Scope and functional areas of management - Management as a science, art or profession Management & Administration - Roles of Management, Levels of Management,

SLE : Development of Management Thought - early management approaches - Modern management approaches, case studies on administration and management.

7 Hours
SEMINAR (0:0:2)

Sub code : CS0110

Course outcome

On successful completion of the course the students will be able to

1. Identify a research topic of current interest.
2. Present the same formally.
PROJECT WORK – II (0:0:12)

Sub code : CS0601

Course outcome

On successful completion of the course the students will be able to:

1. Analyze real world /scientific problems and convert them to computable algorithms.

2. Make a comparative study of the complexities of the various approaches to solve such algorithms.

3. Evaluate the available technological alternatives (like client server, cloud, mobile, stand-alone etc) to implement the same.

4. Evaluate the various hardware /software/networking/ financial requirements for the solution.

5. Articulate the issues effectively with fellow members as well as end users.

6. Code the solutions using appropriate languages, port them to different platforms and operate & maintain them effectively.
# CRYPTOGRAPHY (3:0:2)

<table>
<thead>
<tr>
<th>Sub code</th>
<th>CS0421</th>
<th>CIE</th>
<th>50 % Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hrs / week</td>
<td>05</td>
<td>SEE</td>
<td>50 % Marks</td>
</tr>
<tr>
<td>SEE Hrs</td>
<td>03 Hours</td>
<td>Max. Marks</td>
<td>100</td>
</tr>
</tbody>
</table>

## Course Outcome

On successful completion of the course the students will be able to:

1. Explain the foundations of Cryptography and network security. Critical evaluation of the risks and threats to networked computers. Demonstration of the various mechanisms to protect data and their limitations.

2. Demonstrate detailed knowledge of the role of encryption to protect data using DES.

3. Demonstrate advanced encryption technique like AES.

4. Familiar with cryptographic techniques for secure (confidential) communication of two parties over an insecure (public) channel using public key algorithms.

5. Verify the integrity of the messages transmitted via an insecure channel and unique identification of the originator of any message using MAC.

6. Explain the different key management and digital signature systems.

## UNIT 1

**Introduction to Symmetric-Key Encipherment**: Security Goals, Cryptographic Attacks, Services and mechanism, Techniques.

**Traditional Symmetric-Key Ciphers**: Introduction, Substitution Ciphers, Transposition Ciphers.

**SLE**: Stream and Block Ciphers.

9 Hours

## UNIT 2

**Mathematics of Symmetric-Key Cryptography**: Algebraic Structures, GF(2^n) Fields. **Introduction to Modern Symmetric-Key**
Ciphers: Modern Block Ciphers, Modern Stream Ciphers.

Data Encryption Standard (DES): Introduction, DES Structure (overview only), Security of DES.

SLE: Multiple DES-Conventional Encryption Algorithms

UNIT 3

Advanced Encryption Standard (AES): Introduction, Transformations, Key expansion, The AES Ciphers, Examples, Analysis of AES.

Encipherment Using Modern Symmetric-Key Ciphers: Use of Modern Block Ciphers (overview only)

SLE: Use of Stream Ciphers (overview only)

UNIT 4

Asymmetric-Key Cryptography: RSA Cryptosystems, Elliptic curve crypto systems (overview only)

SLE: ElGamal Cryptosystem.

UNIT 5

Message Integrity and Message Authentication

Message Integrity, Random Oracle Model, Message Authentication.

Digital Signature: Comparison, Process, Services, Attacks of Digital Signature, Digital Signature Schemes.

SLE: Variations and Applications.

UNIT 6


Key Management: Symmetric-Key Distribution, Kerberos, Symmetric-key Agreement.
SLE : Public-Key Distribution.

8 Hours

TEXT BOOK


REFERENCE BOOKS


COMPUTER CONCEPTS AND ‘C’ PROGRAMMING

(4:0:0)

Sub code : CS0401
Hrs/week : 04
SEE Hrs : 03 Hours

CIE : 50% Marks
SEE : 50% Marks
Max. Marks : 100

Course Outcomes:

On successful completion of the course, the students will be able to

1. Identify the various terminologies and basic interaction methods in the computers

2. Describe the basic methods of storing and processing data and also the role of operating systems and networking.

3. Analyse problems and convert them to flowcharts & algorithms.

4. Illustrate the concepts of operators, decision making and branching.

5. Demonstrate the use of arrays & structures.

6. Implement user defined functions, pointers & File management.

UNIT – 1

Introducing Computer Systems:

The Computer defined, Computers for individual users, Computers for organizations, The parts of a computer system, The information processing cycle, Essential computer hardware.

Interacting with Computer:


SLE: Identification of latest printers and its features.

9 Hrs
UNIT – 2

Processing Data:
Transforming Data into Information: How computers represent data?, How computers process data?, Factors affecting processing speed,

Storing Data: Magnetic Storage Devices- How data is stored?, How data is organized on a disk?, Diskettes-Floppy, Hard disk, Optical Storage Devices-CD-ROM, DVD-ROM, Solid-State storage devices- Flash memory and smart card.

Using Operating Systems:

Networks and the Internet:
Networking basics - The uses of a network, Common types of networks, Introduction to Internet, Internet’s major services, WWW, E-mail.

SLE: Importance of Android OS.

UNIT – 3

Algorithms and Flowcharts:
Algorithms, Flowcharts, Writing algorithms and drawing flowcharts for simple exercises.

Introduction to C

Constants, Variables, and Data types:
Structure of C, Characters set, C tokens, Keywords and Identifiers, Constants, Variables, Data types, Declaration of variables.

Operators and Expressions:
Arithmetic operators, Relational operators, Logical operators, Assignment operators.
SLE: Identification of latest browsers used in different devices (computer and mobile).

UNIT – 4

Operators and Expressions:
Increment and decrement operators, conditional operator, Bitwise operators, Special operators, Arithmetic expressions, Evaluation of expressions, Precedence of Arithmetic operators, Operator precedence and associativity.

Managing Input and Output Operations:
Reading a character, Writing a character, Formatted Input, Formatted Output.

Decision making and Branching:
Decision making with if statement, Simple if statement, The if...else statement, Nesting of if...else statements, The else ... if ladder, The switch statement, the?: operator, The Goto statement.

SLE: Type conversions in expressions

UNIT – 5

Decision making and Looping:
The while statement, the do statement, the for statement, Jumps in Loops.

Arrays:
One-dimensional Arrays, Declaration of one-dimensional Arrays, Initialization of one-dimensional Arrays, Two-dimensional Arrays, Initializing two-dimensional Arrays.

Structures:
Defining a Structure, Declaring Structure Variables, Accessing Structure Variables, Structure Initialization.

SLE: List applications of arrays and structures.
UNIT – 6

User-defined Functions:
Need for User-defined Functions, A multi-function Program, Elements of User-defined Functions, Definition of Functions, Return Values and their Types, Function Calls, Function Declaration, Parameter Passing technique-call by value and call by reference, Category of Functions.

Pointers:
Understanding Pointers, Accessing the Address of a Variable, Declaring Pointer Variables, Initialization of Pointer Variables, Accessing a variable through a Pointer.

File Management in C:
Introduction, Defining and Opening a File, Closing File & Input/output operations on Files.

SLE: Applications of pointers.

8 Hrs

TEXT BOOK:


REFERENCE BOOK:

1. Introduction to Computer Science, ITL Education Solutions Ltd.
4. The C programming language, Brian w. Kernighan, Dennis Ritchie, 2nd edition
On Successful completion of this course, the students will be able to

1. Demonstrate the mechanics of creating a Word document, the use of basic functions and formulas in Microsoft Excel and the basic mechanics of creating a PowerPoint presentation.

2. Apply and practice logical ability to solve the problems.

3. Understand C programming development environment, compiling, debugging, linking and executing a program using the development environment.

4. Analyzing the complexity of problems, modularize the problems into small modules and then convert them into programs.

5. Understand and apply the in-built functions and customized functions for solving the problems.

6. Understand and apply the pointers, memory allocation techniques and use of files for dealing with variety of problems.

**PART – A**

1. Create a document using a suitable word processing package, with at least three paragraphs and perform the following operations:
   1. Set left margin 1" and right margin 0.75"
   2. Centre the heading and make it bold. Increase the font size
   3. Underline the specified words in the document and change them to italics
   4. Conduct spell check and correct them suitably
   5. Demonstrate use of numbering and bullets
6. Exchange paragraphs 2 and 3 using cut and paste facility
7. Put suitable headers and footers
8. Count the number of words and lines and demonstrate the use of drawing tools
9. Include suitable logo/emblem/symbol
10. Include the a table

2. You have a monthly income of Rs.11000. Your monthly expenses are Rent-Rs.4000, Food-Rs.3000, Electricity-Rs.40, Phone-Rs. 60, and Cable TV-Rs. 30. Make out a clear spreadsheet with the Monthly Income, the Monthly Expenses listed and summed, the remainder (what’s left over each month) calculated, and the amount left over per day (assuming 30 days in a month). Draw a pie chart to describe the monthly expenses.

3. Generate pay slip for different categories of employees (example: manager, assistant manager, supervisor, clerk etc) using spreadsheet which includes Basicpay, DA, HRA and other allowances along with deductions like PF, Professional Tax, and other deductions. DA and HRA are calculated in terms of percentage which keeps varying. Use formula while performing calculations. Each column and row should display total. Sort the name list in ascending order. Draw the bar chart to describe the DA percentage for different types of employees.

4. Create a PowerPoint presentation consisting of
   Master slide generation
   Minimum 8 slides
   Information about your college
   Include animation, Hyperlinks, Insert Header and Footer, Insert files, images, tables and Charts. Slide show features.

5. Write a C program to find and output all the roots of a given quadratic equation, for non-zero coefficients. (Using if…else statement)

6. Write a C program to simulate a simple calculator that performs arithmetic operations like addition, subtraction, multiplication, and division only on integers. Error message should be reported, if any attempt is made to divide by zero. (Using switch statement)
7. Write a C program to reverse a given six digit integer number and check whether it is a palindrome or not. Output the given number with suitable message. (Using while statement)

8. Write a C program to generate and print first 'N' Fibonacci numbers. (Using do-while statement)

9. Write a C program to find the GCD and LCM of two integers and output the results along with the given integers. Use Euclid's algorithm. (Using looping constructs)

10. Write a C program to find whether a given number is prime or not. Output the given number with suitable message. (Using for statement)

PART – B

11. Write a C program to input N real numbers in ascending order into a single dimension array. Conduct a binary search for a given key integer number and report success or failure in the form of a suitable message.

12. Write a C program to input N integer numbers into a single dimension array. Sort them in ascending order using bubble sort technique. Print both the given array and the sorted array with suitable headings.

13. Write a C program to read two matrices A(M x N) and B(P x Q) and perform addition. Output the input matrices and the resultant matrix with suitable headings and format. (Using two dimension arrays where array size M, N, P, Q ≤ 3)

14. Write C user defined functions:
   1. To input N integer numbers into a single dimension array.
   2. To conduct a linear search.

Using these functions, write a C program to accept the N integer numbers & given key integer number and conduct a linear search. Report success or failure in the form of a suitable message.

15. Write C user defined functions:
   1. To input N integer numbers into a single dimension array.
   2. To sort the integer numbers in descending order using selection sort technique.
To print the single dimension array elements.

Using these functions, write a C program to input N integer numbers into a single dimension array, sort them in descending order, and print both the given array & the sorted array with suitable headings.

16. Write C user defined functions:
To input N real numbers into a single dimension array.
Compute their mean.
Compute their variance
Compute their standard deviation.
Using these functions, write a C program to input N real numbers into a single dimension array, and compute their mean, variance & standard deviation. Output the computed results with suitable headings.

17. Write a C program to read a matrix A(M x N) and to find the following using user defined functions:
   1. Sum of the elements of the specified row
   2. Sum of the elements of the specified column
   3. Sum of all the elements of the matrix
Output the computed results with suitable headings.

18. Write C user defined functions:
   1. To read the elements of a given matrix of size M x N
   2. To print the elements of a given matrix of size M x N
   3. To compute the product of two matrices
Using these functions, write a C program to read two matrices A (M x N) and B (P x Q) and compute the product of A and B after checking compatibility for multiplication. Output the input matrices and the resultant matrix with suitable headings and format. (Using two dimension arrays where array size M, N, P, Q ≤ 3)

19. Write a C program to create an integer variable, Assign value to the variable, create a pointer to this variable and print the value of the variable and address of the variable using pointer.
20. Write a C program to create a structure called Employee with members Name, Job, Salary.

Create a structure variable. Accept the input values for the structure members at run time. Suitably display the same.

21. Write a C program to create a C file called student.rec and store information about a student, in terms of his NAME, USN, BRANCH, SEMESTER.

REFERENCE BOOK:

FUNCTIONAL ENGLISH

(MANDATORY LEARNING COURSE)

Introduction
Grammar
Importance of Languages
Parts of Speech, Usage of Preposition and Article, Punctuation

3 Hrs

Tenses & Degrees of Comparison
Transformation of Sentences
Active-Passive, Affirmative-Negative, Exclamatory-Assertive, Interrogative-Assertive, Kinds of sentences

5 Hrs

Direct-Indirect Speech

5 Hrs

Vocabulary Usage
Homonyms, Correcting Spelling, One-word equivalents

7 Hrs

Precis Writing

3 Hrs

Essay/Report Writing

5 Hrs

Letter Writing
Personal, Official, Applications

5 Hrs

Idioms & Phrases
Meaning & Usage in sentences

5 Hrs

Comprehension
Of an unseen passage

2 Hrs

Elaboration
Expansion of ideas, proverbs

2 Hrs

Presentation
Preparation of materials and presentation – step

3 Hrs

TEXT BOOKS:


REFERENCE BOOKS:

1. English Rank Scorer, G. Sankaran, Addone Publishing group, Thiruvanantapuram, Kerala

2. English Grammar, Wren & Martin


DATA STRUCTURES WITH ‘C’ (4:0:0)

Sub code : CSO404
Hrs/week : 04
SEE Hrs : 03 Hours
CIE : 50% Marks
SEE : 50% Marks
Max. Marks: 100

Prerequisite: Programming in ‘C’

Course Outcome

On Successful completion of the course, the students will be able to:

1. Illustrate basic operations to be performed using pointers and dynamic memory allocations.
2. Demonstrate Abstract Data Types using an Arrays and recursion.
3. Illustrate various operations of different types of Queues.
4. Explain the basic operations of linked list and its implement using dynamic variables.
5. Demonstrate the operations and applications of Binary trees.
6. Apply different sorting and searching methods.

UNIT – 1

Pointer: Understanding pointers, Pointer variables, Accessing address of a variable, Initialization of pointer variables, Accessing variables through pointers, Chain of pointers, Pointer Arithmetic and arrays, Array of pointers, Pointers as function argument, Function returning pointers, Pointers to function, Pointers and structures. Dynamic memory allocation: malloc, calloc, free.

SLE: Pointer expressions, realloc.

9 Hours

UNIT – 2

SLE: Prefix, recursive chains

UNIT – 3
Queue: The Queue and its representation, Circular Queue, Priority Queue
SLE: Applications of Queues.

UNIT – 4
Linked Lists: Inserting and Removing nodes from a list, Implementation of stacks, getnode and free node operations, Linked implementation of queues, Linked list as a data structure, Example of list operations, Header nodes, List in C, Array implementation of lists, Limitations of array implementation, Allocating and freeing dynamic variables, Queues as lists in C, Examples of list operations in C, Circular lists, Stack as a circular lists, Double linked list.
SLE: Linked lists using dynamic variables, Primitive operations on circular lists, Queue as a circular list.

UNIT – 5
SLE: An Example : The Huffman algorithm, C Representation of trees, Tree traversal.

Unit -6
Sorting: Binary tree Sort, Heap sort, Insertion Sort, Shell Sort and Merge Sort.
Searching: Sequential Search, Binary Searches, Hashing-Hash Functions, Collisons, Collision resolution.
SLE: Radix Sorts, Separate chaining.
TEXT BOOKS


REFERENCE BOOKS


OBJECT ORIENTED PROGRAMMING WITH C++

Sub code: CS0405  CIE: 50%
Hrs/week: 04      SEE: 50%
SEE Hrs: 3 Hrs    Max. Marks: 100

Prerequisite: C++

Course Outcome

On successful completion of the course the students will be able to
1. Provide the basic knowledge about classes, syntax, semantics, and terminology of C++.
2. Understand types of constructors, destructor, friend functions, static member functions and copy control.
3. Gain knowledge about function and operator overloading.
4. Learn the concepts of various types of inheritance and its uses.
5. Apply generic programming technique using function and class templates.
6. Handle different types of errors by using exception handling techniques.

UNIT 1

Classes:

Class Definitions and Declarations: Class Definitions, Data Abstraction and Encapsulation, More on Class Definitions, Class Declarations versus Definitions, Class Objects, The Implicit this Pointer, Class Scope: Name Lookup in Class Scope, References, The new and delete Expressions, Argument Passing: Non-Reference Parameters, Reference Parameters, Function Declarations: Default Arguments, Local Objects: Automatic Objects, Static Local Objects, Inline Functions, Class Member Functions: Defining the Body of a Member Function, Defining a Member Function outside the Class.
SLE: Data Abstraction and Encapsulation

UNIT 2

Initialization

Constructors: The Constructor Initializer, Default Arguments and Constructors, The Default Constructor, Implicit Class-Type Conversions, Explicit Initialization of Class Members, Friends, Static Class Members: Static Member Functions, Static Data Members.

Copy Control:


SLE: Defining Our Own Copy Constructor

UNIT 3

Function Overloading:

Overloaded Functions: Overloading and Scope, Function Matching and Argument Conversions, The Three Steps in Overload Resolution, Argument-Type Conversions, Pointers to Functions.

Overload Operations and Conversions:


SLE: Subscript Operator, Member Access Operators

8 Hours

8 Hours

10 Hours
UNIT 4

Object Oriented Programming:

OOP-An Overview. Defining Base and Derived Classes: Defining a Base Class, protected Members, Derived Classes, virtual and Other Member Functions, Public, Private and Protected Inheritance, Friendship and Inheritance, Inheritance and Static Members, Conversions and Inheritance: Derived-to-Base Conversions, Conversions from Base to Derived, Constructors and Copy Control: Base-Class Constructors and Copy Control, Derived-Class Constructors, Copy Control and Inheritance, Virtual Destructors, Virtuals in Constructors and Destructors, Class Scope under Inheritance: Name Lookup Happens at Compile Time, Name Collisions and Inheritance, Scope and Member Functions, Virtual Functions and Scope, Pure Virtual Functions.

SLE: Friendship and Inheritance

10 Hours

UNIT 5

Templates and Generic Programming:

Template Definitions: Defining a Function Template, Defining a Class Template, Template Parameters, Template Type Parameters, Nontype Template Parameters, Writing Generic Programs, Instantiation: Template Argument Deduction, Function-Template Explicit Arguments, Template Compilation Models, Class Template Members: Class-Template Member Functions, Template Arguments for Nontype Parameters, Friend Declarations in Class Templates, Member Templates, Static Members of Class Templates.

SLE: Friend Declarations in Class Templates

8 Hours

UNIT 6

Tools and Special Techniques:

Exception Handling: Throwing an Exception of Class Type, Stack Unwinding, Catching an Exception, Rethrow, The Catch-All Handler, Function Try Blocks and Constructors, Exception Class Hierarchies, Automatic Resource De-allocation.
Multiple and Virtual Inheritance: Multiple Inheritance, Conversions and Multiple Base Classes, Copy Control for Multiply Derived Classes, Class Scope under Multiple Inheritance, Virtual Inheritance, Virtual Base Class Declaration, Special Initialization Semantics.

SLE: The Catch-All Handler

8 Hours

Text Book


Reference Books


DATA STRUCTURES LAB (0:0:3)

Sub code : CS0102                     Hrs/week : 03

Course Outcomes

On successful completion of the course the students will be able to implement Data structures like Stacks, Queues, Linked List, and Trees using C.

a. Implement the basic search and sort algorithms.

b. Appropriate use of a particular data structure and algorithm to solve a problem
OBJECT ORIENTED PROGRAMMING LAB (0:0:3)

Sub code : CS0103

Hrs/week : 03

Course Outcome

On successful completion of the course the students will be able to

1. Understand object-oriented concepts and how they are supported by C++ and implementation issues related to object-oriented techniques.

2. Demonstrate the ability to analyze, use, and create functions, classes, to overload operators, and use inheritance and Pointers when creating or using classes and create templates

3. Demonstrate the ability to understand and use Exception handling and file handling mechanism.

4. Design and write programs that make appropriate use of advanced object-oriented facilities common to many object-oriented languages such as classes, message passing, overloading and inheritance.
Sub code : CS0413
Hrs / week : 05
SEE Hrs : 3 Hrs

Course Outcome

On successful completion of the course the students will be able to

1. Explain the difference between processes and threads and
   Analyze issues of scheduling user-level processes/threads.

2. Analyze the issues and use of locks, semaphores and
   monitors for synchronizing multithreaded systems and implement
   them in multithreaded programs.

3. Analyze the issues and design of memory management.

4. Analyze the design and implementation issues in virtual
   memory management.

5. Explain the concepts of file system and its implementation.

6. Explain concepts of deadlock in operating systems and
   analyze how they can be managed / avoided.

UNIT 1

INTRODUCTION, PROCESSES AND THREADS

WHAT IS AN OPERATING SYSTEM? : The Operating system as
an extended machine, The Operating system as a resource
manager.

PROCESSES: The process model, Process creation, Process
termination, Process hierarchies, Process states. THREADS: The
thread model, Thread usage, implementing threads in user space,
Implementing threads in the kernel, Hybrid implementations,
Scheduler activations. SCHEDULING: Introduction to scheduling,
Scheduling in interactive systems,

SLE: Thread scheduling.

9 Hours
UNIT 2

INTERPROCESS COMMUNICATION

Race conditions, Critical regions, Mutual exclusion with busy waiting, Sleep and wakeup, Semaphores, Mutexes, Monitors, Message passing. CLASSICAL IPC PROBLEMS: The dining philosophers problem, The readers and writers problem.

MEMORY MANAGEMENT

BASIC MEMORY MANAGEMENT: Monoprogramming without swapping or paging, Multiprogramming with fixed partitions, Modeling multiprogramming, Relocation and protection.

SLE: Analysis of multiprogramming system performance

8 Hours

UNIT 3

MEMORY MANAGEMENT, PAGE REPLACEMENT ALGORITHMS & DESIGN ISSUES

SWAPPING: Memory management with bitmaps, Memory management with linked lists. VIRTUAL MEMORY: Paging, Page tables, TLBs—Translation Look aside Buffers, Inverted page tables.


SLE: Belady’s anomaly

9 Hours

UNIT 4

DESIGN ISSUES FOR PAGING SYSTEMS

Local versus Global allocation policies, Load control, Page size, Separate instruction and data spaces, Shared pages, Cleaning policy, Virtual memory interface. IMPLEMENTATION ISSUES: Operating system involvement with paging, Page fault handling, Instruction backup, Backing store, Separation of policy and
mechanism. SEGMENTATION: Implementation of pure segmentation, Segmentation with paging.

SLE: Locking pages in memory.

UNIT 5

INPUT/OUTPUT

I/O SOFTWARE LAYERS: Interrupt handlers, Device drivers, Device-Independent I/O software.

FILE SYSTEMS


SLE: File system reliability

UNIT 6

DEADLOCKS

INTRODUCTION TO DEADLOCKS: Conditions for deadlock, Deadlock modeling. THE OSTRICH ALGORITHM, DEADLOCK DETECTION AND RECOVERY: Deadlock detection with one resource of each type, Deadlock detection with multiple resource of each type, Recovery from deadlock, DEADLOCK AVOIDANCE: Resource trajectories, Safe and Unsafe States, The Banker's algorithm for a single resource, The banker's algorithm for multiple resources, DEADLOCK PREVENTION: Attacking the mutual exclusion condition, Attacking the Hold and Wait condition, Attacking the No Preemption condition.

SLE: Attacking the Circular Wait condition

TEXT BOOKS:

DATABASE MANAGEMENT SYSTEMS (3:0:2)

Sub code : CS0414
Hrs / week : 05
SEE Hrs : 03 Hrs

CIE : 50 %
SEE : 50 %
Max. Marks : 100

Prerequisite: File Systems

Course Outcome:
On successful completion of the course the students will be able to

1. Describe different database models and creation of MySQL database.
2. Design the database using Entity Relationship model and Normalization.
3. Write the SQL query to interact with the database and illustrate transaction concepts.
4. Create triggers, constraints and objects in MySQL.
5. Demonstrate the use of Indexes and Views.
6. Differentiate different types of failures and recovery techniques.

UNIT 1

Introduction: Advantages of using a DBMS approaches, Definition of schema, data model and instances, Three-schema architecture and data independence, Different Database Models, Interfacing with a Database, The Mechanics, Disk versus Main-Memory,

An Example Database: The Everest Books Database

Relational Databases(with a MySQL Flavor): MySQL Database System, Database Organization, Creating and using the Database,

SLE: Steps in Designing a Database.

9 Hours

UNIT 2

Manipulating the Database: Example Tables, Relational Algebra.
Database Design 1: *The Entity-Relationship Model* - Entity, Entity Sets, Attributes, Relationships, Constraints on Relationship types, Weak entity type

Database Design 2: Definition of Functional Dependencies, Normal Forms Based on Primary Keys - 1NF, 2NF, 3NF, BCNF

SLE: Notations for ER Diagrams

UNIT 3

SQL: Basics, Data Definition Language, Data Manipulation Language, Stored Routines, Formatting and SQL.

Transactions: Informally Speaking, Transactions in SQL, Transaction Semantics, Serializability, Locks, SQL Isolation Levels, Transactions in MySQL

SLE: Orders, Invoices & Reports

UNIT 4

Constraints: Constraints in SQL, Constraint Check Time.

Triggers: Triggers in SQL.

Objects: Objects in SQL, Referencing Objects

SLE: Comparing Objects.

UNIT 5

Indexes: How does an Index Work?, Types of Indexes, Different Types of Indexes, Hash Indexes, Indexes in MySQL,

Views: Two Implementations of Views, Views in MySQL, Examples.

SLE: Some Thoughts on Indexing.

UNIT 6

Spatial Databases: MySQL, Example

Logs and Recovery: Failure Types, Logs, Log Details, Some Log
SOFTWARE ENGINEERING(4:0:0)

Sub Code : CS0417          CIE : 50%
Hrs / Week : 04            SEE : 50%
SEE Hrs : 3 Hrs            Max Marks : 100

Course Outcome

On successful completion of the course the students will be able to

1. Describe the fundamental knowledge of Software Engineering Process.

2. Discuss the development guidelines for Agile View of Process and System Engineering.

3. Determine the key steps in gathering requirement for a problem and building an Analysis Model.

4. Provide the set of design principles and practices to build a high quality system.

5. Validate System functions using various Testing Skills

6. Identify an effective estimation model for Software Project Management.

UNIT 1


SLE: The changing nature of Software.

9 Hours
COMPUTER NETWORKS (4:0:0)

Sub Code : CS0418
Hrs / Week : 04
SEE Hrs : 3 Hrs
Prerequisite: Data Communications and Networking

Course Outcomes:

1. Identify the different layers in the networking software with their functionality.
2. Explain the functions of the Networking layer.
3. Acquire the knowledge about working of different Routing algorithms.
4. Identify the functions of the Transport layer and analyze the performances of UDP and TCP protocols.
5. Distinguish between various application layer protocols and analyze the need for information on WWW with document formats.
6. Apply Network Management skills to enhance the network performance focusing on configuration, fault, security and accounting management.

UNIT 1


SLE: Network Address Translation (NVT).

9 hours

UNIT 2

Network Layer Protocols-1: Internet Protocol (IP): Datagram format, fragmentation, options, Security of IPv4 datagrams,

8 hours

SLE: Path Vector Routing Algorithm

UNIT 3


SLE: BGP

9 hours

UNIT 4


8 hours

SLE: Transition from IPv4 to IPv6

UNIT 5

Diagram, Windows in TCP, Flow control, Error control, TCP Congestion control, TCP timers and options.


9 hours

**SLE:** Web documents.

**UNIT 6**

**Application Layer-2:** FTP: Two Connections, Control connection, Data connection, Security for FTP, Electronic Mail: Architecture, Web Based Mail, E-Mail security, Domain Name System: Name space, DNS in Internet, Resolution, Caching, Resource Records, DNS Messages, Registras, DDNS, Security of DNS, Network Management: Introduction, SNMP: Managers and Agents, Management Components, SMI, MIB, and SNMP.

9 hours

**SLE:** ASN.1 Data types.

**TEXT BOOK:**


**REFERENCE BOOKS:**


WEB TECHNOLOGIES AND APPLICATIONS (3:0:2)

Sub code : CS0420
Hrs / week : 05
SEE Hrs : 3 Hours

CIE : 50 %
SEE : 50 %
Max. Marks : 100

Prerequisite: Object Oriented Programming Concepts

Course outcome

On successful completion of the course the students will be able to

1. Explain the importance of java in the internet, its difference compared with other languages and cite examples using data types, variables, operators, control statements, arrays and vectors.

2. Compare and demonstrate the use of different category of classes. Recognize group of classes with different visibility control to create package; identification of errors and handling them using exceptional handling;

3. Analyze working of I/O operations, compare functions of string and StringBuffer class

4. Prepare web pages using HTML/XHTML and CSS.

5. Creation of dynamic web pages using CGI script. Illustrate how to create database using MySQL and link them with perl script.

6. Demonstrate how to embed php into HTML and how to link php with MYSQL to create dynamic pages

UNIT 1

The Java Language – 1: The Genesis of Java: Java’s Lineage, The creation of Java, Why Java is important to the Internet? Java’s Magic, Java features; An overview of Java Programming: Object oriented programming, simple java program, Lexical issues; Data types, Variables and Arrays: Simple types, Literals, Variables, Arrays, Operators, Control Statements, Introduction to Java Classes: Classes, Object, Methods, Constructors, The this keyword, Garbage collection, The finalize() Method, A Stack Class.

SLE: Type conversions and Casting
Implementation on above important concepts.

**UNIT 2**

The Java Language – 2: Methods and Classes; Inheritance; Inheritance Basic, Using Super, Creating Multilevel hierarchy, When Constructors are Called, Method Overriding, Dynamic Method Dispatch, Using Abstract Classes, Using Final, the Object Class; Java Packages and Interfaces; Exception Handling; Multithreading Programming.

**SLE:** Creation of User defined package
Implementation on above important concepts.

9 Hours

**UNIT 3**


**SLE:** StringBuffer Class
Implementation on above important concepts.

9 Hours

**UNIT 4**

XHTML, XML and CSS: XHTML: Basic syntax; Standard XHTML document structure; Basic text markup, Images; Hypertext Links; Lists; Tables; Forms; Frames; CSS: Introduction; Levels of style sheets; Style specification formats; Selector forms; Property value forms; Font properties; List properties; Color; Alignment of text; The Box model; Background images; The and tags; Conflict resolution. XML: Introduction; Syntax; Document structure; Document Type definitions; Namespaces; XML schemas; Displaying raw XML documents; Displaying XML documents with CSS.

**SLE:** Creation of Web Pages
Implementation on above important concepts.

UNIT 5

CGI and MYSQL: Introduction to Apache Web Server, Introduction, Starting, stopping and Restarting Apache, Apache Log Files, MYSQL: Introduction, Tutorial, DBI, Table Joins CGI – Introduction. A Simple CGI Program, what can go wrong in the program? CGI.pm: Introduction, CGI.pm HTML Shortcuts, Information received by the CGI program, Form Widget Methods, Security consideration, A note on die().

SLE: , Loading and Dumping a Database, Apache Configuration

Implementation on above important concepts.

UNIT 6

PHP and MYSQL: PHP Introduction, Embedding PHP into HTML, Configuration, Quick examples, Language Syntax, Built-IN PHP Functions, PHP and MySQL.

SLE: Connecting php to Mysql

Implementation on above important concepts.

Note: Implementation of the concepts taught in the theory classes will be done in laboratory sessions.

TEXT BOOKS


3. Open Source Web Development with Lamp, James Lee and Brent Ware, Pearson Education, 2009
CLOUD COMPUTING (4:0:0)

Sub code : CS0437
Hrs / week : 04
SEE Hrs : 3 Hrs

Course Outcome

On successful completion of the course the students will be able to

1. Categorize different cloud computing models.
2. Describe basic cloud architecture and cloud services
3. Use the concepts of abstraction and virtualization to plan a cloud environment
4. Explain cloud management using Google web services
5. Discuss cloud security issues
6. Determine the relationship between SOA and cloud computing by moving application to Cloud.

UNIT 1

Introduction


SLE: Measuring Cloud Computing cost

8 Hours

UNIT 2

Understanding Cloud Architecture

Exploring the Cloud Computing Stack: Composability, Communication
Protocols, Applications, Connecting to the cloud: Jolicloud Netbook OS.

SLE: Chromium OS: The Browser as an Operating System

8 Hours

UNIT 3
Understanding Services, Applications, Abstraction and Virtualization

Defining Infrastructure as Service (IaaS), Defining Platform as a Service (PaaS), Defining Software as a Service (SaaS), Defining Identity as a Service (IDaaS), Using Virtualization Technologies, Load Balancing and Virtualization, Understanding Hypervisors, Understanding Machine Imaging.

SLE: Defining Compliance as a Service (CaaS), Porting Application

10 Hours

UNIT 4
Using Google Web Services and Managing the Cloud

Exploring Google Applications, Surveying the Google Application Portfolio: Indexed Search, The dark Web, Aggregation and disintermediation, Productivity applications and services, Enterprise offerings, Google Analytics, Google Translate, Administrating the Clouds, Cloud Management Products, Emerging Cloud Management Standards.

SLE: Google Web Toolkit, Google App Engine

8 Hours

UNIT 5
Understanding Cloud Security


SLE: Windows Azure identity standards, Presence

8 Hours
UNIT 6

Understanding Service Oriented Architecture and Moving Applications to the Cloud


SLE: Relating SOA and Cloud Computing, Applications and Cloud APIs, Development of Cloud applications using SOA.

10 Hours

TEXT BOOK:

1. Cloud Computing Bible by Barrie Sosinsky, Wiley India.

REFERENCE BOOKS:


4. Google Apps by Scott Granneman, Pearson

INTRODUCTION TO C# PROGRAMMING AND .NET CONCEPTS (4:0:0)

Sub code : CS0426
Hrs/week : 04
SEE Hrs : 03 Hours

CIE : 50% Marks
SEE : 50% Marks
Max. Marks : 100

Course Outcome

On successful completion of the course the students will be able to
1. Explain the basics of .Net platform and the role of base class libraries, role of common intermediate language and namespaces and also to Define and deploy the different command line compiler options.
2. Discuss the fundamentals of c# and to build the basic c# program using different constructs.
3. Review the basic pillars of object oriented programming concepts and apply the exception handling technique to handle different types of errors.
4. Provide the knowledge about basics of object lifetime and to define the use of interfaces and collections.
5. Explain the callback interfaces, delegates , advanced C# keywords and its implementation .
6. Build and deploy local and shared assemblies.

UNIT 1


BUILDING C# APPLICATIONS: The Role of the Command Line
EMBEDDED SYSTEMS (4:0:0)

Sub code : CS0428
Hrs / week : 05
SEE Hrs : 03 Hours
CIE : 50 %Marks
SEE : 50 %Marks
Max. Marks : 100

Course Outcome

On successful completion of the course the students will be able to
1. Describe the fundamentals of Embedded Systems.
2. Develop necessary skills to understand and design an embedded system application.
3. Identify the challenges of Concurrent Process and its solutions.
4. Compare the software Architectures and design an embedded system application using an RTOS.
5. Relate the tasks with inter task communication primitives like, semaphores and message queues and design the hardware along with the choice of the RTOS.
6. Describe the Embedded Software Development Tools and Model the requirements of an application as a set of tasks.

UNIT 1

Custom single-purpose processor: Hardware; Introduction, Combinational Logic, Sequential Logic, Custom Single-Purpose Processor Design, RT-level custom single-purpose processor design; Optimizing custom single-purpose processors: optimizing the FSMD, Optimizing the data-path, optimizing the FSM.

SLE : Optimizing the Original Program.

9 Hours

UNIT 2

Timers, counters, and watchdog timers.

State machine models: Introduction; finite-state machines (FSM); Finite-state machines with data path model (FSMD); Using state machines: Describing a system as a state machine, Comparing state machine and sequential program models.
Capturing a state machine model in a sequential programming language; Hierarchical/Concurrent state machine model (HCFSM) and the State charts language; Program state machine model (PSM); The role of an appropriate model and language.

**SLE**: An introductory example, A basic state machine model.

**UNIT 3**

**Concurrent process models**: Concurrent processes: Process create and terminate, Process suspend and resume, Process join; Communication among processes: Shared memory, Message passing; Synchronization among processes: Condition variables, Monitors.

**Interrupts**: The Shared Data Problem; Interrupt latency.

**SLE**: Interrupt Basics.

**UNIT 4**

**Survey of Software Architecture**: Round Robin with Interrupts, Function Queue Scheduling Architecture; Real Time Operating System Architecture, Selecting architecture.

**Introduction to RTOS**: Tasks and Task States, Tasks and Data, Semaphores and shared data.

**SLE**: Round Robin.

**UNIT 5**

**Operating Systems Services**: Message Queues, Mailboxes, and Pipes; Timer Functions; Events; Memory Management, Interrupt Routines in an RTOS environment. **Basic Design Using an RTOS**: Overview, Principles, An Example, Encapsulating semaphores and Queues, Hard Real-Time Scheduling Considerations.

**SLE**: Saving Power.
Embedded Software Development Tools: Host and Target Machines, Linker/Locator for Embedded Software, getting embedded software into the Target System

An example system: What the program does
SLE: Environment in which the program operates.

TEXT BOOKS

1. Embedded System Design: A Unified Hardware/Software Introduction - Frank Vahid, Tony Givargis, John Wiley & Sons, Inc. 2002 (Articles: 2.4, 2.5, 2.6, 3.8, 4.2)

2. An Embedded Software Primer - David E. Simon: Pearson Education, 1999. (Chapters: 4, 5, 6, 7, 8, 9, 10 and 11)

REFERENCE BOOKS


INTRODUCTION TO ANDROID PROGRAMMING

Sub code : CS0443
Hrs / week : 04
SEE Hrs : 4 Hours
CIE : 50%
SEE : 50%
Max. Marks : 100

Pre-requisite: DBMS, Java Programming

Course outcomes:

On successful completion of the course the students will be able to

1. Understand the basic versions, the evolution of Android as a Mobile OS, create simple apps and apply different styles.
2. Experiment with the User Interface of Android.
3. Manipulate the UI with different common elements.
4. Apply the knowledge to a real time Internet download application.
5. Able to learn how to persist data using preferences.
6. Apply the knowledge of DBMS to persist data.

Unit 1

GETTING STARTED WITH ANDROID PROGRAMMING:


ACTIVITIES, FRAGMENTS AND INTENTS:

Understanding Activities, Applying Styles and Themes to an Activity, Hiding the Activity Title, Displaying a Dialog Window, Displaying a Progress Dialog, Displaying a More Sophisticated Progress Dialog, Linking Activities Using Intents, Resolving Intent Filter, Collision, Returning Results from an Intent, Passing Data Using an Intent Object, Fragments, Adding Fragments
Dynamically, Life Cycle of a Fragment, Interactions between Fragments, Calling Built-In Applications Using Intents, Understanding the Intent Object, Using Intent Filters, Adding Categories, Displaying Notifications

SLE: Designing a Android Application

9 Hours

UNIT 2

GETTING TO KNOW THE ANDROID USER INTERFACE:

Understanding the Components of a Screen, Views and View Groups, Linear Layout, Absolute Layout, Table Layout, Relative Layout, Frame Layout, Scroll View, Adapting to Display Orientation, Anchoring Views, Resizing and Repositioning, Managing Changes to Screen Orientation, Persisting State Information during Changes in Configuration, Detecting Orientation Changes, Controlling the Orientation of the Activity, Utilizing the Action Bar, Adding Action Items to the Action Bar, Customizing the Action Items and Application Icon, Creating the User Interface Programmatically, Listening for UI Notifications, Overriding Methods Defined in an Activity, Registering Events for Views


SLE: Designing User Interface

9 Hours

UNIT 3

DISPLAYING PICTURES AND MENUS WITH VIEWS: Using ImageView to Display Pictures, Gallery and ImageView Views, Image Switcher, GridView, Using Menus with Views, Creating the Helper Methods, Options Menu, Context Menu, Some Additional Views, Analog Clock and Digital Clock Views, Web View
UNIT 5

USING INTERNET RESOURCES:

Downloading and Parsing Internet Resources, Connecting to an Internet Resource, Parsing XML Using the XML Pull Parser, Creating an Earthquake Viewer, Using the Download Manager, Downloading Files, Customizing Download Manager Notifications, Specifying a Download Location, Cancelling and Removing Downloads, Querying the Download Manager, Using Internet Services, Connecting to Google App Engine, Best Practices for Downloading Data Without Draining the Battery

SLE: Using Internet Resources to develop Android Applications

UNIT 5

FILES, SAVING STATE, AND PREFERENCES:

BIG-DATA ANALYTICS (4:0:0)

Sub code : CS0444
Hrs / week : 04
SEE Hrs : 4 Hours
CIE : 50 %
SEE : 50 %
Max. Marks : 100

Course Outcome

On successful completion of the course the students will be able to
1. Analyze several key technologies used in manipulating, storing and analyzing big data.
2. Experiment with R & Hadoop.
3. Acquire clear understanding of Integrating R & Hadoop
4. Acquire clear understanding of Hadoop Streaming and its importance.
5. Manage Big Data and analyze Big Data.
6. Apply tools and techniques to analyze Big Data.

UNIT 1
INTRODUCTION TO BIG DATA

Big Data and its Importance – Four V's of Big Data – Drivers for Big Data –Introduction to Big Data Analytics..

SLE: Big Data Analytics Applications

8 hrs

UNIT 2
INTRODUCTION TO R & HADOOP

Getting Ready to Use Rand Hadoop, Installing R, Understanding the features of R language, Installing Hadoop, Understanding Hadoop features ,Learning the HDFS and Map Reduce architecture ,Writing Hadoop Map Reduce Programs, Introducing Hadoop Map Reduce ,Understanding the Hadoop Map Reduce fundamentals, Writing a Hadoop Map Reduce example ,Learning the different ways to write Hadoop Map Reduce in R

SLE: Installing R Studio

10 hrs
INTRODUCTION TO DATA MINING (2:0:0)

Sub code : CS0201  
CIE : 50% Marks
Hrs/week : 02  
SEE : 50% Marks
SEE Hrs : 02 Hours  
Max. Marks : 50

Course Outcome

On successful completion of the course the students will be able to
1. Understand Data Mining concepts and applications of Data Mining Applications.
2. Understand Data Warehouse Implementation.
3. Discuss Data Preprocessing Techniques.

UNIT 1

Introduction to Data Mining:

Motivation and importance, What is Data Mining, Relational Databases, Data Warehouses, Transactional Databases, Advanced Database Systems and Advanced Database Applications, Data Mining Functionalities, Interestingness of a pattern Classification of Data Mining Systems.

SLE: Major issues in Data Mining.

9 Hours

UNIT 2

Data Warehouse and OLAP Technology for Data Mining

What is a Data Warehouse? Multi-Dimensional Data Model, Data Warehouse Architecture, Data Warehouse Implementation, Development of Data Cube Technology.

SLE: Data Warehousing to Data Mining

9 Hours

UNIT 3

Data Preprocessing

Why Pre-process the Data? Data Cleaning, Data Integration and Transformation Data Reduction, Discretization.
WEB TECHNOLOGIES AND APPLICATIONS (3:0:2)

Sub code : CS0420  
CIE : 50 %
Hrs / week : 05  
SEE : 50 %
SEE Hrs : 3 Hours  
Max. Marks : 100

Prerequisite: Object Oriented Programming Concepts

Course outcome

On successful completion of the course the students will be able to

1. Explain the importance of java in the internet, its difference compared with other languages and cite examples using data types, variables, operators, control statements, arrays and vectors

2. Compare and demonstrate the use of different category of classes, Recognize group of classes with different visibility control to create package; identification of errors and handling them using exceptional handling;

3. Analyze working of I/O operations, compare functions of string and string buffer class

4. Prepare web pages using HTML/XHTML and CSS.

5. Creation of dynamic web pages using CGI script. Illustrate how to create database using MYSQL and link them with Perl script.

6. Demonstrate how to embed php into HTML and how to link php with MYSQL to create dynamic pages

UNIT 1

The Java Language – 1: The Genesis of Java: Java’s Lineage, The creation of Java, Why Java is important to the Internet? Java’s Magic, Java features; An overview of Java Programming: Object oriented programming, simple java program, Lexical issues; Data types, Variables and Arrays: Simple types, Literals, Variables, Arrays, Operators, Control Statements, Introduction to Java Classes: Classes, Object, Methods, Constructors, The this keyword, Garbage collection, The finalize() Method, A Stack Class.
SLE: Type conversions and Casting
Implementation on above important concepts.

9 Hours

UNIT 2
The Java Language – 2: Methods and Classes; Inheritance; Inheritance Basic, Using Super, Creating Multilevel hierarchy, When Constructors are Called, Method Overriding, Dynamic Method Dispatch, Using Abstract Classes, Using Final, the Object Class; Java Packages and Interfaces; Exception Handling; Multithreading Programming.

SLE: Creation of User defined package
Implementation on above important concepts.

9 Hours

UNIT 3

SLE: StringBuffer Class.
Implementation on above important concepts.

9 Hours

UNIT 4
XHTML, XML and CSS: XHTML: Basic syntax; Standard XHTML document structure; Basic text markup, Images; Hypertext Links; Lists; Tables; Forms; Frames; CSS: Introduction; Levels of style sheets; Style specification formats; Selector forms; Property value forms; Font properties; List properties; Color; Alignment of text; The Box model; Background images; The and tags; Conflict resolution. XML: Introduction; Syntax; Document structure; Document Type definitions; Namespaces; XML schemas; Displaying raw XML documents; Displaying XML documents with CSS.
SLE: Creation of Web Pages
Implementation on above important concepts.

UNIT 5
CGI and MYSQL: Introduction to Apache Web Server-Introduction, Starting, stopping and Restarting Apache, Apache Log Files, MYSQL: Introduction, Tutorial, DBI, Table Joins, CGI – Introduction. A Simple CGI Program, what can go wrong in the program? CGI.pm: Introduction, CGI.pm HTML Shortcuts, Information received by the CGI program, Form Widget Methods, Security consideration, A note on die().

SLE: Loading and Dumping a Database, Apache Configuration.
Implementation on above important concepts.

UNIT 6
PHP and MYSQL: PHP Introduction, Embedding PHP into HTML, Configuration, Quick examples, Language Syntax, Built-IN PHP Functions, PHP and MySQL.

SLE: Connecting php to Mysql.
Implementation on above important concepts.

8 Hours

Note: Implementation of the concepts taught in the theory classes will be done in laboratory sessions.

TEXT BOOKS
3. Open Source Web Development with Lamp, James Lee and Brent Ware, Pearson Education, 2009
REFERENCE BOOKS


CRYPTOGRAPHY (3:0:2)

Sub code : CS0421
Hrs / week : 05
SEE Hrs : 03 Hours

CIE : 50 % Marks
SEE : 50 % Marks
Max. Marks : 100

Course Outcome

On successful completion of the course the students will be able to

1. Explain the foundations of Cryptography and network security. Critical evaluation of the risks and threats to networked computers. Demonstration of the various mechanisms to protect data and their limitations.

2. Demonstrate detailed knowledge of the role of encryption to protect data using DES.

3. Demonstrate advanced encryption technique like AES.

4. Familiar with cryptographic techniques for secure (confidential) communication of two parties over an insecure (public) channel using public key algorithms.

5. Verify the integrity of the messages transmitted via an insecure channel and unique identification of the originator of any message using MAC.

6. Explain the different key management and digital signature systems.

UNIT 1

Introduction to Symmetric-Key Encipherment : Security Goals, Cryptographic Attacks, Services and mechanism, Techniques.

Traditional Symmetric-Key Ciphers: Introduction, Substitution Ciphers, Transposition Ciphers.

SLE : Stream and Block Ciphers.

9 Hours

UNIT 2

Mathematics of Symmetric-Key Cryptography: Algebraic Structures, GF(2^n)Fields. Introduction to Modern Symmetric-Key
Ciphers: Modern Block Ciphers, Modern Stream Ciphers.

**Data Encryption Standard (DES):** Introduction, DES Structure (overview only), Security of DES.

SLE: Multiple DES-Conventional Encryption Algorithms

9 Hours

**UNIT 3**

**Advanced Encryption Standard (AES):** Introduction, Transformations, Key expansion, The AES Ciphers, Examples, Analysis of AES.

**Encipherment Using Modern Symmetric-Key Ciphers:**
Use of Modern Block Ciphers (overview only)

SLE: Use of Stream Ciphers (overview only)

8 Hours

**UNIT 4**

**Asymmetric-Key Cryptography:** RSA Cryptosystems, Elliptic curve crypto systems (overview only)

SLE: ElGamal Cryptosystem.

9 Hours

**UNIT 5**

**Message Integrity and Message Authentication**

Message Integrity, Random Oracle Model, Message Authentication.

**Digital Signature:** Comparison, Process, Services, Attacks of Digital Signature, Digital Signature Schemes.

SLE: Variations and Applications.

9 Hours

**UNIT 6**

**Entity Authentication:** Introduction, Passwords, Challenge-Response, Biometrics.

**Key Management:** Symmetric-Key Distribution, Kerberos.
Symmetric-key Agreement.

SLE : Public-Key Distribution.

TEXT BOOK


REFERENCE BOOKS

MANAGEMENT AND ENTREPRENEURSHIP (4:0:0)

Sub Code: CS0424 CIE: 50%
Hrs/week: 04 SEE: 50%
See Hrs: 3 Hrs Max Marks: 100

COURSE OUTCOME

On successful completion of the course the students will be able to

1. Outline the fundamental concepts of management, types of ownership and organization structures.
2. Demonstrate the insights into fundamentals of economic theory and financial management
3. Elucidate ideation and new product development
4. Understand importance and concepts of quality and frequently used quality tools
5. Elucidate human behavior in organizations
6. Identify critical skills for an entrepreneur and provide insight into feasibility analysis.

UNIT 1

Management – History of scientific management – types of ownership. – types of planning,
SLE-Organization structures.

8 Hours

UNIT 2

SLE-Concepts of Balance Sheet, Profit and loss Account

9 Hours
UNIT 3
New Product Development and Marketing
Product Development Life Cycle, Market Strategy
SLE-Concept of Sales

UNIT 4
Quality Management: Definition, Historical Review, QC Tools: Flowcharts, Histograms, Cause and Effect Diagrams, Check Sheets, Pareto Diagrams,
SLE- Control Charts and Scatter Diagrams.

UNIT 5

UNIT 6

TEXT BOOKS:
3. Total Quality Management, Dale H. Bester field, Publisher - Pearson Education India, Edition 03/e Paperback (Special Indian Edition)

REFERENCE BOOKS:


3. The Frontiers of Management by Peter – F. Drucker, Elsevier publications, Year 2006
PROJECT PHASE -I (0:0:6)

Sub code : CS0301          Hrs / week : 06

Course outcome

On successful completion of the course the students will be able to:

1. Identify a real world engineering problem and formulate it.

2. Outline a software project plan to check the feasibility of the solution in terms of both time and cost and carry out Analysis.

3. Carry out extensive literature survey to evaluate the available tools and adapt it to develop a suitable design.
Sub code : CS0110  
Course outcome

On successful completion of the course the students will be able to

1. Identify current trends in a specific area of interest.

2. Identify real world issues by conducting literature survey of the area.

3. Understand and interpret the results of technical work as indicated by the literature.

4. Demonstrate both technical report writing and presentation skills.

5. Communicate effectively, so as to improve both oral and written skills.
PROJECT PHASE-II (0:0:12)

Sub code : CS0601

Course outcomes

On successful completion of the course the students will be able to:

1. Construct the proposed design of Phase-I using appropriate tools and technology.
2. Implement the constructed design to get working results.
3. Verify and validate the obtained results.
4. Prepare a detailed technical report of the project work carried out.
5. Also suggest limitations and further extensions for the work.
6. Publish the work in a reputed journal/conference.
COMPUTER CONCEPTS AND ‘C’ PROGRAMMING (4:0:0)

Sub code : CS0401
Hrs/week : 04
SEE Hrs : 03 Hours

CIE : 50% Marks
SEE : 50% Marks
Max. Marks: 100

Course Outcomes:
On successful completion of the course, the students will be able to

1. Identify the various terminologies and basic interaction methods in the computers
2. Describe the basic methods of storing and processing data and also the role of operating systems and networking.
3. Analyse problems and convert them to flowcharts & algorithms.
4. Illustrate the concepts of operators, decision making and branching.
5. Demonstrate the use of arrays & structures.
6. Implement user defined functions, pointers & File management.

UNIT – 1

Introducing Computer Systems:
The Computer defined, Computers for individual users, Computers for organizations, The parts of a computer system, The information processing cycle, Essential computer hardware.

Interacting with Computer:

SLE: Identification of latest printers and its features. 9 Hrs

UNIT – 2

Processing Data:
Transforming Data into Information: How computers represent data?, How computers process data?, Factors affecting processing speed.

Storing Data: Magnetic Storage Devices- How data is stored?, How data is organized on a disk?, Solid-State storage devices- Flash memory and smart card.
Using Operating Systems:

Networks and the Internet:
Networking basics - The uses of a network, Common types of networks, Introduction to Internet, Internet’s major services, WWW, E-mail.

SLE: Importance of Android OS, Examples of Magnetics and optical storage devices.

UNIT – 3
Algorithms and Flowcharts:
Algorithms, Flowcharts, Writing algorithms and drawing flowcharts for simple exercises.

Introduction to C
Constants, Variables, and Data types:
Structure of C, Characters set, C tokens, Keywords and Identifiers, Constants, Variables, Data types, Declaration of variables.
Operators and Expressions:
Arithmetic operators, Relational operators, Logical operators, Assignment operators, Increment and decrement operators, conditional operator, Bitwise operators, Special operators, Arithmetic expressions, Evaluation of expressions, Precedence of Arithmetic operators, Type conversions in expressions, Operator precedence and associativity.

SLE: Identification of latest browsers used in different devices (computer and mobile).

UNIT – 4
Managing Input and Output Operations:
Reading a character, Writing a character, Formatted Input, Formatted Output.

Decision making and Branching:
Decision making with if statement, Simple if statement, The if...else statement, Nesting of if...else
Decision making and Looping:
The while statement, the do statement, the For statement, Jumps in Loops.

SLE: Writing Set of Programs on above concept.

UNIT – 5

Arrays:
One-dimensional Arrays, Declaration of one-dimensional Arrays, Initialization of one-dimensional Arrays, Two-dimensional Arrays, Initializing two-dimensional Arrays

Strings:
Introduction to Strings, Declaring and initializing, Reading a String, Writing a String, String Functions (strcat, strcpy, strlen, strcmp, strlcmp)

Structures:
Defining a Structure, Declaring Structure Variables, Accessing Structure Variables, Structure Initialization.

SLC: List applications of arrays and structures. Writing Programs on above concept.

UNIT – 6

User-defined Functions:
Need for User-defined Functions, A multi-function Program, Elements of User-defined Functions, Definition of Functions, Return Values and their Types, Function Calls, Function Declaration, Parameter Passing technique-call by value and call by reference, Category of Functions.

Pointers:
Understanding Pointers, Accessing the Address of a Variable, Declaring Pointer Variables, Initialization of Pointer Variables, Accessing a variable through a Pointer.

File Management in C:
Introduction, Defining and Opening a File, Closing File & Input/output operations on Files.

SLE: Programs on I/O operations on files, applications of pointers.

TEXT BOOKS:
REFERENCE BOOKS:
1. Introduction to Computer Science, ITL Education Solutions Ltd.
4. The C programming language, Brian w. Kernighan, Dennis Ritchie 2nd edition

Electronics Fundamentals (4:0:0)

Sub. Code: EC0401
Hrs/Week: 4
SEE Hrs: 3 Hrs

CIE: 50% Marks
SEE: 50% Marks
Max. Marks: 100

Course Outcome:
At the end of this course the student will be able to,

1. Apply the concept of science and mathematics to explain the working of diodes and its applications, working of transistor and its characteristics and to analyze and design different transistor biasing circuits
2. Apply the concept of stabilization technique, biasing, coupling and feedback of transistors to explain the working of amplifiers and oscillators.
3. Solve problems of various digital logic gates and circuits.
4. Describe the concept of various communication systems and its applications.
5. Correlate the fundamental concepts to various Real life applications of today.

Unit 1: Semiconductor Diodes and Applications
p-n junction diode, Characteristics and Parameters, Diode approximations, Zener diodes, Half-wave diode rectifier, Ripple factor, Full-wave diode rectifier, Other full-wave circuits (Qualitative analysis only) Shunt capacitor - Approximate analysis of capacitor filters, Zener diode voltage regulators, Numerical examples as applicable
SLE: Simple Regulated power supply

8 Hrs

Unit 2: Transistors
Bipolar Junction transistor, Transistor Voltages and currents, amplification, Common Base, Common Emitter Characteristics, DC Load line and Q Point, Biasing methods, Voltage divider Bias, Operating point, Note on stability, Numerical examples as applicable.
SLE: Common Collector Characteristics, Bias compensation for VBE and Ico.

8 Hrs

Unit 3: Amplifiers
Introduction to Amplifiers, Transistor as an Amplifier – Graphical Analysis, Cascading of amplifiers, types of coupling, RC coupled amplifier and its frequency response, Numerical examples as applicable.

SLE: Overview of design of RC coupled Amplifier.

8 Hrs
**PART-B**

1. Estimation of Mohr's salt by Potentiometric titration.
3. Determination of pKa of a weak acid using pH meter.
4. Estimation of copper by using Colorimeter.
5. Determination of viscosity coefficient of a given liquid using Ostwald's viscometer.

(Note: Any ten experiments may be conducted)

**Reference Books**

Vogel's textbook of quantitative inorganic analysis, revised by J. Bassett, R.C. Denny, G.H. Jeffery, 4th Ed.

**COMPUTER PROGRAMMING LABORATORY (0:0:3)**

Sub code : CS0101
Hrs/week : 03
Max. Marks: 50

On Successful completion of this course, the students will be able to

1. Demonstrate the hardware components, DOS commands, mechanics of creating a Word document, the use of basic functions and formulas in Microsoft Excel and the basic mechanics of creating a PowerPoint presentation.
2. Apply and practice logical ability to solve the problems.
3. Understand C programming development environment, compiling, debugging, linking and executing a program using the development environment.
4. Analyzing the complexity of problems, modularize the problems into small modules and then convert them into programs.
5. Understand and apply the in-built functions and customized functions for solving the problems.
6. Understand and apply the pointers, memory allocation techniques and use of files for dealing with variety of problems.
PART - A

1. a. Identify the hardware components of a computer
   b. Execute basic DOS commands.

2. Create a document using a suitable word processing package, with at least three paragraphs and perform the following operations:
   I. Set left margin 1” and right margin 0.75”
   II. Centre the heading and make it bold. Increase the font size
   III. Underline the specified words in the document and change them to italics
   IV. Conduct spell check and correct them suitably
   V. Demonstrate use of numbering and bullets
   VI. Exchange paragraphs 2 and 3 using cut and paste facility
   VII. Put suitable headers and footers
   VIII. Count the number of words and lines and demonstrate the use of drawing tools
   IX. Include suitable logo/emblem/symbol
   X. Include a table

3. Generate pay slip for different categories of employees (example: manager, assistant manager, supervisor, clerk etc) using spreadsheet which includes Basicpay, DA, HRA and other allowances along with deductions like PF, Professional Tax, and other deductions. DA and HRA are calculated in terms of percentage which keeps varying. Use formula while performing calculations. Each column and row should display total. Sort the name list in ascending order. Draw the bar chart to describe the DA percentage for different types of employees.

4. Create a PowerPoint presentation consisting of
   Master slide generation
   Minimum 8 slides
   Information about your college
   Include animation, Hyperlinks, Insert Header and Footer, Insert files, images, tables and Charts.
   Slide show features.

5. Write a C program to find and output all the roots of a given quadratic equation, for non-zero coefficients. (Using if...else statement)

6. Write a C program to simulate a simple calculator that performs arithmetic operations like addition, subtraction, multiplication, and division only on integers. Error message should be reported, if any attempt is made to divide by zero. (Using switch statement)

7. Write a C program to reverse a given six digit integer number and check whether it is a palindrome or not. Output the given number with suitable message. (Using While statement)

8. Write a C program to generate and print first ‘N’ Fibonacci numbers. (Using do-While statement)
9. Write a C program to find the GCD and LCM of two integers and output the results along with the given integers. Use Euclid's algorithm. (Using looping constructs)

10. Write a C program to find whether a given number is prime or not. Output the given number with suitable message. (Using for statement)

PART – B

11. Write a C program to input N real numbers in ascending order into a single dimension array. Conduct a binary search for a given key integer number and report success or failure in the form of a suitable message.

12. Write a C program to input N integer numbers into a single dimension array. Sort them in ascending order using bubble sort technique. Print both the given array and the sorted array with suitable headings.

13. Write a C program to read two matrices A(M x N) and B(P x Q) and perform addition. Output the input matrices and the resultant matrix with suitable headings and format. (Using two dimension arrays where array size M, N, P, Q ≤ 3)

14. Write C user defined functions:
   3. To input N integer numbers into a single dimension array.
   4. To conduct a linear search.
      Using these functions, write a C program to accept the N integer numbers & given key integer number and conduct a linear search. Report success or failure in the form of a suitable message.

15. Write C user defined functions:
   - To input N integer numbers into a single dimension array.
   - To sort the integer numbers in descending order using selection sort technique.
   - To print the single dimension array elements.
      Using these functions, write a C program to input N integer numbers into a single dimension array, sort them in descending order, and print both the given array & the sorted array with suitable headings.

16. Write C user defined functions:
    To input N real numbers into a single dimension array.
    Compute their mean.
    Compute their variance
    Compute their standard deviation.
    Using these functions, write a C program to input N real numbers into a single dimension array, and compute their mean, variance & standard deviation. Output the computed results with suitable headings.
17. Write a C-program that reads a string from keyboard and determines whether the string is a palindrome or not using:
   a. String functions
   b. user-defined functions (without C-library functions)

18. Write C user defined functions:
   - To read the elements of a given matrix of size M \times N
   - To print the elements of a given matrix of size M \times N
   - To compute the product of two matrices
   Using these functions, write a C program to read two matrices A (M \times N) and B (P \times Q) and compute the product of A and B after checking compatibility for multiplication. Output the input matrices and the resultant matrix with suitable headings and format. (Using two dimension arrays where array size M, N, P, Q \leq 3)

19. a. Write a C program to create an integer variable, Assign value to the variable, create a pointer to this variable and print the value of the variable and address of the variable using pointer.

   b. Write a C program to create a structure called Employee with members Name, Job, Salary. Create a structure variable. Accept the input values for the structure members at run time. Suitably display the same.

20. Write a C program to create a C file called student. rec and store information about a student, in terms of his NAME, USN, BRANCH, and SEMESTER. Read the stored contents from the file and display on the screen.
ENGLISH ENHANCEMENT COURSE

Sub code: HS0201
Hrs/Week: 2 Hrs.
SEE Hrs: 2 Hrs.

CIE : 50% Marks
SEE : 50% Marks
Max. Marks : 50

Course Prerequisites :
None

Course Outcomes :

Upon successful completion of this course, the student will be able to:

1. Understand, identify, analyse and apply the concepts for effective communication and give a good introduction for effective presentations
2. Hone listening skills and be able to interpret different accents, follow the directions given and work accordingly
3. Learn to send written messages appropriately
4. Seek information from various sources, assimilate, and voice their opinion effectively
5. Design and deliver a good presentation

Course Content :

**Unit I : 4 Hours : STAR**
- S-T-A-R : (Speak – Transcribe – Analyse - Record )

**Unit II : 10 Hours : Communication and Grammar**
- Basics of communication, its barriers, preparing a good introduction, using comprehensible accent and using proper grammar
- Grammar concepts through common errors

**Unit III: 2 hours**
- Listening skills : listen to different accents and directions. Alternately – Recollect from the visual words, speech and visuals. Face to face and back to back communication
- Email etiquette, conveying messages like acceptance or rejection of job offers, Requesting information of a product or service or follow ups to earlier correspondence, agreement and disagreement etc
Unit IV: 10 hours: Group Discussion and Seminar

Group Discussion – Seek information on the given topic, assimilate and present/share opinions, facts, ideas etc within the given parameters. Topics will pertain to current issues for eg. Occupational disease and role of engineers to combat it (shop floor, operations area, silicosis etc), Elections in India – the new circus, Does India enjoy demographic advantage given its youth population, Start ups – the new employment agency

TEXT BOOKS:

1. A Mirror of Common Errors by Ashok Kumar Singh, Publisher – Students’ Friends
2. English Grammar by Wren and Martin

REFERENCES:

1. King’s English – The first encyclopedia of English Language, Publishers – Addone
2. Internet sources

Note:

Handouts, Questionnaires and materials will be provided

ENGINEERING MATHEMATICS – II (4:0:0)
(Common to all branches)

Sub Code: MA0402
Hrs/Week: 04
SEE Hrs: 03
Total Hrs: 52 Hrs
CIE: 50% Marks
SEE: 50% Marks
Max. Marks: 100

Course Outcomes:

On successful completion of the course the students will be able to:
1. Solve linear second and higher order differential equations with constant coefficients and apply these techniques to compute the solution of differential equation arising from LRC circuit, motion of mass spring.
DATA STRUCTURES WITH ‘C’ (4:0:0)

Sub code : CSO404  
Hrs/week : 04  
SEE Hrs : 03 Hours  
CIE : 50% Marks  
SEE : 50% Marks  
Max. Marks: 100

Prerequisite: Programming in ‘C’

Course Outcome

On Successful completion of the course, the students will be able to:

1. Illustrate basic operations to be performed using pointers and dynamic memory allocations.
2. Demonstrate Abstract Data Types using an Arrays and recursion.
3. Illustrate various operations of different types of Queues.
4. Explain the basic operations of linked list and its implement using dynamic variables.
5. Demonstrate the operations and applications of Binary trees.
6. Apply different sorting and searching methods.

UNIT – 1

Pointer: Understanding pointers, Chain of pointers, Pointer Arithmetic and arrays, Pointer increments and scale factor, Array of pointers, Pointer and character string, Pointers as function argument, Function returning pointers, Pointers to function, Pointers and structures. **Dynamic memory allocation**: malloc, calloc, free, realloc.

SLE: Pointer expressions

9 Hours

UNIT – 2

OBJECT ORIENTED PROGRAMMING WITH C++

(4:0:0)

Sub code : CS0405
Hrs / week : 04
SEE Hrs : 3 Hrs
Prerequisite: C++

Course Outcome

On successful completion of the course the students will be able to
1. Explain the various principles of Object Oriented Programming.
2. Formulate the Objects and classes in the real world application.
3. Discuss the concepts of references, dynamic allocation, function overloading and copy constructors.
4. Identify the importance of Operator overloading and inheritance.
5. Use Virtual functions, Templates and exception handling.
6. Practice the basic I/O operations and file I/O operations in C++.

UNIT 1: 4 Hrs


Self Learning Exercise: An Example with Class, Structure of C++ Program.

UNIT 2: 11 Hrs

Tokens, Expressions and Control Structures: Introduction, Tokens, Keywords, Identifiers and Constants, Basic Data Types, User-Defined Data Types, Derived Data Types, Symbolic Constants, Type Compatibility, Declaration of Variables, Dynamic Initialization of variables, Reference variables, Operators in C++, Scope Resolution Operator, Member Dereferencing Operators,
Sub code : CS0102

Course Outcomes

On successful completion of the course the students will be able to implement data structures like Stacks, Queues, Linked List, and Trees using C.

a) Implement the basic search and sort algorithms.

b) Appropriate use of a particular data structure and algorithm to solve a problem
Course Outcome

On successful completion of the course the students will be able to

1. Understand object-oriented concepts and how they are supported by C++ and implementation issues related to object-oriented techniques.

2. Demonstrate the ability to analyze, use, and create functions, classes, to overload operators, and use inheritance and Pointers when creating or using classes and create templates

3. Demonstrate the ability to understand and use Exception handling and file handling mechanism.

4. Design and write programs that make appropriate use of advanced object-oriented facilities common to many object-oriented languages such as classes, message passing, overloading and inheritance.
Course Outcomes

On successful completion of the course the students will be able to
1. Analyze the mechanisms of Operating Systems to handle processes and threads and their communication
2. Identify the issues and various solutions for inter-process communication
3. Understand the different approaches to memory management.
4. Identify the design and implementation issues in virtual memory management
5. Explain the concepts of file system and its implementation
6. Explain the concepts of deadlock and analyze how they can be managed/avoided

UNIT 1

INTRODUCTION, PROCESSES AND THREADS: WHAT IS AN OPERATING SYSTEM?: The Operating system as an extended machine. The Operating system as a resource manager.


THREADS: The thread model, Thread usage, implementing threads in user space, implementing threads in the kernel. Hybrid implementations, Scheduler activations. SCHEDULING: Introduction to scheduling, Scheduling in batch Systems, Scheduling in interactive systems.

SLE: Thread scheduling 9 Hours

UNIT 2

INTERPROCESS COMMUNICATION


Memory Management: No Memory Abstraction, A Memory Abstraction: Address Spaces - The Notion of an Address Space, Swapping, Managing Free Memory.

SLE: Barriers 8 Hours

UNIT 3


PAGE REPLACEMENT ALGORITHMS: The optimal page replacement algorithm. The not recently used page replacement algorithm, The first-in first-out, The second chance page replacement algorithm. The clock page replacement algorithm, The least recently used,

SLE: Comparison of Page Replacement Algorithms 9 Hours
UNIT 4

**DESIGN ISSUES FOR PAGING SYSTEMS**: Local versus Global allocation policies, Load control, Page size, Separate instruction and data spaces, Shared pages, Virtual memory interface. **IMPLEMENTATION ISSUES**: Operating system involvement with paging, Page fault handling, Instruction backup, locking pages in memory, Backing store. **SEGMENTATION**: Implementation of pure segmentation, Segmentation with paging Multics.

**SLE**: Separation of policy and mechanism.

8 Hours

UNIT 5


**SLE**: Defragmenting Disks

9 Hours

UNIT 6

**DEADLOCKS**: INTRODUCTION TO DEADLOCKS: Conditions for deadlock, Deadlock modeling. THE OSTRICH ALGORITHM, DEADLOCK DETECTION AND RECOVERY: Deadlock detection with one resource of each type, Deadlock detection with multiple resource of each type, Recovery from deadlock. **DEADLOCK AVOIDANCE**: Resource trajectories, Safe and Unsafe States, The Banker’s algorithm for a single resource, The banker’s algorithm for multiple resources, DEADLOCK PREVENTION: Attacking the mutual exclusion condition, Attacking the Hold and Wait condition, Attacking the No Preemption condition.

**SLE**: Attacking the Circular Wait condition

9 Hours

TEXT BOOK:


REFERENCE BOOKS:

DATABASE MANAGEMENT SYSTEMS (4:0:0)

Sub code : CS0414
Hrs / week : 04
SEE Hrs : 03 Hrs
CIE : 50 %
SEE : 50 %
Max. Marks : 100

Prerequisite: File Systems

Course Outcome:

On successful completion of the course the students will be able to

1. Describe different database models and creation of MySQL database.
2. Design the database using Entity Relationship model and Normalization.
3. Write the SQL query to interact with the database and illustrate transaction concepts.
4. Create triggers, constraints and objects in MySQL.
5. Demonstrate the use of Indexes and Views.
6. Differentiate different types of failures and recovery techniques.

UNIT 1

Introduction: Advantages of using a DBMS approaches, Definition of schema, data model and instances, Three-schema architecture and data independence, Different Database Models, Interfacing with a Database, The Mechanics, Disk versus Main-Memory,

An Example Database: The Everest Books Database

Relational Databases (with a MySQL Flavor): MySQL Database System, Database Organization, Creating and using the Database.

SLE: Steps in Designing a Database. 9 Hours

UNIT 2

Manipulating the Database: Example Tables, Relational Algebra.

*Database Design 1: The Entity-Relationship Model*: Entity, Entity Sets, Attributes, Relationships, Constraints on Relationship types, Weak entity type, Notations for ER Diagrams, Modeling The Everest Books’ Database

*Database Design 2*: Definition of Functional Dependencies, Normal Forms Based on Primary Keys - 1NF, 2NF, 3NF, BCNF

SLE: Ad Hoc Database Design 9 Hours

UNIT 3

SQL: Basics, Data Definition Language, Data Manipulation Language, Stored Routines, Formatting and SQL.

Transactions: Informally Speaking, Transactions in SQL, Transaction Semantics, Serializability, Locks,
SQL Isolation Levels, Transactions in MySQL
SLE: Orders, Invoices & Reports

UNIT 4

Constraints: Constraints in SQL, Constraint Check Time.
Triggers: Triggers in SQL.
Objects: Objects in SQL, Referencing Objects

SLE: Comparing Objects.

9 Hours

UNIT 5

Indexes: How does an Index Work?, Types of Indexes, Different Types of Indexes, Hashindexes, Indexes in MySQL,
Views: Two Implementations of Views, Views in MySQL, Examples.

SLE: Some Thoughts on Indexing.

9 Hours

UNIT 6

Spatial Databases: MySQL, Example
Logs and Recovery: Failure Types, Logs, Log Details, Some Log Characteristics, Database Recovery, Database Recovery Example, MySQL

SLE: Spatial Databases: Example

8 Hours

TEXT BOOKS:


REFERENCE BOOKS:

MANAGEMENT AND ENTREPRENEURSHIP (4:0:0)

Sub Code: CS0424
Hrs/week: 04
See Hrs: 3 Hrs

Course Outcome

On successful completion of the course the students will be able to

1. Provide the information about management as science, art, administration and as a key approach in doing a assigned work.
2. Explain planning and organizing, with planning students learn its importance and hierarchy and with organizing students learn about departmentation, committees, centralization and decentralization.
3. Explain the importance of staffing directing and controlling, meaning and steps in controlling.
4. Identifying who is an entrepreneur, his functions and types and identifying based on geography.
5. Provide knowledge of small scale industry and institutional support.
6. Prepare project Selection; Project Report; Need and Significance of Report.

UNIT 1
Management: Introduction: Meaning - nature and characteristics of Management, Scope and functional areas of management - Management as a science, art or profession Management & Administration - Roles of Management, Levels of Management.

SLE: Development of Management Thought - early management approaches - Modern management approaches, case studies on administration and management. 7 Hours

UNIT 2
Planning: Nature, importance and purpose of planning process - Objectives - Types of plans (Meaning only) - Decision making - Importance of planning - steps in planning & planning premises - Hierarchy of plans.

Organizing: Nature and purpose of organization - Principles of organization - Types of organization - Departmentation - Committees – CentralizationVsDecentralization of authority and responsibility - Span of control - MBO and MBE (Meaning only).

SLE: Case studies on planning and organizing. 9 Hours

UNIT 3

SLE: Cases studies on staffing, directing & controlling. 10 Hours

UNIT 4
Entrepreneur: Meaning of Entrepreneur; Evolution of the Concept; Functions of an Entrepreneur, Types of Entrepreneur, Entrepreneur - an emerging Class; Concept of Entrepreneurship-Evolution of Entrepreneurship, Development of Entrepreneurship; Stages in entrepreneurial process; Role of entrepreneurs in Economic Development; Entrepreneurship in India; Entrepreneurship – its Barriers.
MICROPROCESSOR INTERFACING LABORATORY (0:0:3)

Sub code: CS0106

Course Outcome

On successful completion of the course the students will be able to

1. Implement assembly language programs for 8086 microprocessor for the given applications.

2. Implement assembly language interfacing applications for Elevator, Stepper Motor, Keypad, logic controller Display interface.

DATABASE LABORATORY (0:0:3)

Sub code: CS0107

Course Outcome:

At the end of this course the student will be able to

1. Explain his/her knowledge of database technology, its importance, its architectures, and the central role Database technology plays in Information Systems.

2. Apply appropriate development methodologies of data analysis, design and use appropriate modeling techniques for databases.

3. Demonstrate query facilities to formulate queries and manipulate the database e.g. Structured Query Language (SQL)

4. Appreciate the issues underlying database implementation in any database

5. Concentrate on a methodology for good database design and practical experience in designing and implementing standalone database system.
COMPUTER NETWORKS (4:0:0)

Sub code: CS0418
Hrs / week: 04
SEE Hrs: 03 Hours

CIE: 50 %
SEE: 50 %
Max. Marks: 100

Prerequisite: Data Communications and Networking

Course Outcomes:

On successful completion of the course the students will be able to

1. Discuss the network layer services and learn the concepts of IP address, Classes, IPv4 Datagram.

2. Learn the auxiliary protocols in IPv4 and working of different Unicast Routing algorithms and protocols.

3. Discuss the concepts and working of IPv6 and various Multicast routing protocols.

4. Identify the services of the Transport layer and analyze the performances of UDP and TCP protocols.

5. Explain the congestion control in TCP and the working of SCTP.

6. Distinguish between various application layer protocols and analyze the need for information on DNS.

UNIT 1


SLE: Security of IPv4 datagrams. 9 hours

UNIT 2


SLE: OSPF 8 hours

UNIT 3

SLE: Transition from IPV4 to IPV6

UNIT 4

SLE: UDP Applications

UNIT 5

SLE: Taho TCP

UNIT 6

SLE: Web documents

TEXT BOOK:


REFERENCE BOOKS:

WEB TECHNOLOGIES AND APPLICATIONS (4:0:0)

Sub code: CS0450
Hrs / week: 04
SEE Hrs: 03 Hours
CIE: 50 %
SEE: 50 %
Max. Marks: 100

Prerequisite: Object Oriented Programming Concepts

Course outcome

On successful completion of the course the students will be able to

1. Interpret the basics of java language
2. Implement advance features of java – inheritance, multiple inheritance, packages, interface, multithreading and exception handling
3. Illustrate the usage of I/O streams, string handling, and networking in java
4. Prepare dynamic web pages using XHTML, CSS and XML
5. Create dynamic web pages using CGI and MYSQL
6. Demonstrate the creation of dynamic web pages using XHTML, PHP, MYSQL

UNIT 1

The Java Language – 1: The Genesis of Java: Java’s Lineage, The creation of Java, Why Java is important to the Internet? Java’s Magic, Java features; An overview of Java Programming: Object oriented programming, simple java program, Lexical issues, Data types, Variables and Arrays: Simple types, Literals, Variables, Arrays, Operators, Control Statements, Introduction to Java Classes: Classes, Object, Methods, Constructors, The this keyword, Garbage collection, The finalize() Method, A Stack Class.

SLE: Type conversions and Casting

9 Hours

UNIT 2

The Java Language – 2: Methods and Classes; Inheritance; Inheritance Basic, Using Super, Creating Multilevel hierarchy, When Constructors are Called, Method Overriding, Dynamic Method Dispatch, Using Abstract Classes, Using Final, the Object Class; Java Packages and Interfaces; Exception Handling, Multithreading Programming.

SLE: Creation of User defined package

9 Hours

UNIT 3


SLE: StringBuffer Class.

9 Hours
UNIT 4

**XHTML, XML and CSS:** XHTML: Basic syntax; Standard XHTML document structure; Basic text markup; Images; Hypertext Links; Lists; Tables; Forms; Frames; CSS: Introduction; Levels of style sheets; Style specification formats; Selector forms; Property value forms; Font properties; List properties; Color; Alignment of text; The Box model; Background images; The and tags; Conflict resolution. XML: Introduction; Syntax; Document structure; Document Type definitions; Namespaces; XML schemas; Displaying raw XML documents; Displaying XML documents with CSS.

**SLE:** Creation of Web Pages 8 Hours

UNIT 5

**CGI and MySQL:** Introduction to Apache Web Server- Introduction, Starting, stopping and Restaring Apache, Apache Log Files, MySQL: Introduction, Tutorial, DBI, Table Joins, CGI - Introduction, A Simple CGI Program, what can go wrong in the program? CGI.pm: Introduction, CGI.pm HTML Shortcuts, Information received by the CGI program, Form Widget Methods, Security consideration, A note on die().

**SLE:** Loading and Dumping a Database, Apache Configuration 9 Hours

UNIT 6

**PHP and MySQL:** PHP Introduction, Embedding PHP into HTML, Configuration, Quick examples, Language Syntax, Built-IN PHP Functions, PHP and MySQL.

**SLE:** Connecting PHP to Mysql 8 Hours

**TEXT BOOKS**

3. *Open Source Web Development with Lamp*, James Lee and Brent Ware, Pearson Education, 2009

**REFERENCE BOOKS**

WEB TECHNOLOGIES AND APPLICATIONS LABORATORY (0:0:3)

Sub code: CS0112

Course Outcome:

At the end of this course the student will be able to

1. Implement programs on Arrays, Stack, Class and Methods using java
2. Demonstrate usage of Interface, Packages, Multithreading and Exception handling in java
3. Explore working of I/O operations, String class in java
4. Prepare web pages using HTML/XHTML and CSS
5. Create dynamic web pages using XHTML, CGI script and MYSQL
6. Construct dynamic web pages using XHTML/HTML, PHP and MYSQL

COMPUTER NETWORKING LABORATORY (0:0:3)

Sub code: CS0109

Course Outcomes:

On Successful completion of the course the students will be able to:

1. Understand and illustrate cabling techniques and networking devices.
2. Configure and demonstrate working of Switch, Router and Wireless Devices.
3. Understand and Illustrate ACLs and NAT.
4. Configure and demonstrate DHCP Server, Static Routing and Dynamic Routing
INTRODUCTION TO C# PROGRAMMING AND .NET CONCEPTS (3:0:0)

Sub code : CS0312
Hrs/week : 03
SEE Hrs :03 Hours

CIE : 50% Marks
SEE : 50% Marks
Max. Marks :100

Course Outcome

On successful completion of the course the students will be able to

1. Explain the basics of .Net platform and the role of base class libraries, role of common intermediate language and namespaces.
2. Discuss the fundamentals of c# and to build the basic c# program using different constructs.
3. Review the basic pillars of object oriented programming concepts.
4. Apply the exception handling technique to handle different types of errors.
5. Provide the knowledge about basics of object lifetime and to define the use of interfaces and collections.
6. Explain the callback interfaces, delegates, events and its implementation.

UNIT 1
SLE: Increasing Your Namespace Nomenclature

7 Hours

UNIT 2
SLE: Programming using basic constructs of c#  

7 Hours

UNIT 3
OBJECT- ORIENTED PROGRAMMING WITH C#: Forms Defining of the C# Class, Definition the "Default Public Interface" of a Type, Recapping the Pillars of OOP, The First Pillar: C#s Encapsulation Services, Pseudo- Encapsulation: Creating Read-Only Fields, The Second Pillar: C#s Inheritance Supports, keeping Family Secrets: The "Protected" Keyword, Nested Type Definitions, The Third Pillar: C #s Polymorphic Support, Casting Between.
SLE: Object oriented programming using VS NET  

6 Hours
Course Outcome
On successful completion of the course the students will be able to

1. Describe the fundamentals of Embedded Systems.
2. Develop necessary skills to understand and design an embedded system application.
3. Identify the challenges of Concurrent Process and its solutions.
4. Compare the advantages of the software Architectures and design an embedded system application using an RTOS.
5. Describe the inter task communication primitives like, semaphores and message queues, and design the hardware along with the choice of the RTOS for the application at hand.
6. Describe the Embedded Software Development Tools and Model the requirements of an application as a set of tasks.

UNIT 1
Custom single-purpose processor design; RT-level custom single-purpose processor design; Optimizing custom single-purpose processors; optimizing the FSMD, Optimizing the data-path, optimizing the FSM.
SLE : Optimizing the Original Program. 7 Hours

UNIT 2
Timers, counters, and watchdog timers.
State machine models: Introduction; finite-state machines (FSM); Finite-state machines with data path model (FSMD); Using state machines: Describing a system as a state machine, Comparing state machine and sequential program models, Capturing a state machine model in a sequential programming language; Hierarchical/Concurrent state machine model (HCFSM) and the State charts language; Program state machine model (PSM);
SLE : An introductory example, A basic state machine model. 7 Hours
UNIT 3
Concurrent process models: Concurrent processes: Process create and terminate, Process suspend and resume, Process join; Communication among processes: Shared memory, Message passing; Synchronization among processes: Condition variables, Monitors.
Interrupts: The Shared Data Problem.
SLE: Interrupt Basics. 7 Hours

UNIT 4
Survey of Software Architecture: Round Robin with Interrupts, Function Queue Scheduling Architecture; Real Time Operating System Architecture, Selecting architecture.
Introduction to RTOS: Tasks and Task States, Tasks and Data, Semaphores and shared data.
SLE: Round Robin. 7 Hours

UNIT 5
SLE: Saving Power. 6 Hours

UNIT 6
Embedded Software Development Tools: Host and Target Machines, Linker/Locator for Embedded Software, getting embedded software into the Target System
SLE: Environment in which the program operates. 5 Hours

TEXT BOOKS

REFERENCE BOOKS
2. Real-Time Systems and Programming Languages: Alan Burns and Andy Wellings
INTRODUCTION TO ANDROID PROGRAMMING (3:0:0)

Sub code : CS0313
Hours / week : 03
SEE Hours : 3 Hours

CIE : 50 %
SEE : 50 %
Max. Marks : 100

Pre-requisite: DBMS, Java Programming

Course outcomes:
On successful completion of the course the students will be able to
1. Understand the basic versions, the evolution of Android as a Mobile OS, create simple apps and apply different styles.
2. Experiment with the User Interface of Android using Activities, Fragments and Intents.
3. Manipulate the UI with different common elements.
4. Apply the knowledge to a real time Android Application in terms of Design and Visualization.
5. Apply the knowledge to a real time Internet Download Application.
6. Able to learn how to persist data using preferences.

UNIT 1
SLE: Designing a Android Application

6 Hours

UNIT 2
ACTIVITIES, FRAGMENTS AND INTENTS: Understanding Activities, Applying Styles and Themes to an Activity, Hiding the Activity Title, Displaying a Dialog Window, Displaying a Progress Dialog, Displaying a More Sophisticated Progress Dialog, Linking Activities Using Intents, Resolving Intent Filter, Collision, Returning Results from an Intent, Passing Data Using an Intent Object, Fragments, Adding Fragments Dynamically, Life Cycle of a Fragment, Interactions between Fragments, Calling Built-In Applications Using Intents, Understanding the Intent Object, Using Intent Filters, Adding Categories.
SLE: Displaying Notifications in Android Application

7 Hours

UNIT 3
GETTING TO KNOW THE ANDROID USER INTERFACE: Understanding the Components of a Screen, Views and View Groups, Linear Layout, Absolute Layout, Table Layout, Relative Layout, Frame Layout, Scroll View, Adapting to Display Orientation, Anchoring Views, Resizing and Repositioning, Managing
Changes to Screen Orientation, Persisting State Information during Changes in Configuration, Detecting Orientation Changes, Controlling the Orientation of the Activity, Utilizing the Action Bar, Adding Action Items to the Action Bar, Customizing the Action Items and Application Icon, Creating the User Interface Programmatically, Listening for UI Notifications, Overriding Methods Defined in an Activity, Registering Events for Views.

**SLE:** Designing User Interface

**UNIT 4**

**DESIGNING YOUR USER INTERFACE WITH VIEWS:** Using Basic Views, TextView, View, Button, ImageButton, EditText, CheckBox, ToggleButton, RadioButton, and RadioGroup Views, ProgressDialog View, AutoCompleteTextView, View Using Picker Views, TimePicker View, DatePicker View, Using List Views to Display Long List, ListView View, Using the Spinner View, Understanding Specialized Fragments, Using a ListFragment, Using a DialogFragment, Using a PreferenceFragment

**DISPLAYING PICTURES AND MENUS WITH VIEWS:** Using Image Views to Display Pictures, Gallery and ImageView Views, Image Switcher, Grid View, Using Menus with Views, Creating the Helper Methods, Options Menu, Context Menu, Some Additional Views, Analog Clock and Digital Clock Views, WebView.

**SLE:** Designing Menus in Android

**UNIT 5**

**USING INTERNET RESOURCES:** Downloading and Parsing Internet Resources, Connecting to an Internet Resource, Parsing XML Using the XML Pull Parser, Creating an Earthquake Viewer, Using the Download Manager, Downloading Files, Customizing Download Manager Notifications, Specifying a Download Location, Cancelling and Removing Downloads, Querying the Download Manager, Using Internet Services, Connecting to Google App Engine, Best Practices for Downloading Data Without Draining the Battery

**SLE:** Using Internet Resources to develop Android Applications

**UNIT 6**

**FILES, SAVING STATE, AND PREFERENCES:**

BIG DATA ANALYTICS (2:0:2)

Sub code : CS0324
Hours / week : 03
SEE Hours : 3 Hours

CIE : 50 %
SEE : 50 %
Max. Marks : 100

Course Outcome
On successful completion of the course the students will be able to
1. Overview of Big Data and Related Technologies
2. Analyze Technologies for Handling Big Data and Hadoop Ecosystem
3. Acquire clear understanding of MapReduce Fundamentals and HBase
4. Acquire clear understanding of Virtualizing and Processing Data using MapReduce
5. Acquire a clear understanding of YARN and Mahout
6. Acquire a clear understanding of Hive

UNIT 1
Getting an Overview of Big Data
SLE: Future of Big Data in Automation Industry

UNIT 2
Introducing Technologies for Handling Big Data and Hadoop Ecosystem
SLE: Sqoop, Flume

UNIT 3
Understanding MapReduce Fundamentals and HBase
The MapReduce Framework, Exploring the Features of MapReduce, Working of MapReduce, Exploring Map and Reduce Functions, Techniques to Optimize MapReduce Jobs, Hardware/Network Topology, Synchronization, File System, Uses of MapReduce, Role of HBase in Big Data Processing, Characteristics of HBase
SLE: Installation of HBase

UNIT 4
Understanding Big Data Technology Foundations and Processing your Data with MapReduce
Exploring the Big Data Stack, Virtualization and Big Data, Virtualization Approaches, Developing a Simple MapReduce Application, Points to Consider while designing MapReduce.
SLE: Managing Virtualization with Hypervisor

6 hours
WIRELESS AND MOBILE NETWORKS (3:0:0)

Sub code: CS0330
Hrs/week: 03
SEE Hrs: 3 Hours

CIE: 50%
SEE: 50%
Max Marks: 100

Course Outcome:

On successful completion of the course the students will be able to

1. Describe an advanced element of learning in the field of wireless communication.
2. Outline the basic concepts of wireless connectivity and mobile computing.
4. Have a working understanding of the characteristics and limitations of mobile hardware devices including their user-interface modalities.
5. Gain knowledge about different mobile platforms and application development
6. Understand and analyse how Internet Protocols are used for mobile devices and how they are applied differently compared to static devices.

UNIT 1

SLE: Comparison of Common Wireless Communications Systems, VoLTE, usage in India and its difference with normal voice calls. 5 Hours

UNIT 2
Mobility and Handoff management: Handoff, Mobility management, Roaming management, Strategies for handoff detection, MAHO NCHO, MCHO, channel assignment, radio link transfer, hard handoff and soft handoff

SLE: Ongoing call termination vs New call blocking – precedence during handoff and implementation issues involved. 7 Hours

UNIT 3
Number portability, SRF and challenges: Mobile Computing fundamental challenges, Mobile Number Portability-types, architecture and implementation, fixed networks and mobile networks, Signaling relay approaches, SRF

SLE: Mobile number portability in India, VoIP service for mobile networks 7 Hours
UNIT 4:
GSM and GPRS: GSM architecture, Location tracking and call setup, security, GPRS Services and architecture, classes of GPRS devices, GPRS state model and radio resource and multiple access management, sharing of radio channels between voice and data, USSD
SLE: Phone theft security and its relation with GSM network

7 Hours

UNIT 5:
**Mobile OS**: Mobile Devices – Special Constrains & Requirements – Commercial Mobile Operating Systems, Palm OS, Win CE(basics) and Android OS architecture, Dalvik Virtual Machine working and its differences with JVM, SDKs
SLE: Wireless local loop (WLL)

7 Hours

UNIT 6:
Mobile IP and Short message Services: Mobile IP and IP v 6 and its application in mobile computing, Triangle routing, logical problems and optimization, SMS Architecture, Mobile Originating and Mobile Terminating SMS, SMS TDPUs, SMS encoding & decoding
SLE: WAP protocol stack

6 hours

TEXT BOOKS:

REFERENCE BOOKS:
Python Programming (3:0:0)

Sub Code: CS0315
Hours/Week: 03
SEE Hours: 03 Hours

Pre-requisite: Computer Programming Concepts

Course Outcomes:
On successful completion of the course, the students will be able to:

1. Explain core ideology of Python and write simple python programs using numeric types.
2. Use String, List and Dictionary types in Python for data storage and processing.
3. Use various python control statements to develop applications.
4. Demonstrate use of advanced function features in Python.
5. Apply OOP features in python.
6. Demonstrate advanced function features and exception handling.

UNIT 1:

SLE: Shared References.

UNIT 2:
Lists and Dictionaries: Lists, Lists in action, Dictionaries, Dictionaries in action.
‘Tuples-Files and Everything Else’: Tuples, Files.

SLE: Sets

UNIT 3:
Statements and Syntax: Assignments, Expressions, and Prints: Assignment Statements-Assignment Statement Forms, Sequence Assignments, if Tests and Syntax Rules: if statements, while and for Loops: while loops, break, continue, pass, and the Loop else. For Loops.

SLE: Loop Coding Techniques

UNIT 4:

SLE: Keyword and Default Examples.

UNIT 5: Classes and OOP:
Class Coding Basics, Classes Generate Multiple Instance Objects, Classes Are Customized by Inheritance, Classes Can Intercept Python Operators, The World’s Simplest Python Class, Class Coding Details: The class Statement, Methods, Inheritance, Namespaces: The Conclusion.

SLE: User-Defined Exceptions, Termination Actions.

CIE: 50% Marks
SEE: 50% Marks
Max. Marks: 100
Sub Code: CS0316
Hours/Week: 03
SEE Hours: 03 Hours

Course Outcome
On successful completion of the course the students will be able to
1. Demonstrate the usage of Applet class and its interfaces.
2. Illustrate the main event classes and interfaces used by the AWT along with Creation, management of windows, fonts, output text, and utilize graphics.
3. Discuss standard AWT controls and layout managers
4. Explain the flexibility of GUI components of Swing compare to AWT
5. Discuss building complex systems from software components using Beans and how servlets dynamically extend the functionality of a web server
6. Demonstrate JDBC connectivity to Databases

Unit-1
SLE: The AudioClip Interface, TheAppletStub Interface

Unit-2
Event handling in Java: Two Event Handling Mechanisms, The Delegation Event Model, Events classes, source of events, events listener interfaces, Using the Delegation Event Model, Adapter Classes.
SLE: Inner Classes, Managing text output using FontMetric

Unit-3
Using AWT Controls, Layout Manager and Menus: Control Fundamentals, Labels, Using Buttons, Checkboxes, Checkbox Group, Choice Controls Using Lists, Managing Scroll Bars, Using Text field, Text area, Understanding Layout Managers, Menu bars And Menus, Dialog Boxes.
SLE: File Dialog

Unit-4
CLOUD COMPUTING (3:0:2)

Sub code : CS0437
Hrs / week : 05
SEE Hrs : 3 Hours

CIE : 50 %
SEE : 50 %
Max. Marks : 100

Course outcome

On successful completion of the course the students will be able to

1. Overview of Cloud Computing and various distributed system models with enabling technologies.
2. Analyze various Computer Clusters for Scalable Parallel Computing
3. Acquire the clear understanding of Virtual Machines and Virtualization of Clusters
4. Acquire the basic knowledge of Cloud Platform Architecture over Virtualized Data Centers
5. Acquire the clear understanding of Service-Oriented Architectures for Distributed Computing
6. Develop various applications using Cloud Programming and Software Environments

UNIT 1

Introduction


Distributed System Models and Enabling Technologies


SLE: Role of Open Standards in Cloud Computing

8 hrs
UNIT 2

Computer Clusters for Scalable Parallel Computing


SLE: Tianhe-1A

8 hrs

UNIT 3

Virtual Machines and Virtualization of Clusters and Data Centers

Implementation Levels of Virtualization, Virtualization Structures/Tools and Mechanisms, Virtualization of CPU, Memory, and I/O Devices, Virtual Clusters and Resource Management, Virtualization for Data-Center Automation

SLE: Cloud OS for Virtualized Data Centers

10 hrs

UNIT 4

Cloud Platform Architecture over Virtualized Data Centers


SLE: Google App Engine

9 hrs

UNIT 5

Service-Oriented Architectures for Distributed Computing

Services and Service-Oriented Architecture, Message-Oriented Middleware, Portals and Science Gateways, Discovery, Registries, Metadata, and Databases, Workflow in Service-Oriented Architectures
SLE: Semantic Web and Grid

UNIT 6
Cloud Programming and Software Environments
Features of Cloud and Grid Platforms, Parallel and Distributed Programming Paradigms, Programming Support of Google App Engine, Programming on Amazon AWS and Microsoft Azure, Emerging Cloud Software Environments

SLE: Installation of Openstack

TEXT BOOKS
1. Cloud Computing Bible by Barrie Sosinsky, Wiley India (Chapter 1)
2. Distributed And Cloud Computing, Hwang, Kai; Fox, Geoffrey C; Dongarra, Jack J. ELSEVIER INDIA PVT. LTD (Chapter 1, 2, 3, 4, 5, 6)

REFERENCE BOOKS
4. Google Apps by Scott Granneman, Pearson
CRYPTOGRAPHY (3:2:0)

Sub code : CS0421     CIE : 50 % Marks
Hrs / week : 05      SEE : 50 % Marks
SEE Hrs : 03 Hours    Max. Marks : 100

Course Outcome

On successful completion of the course the students will be able to

1. Explain the foundations of Cryptography and network security. Critical evaluation of the risks and threats to networked computers. Demonstration of the various mechanisms to protect data and their limitations.

2. Demonstrate detailed knowledge of the role of encryption to protect data using DES.

3. Demonstrate advanced encryption technique like AES.

4. Familiar with cryptographic techniques for secure (confidential) communication of two parties over an insecure (public) channel using public key algorithms.

5. Verify the integrity of the messages transmitted via an insecure channel and unique identification of the originator of any message using MAC.

6. Explain the different key management and digital signature systems.

UNIT 1

Introduction to Symmetric-Key Encipherment : Security Goals, Cryptographic Attacks, Services and mechanism, Techniques.

Traditional Symmetric-Key Ciphers: Introduction, Substitution Ciphers, Transposition Ciphers.

Mathematics of Cryptography: Integer arithmetic, Modular arithmetic, Matrices, Linear Congruence

SLE : Stream and Block Ciphers.

9 Hours
UNIT 2

Data Encryption Standard (DES): Introduction, DES Structure (overview only), Security of DES.

SLE: Multiple DES-Conventional Encryption Algorithms

9 Hours

UNIT 3
Advanced Encryption Standard (AES): Introduction, Transformations, Key expansion, The AES Ciphers, Examples, Analysis of AES.

Encipherment Using Modern Symmetric-Key Ciphers:
Use of Modern Block Ciphers (overview only)

SLE: Use of Stream Ciphers (overview only)

8 Hours

UNIT 4
Asymmetric-Key Cryptography: RSA Cryptosystems, Elliptic curve crypto systems (overview only)

SLE: ElGamal Cryptosystem

9 Hours

UNIT 5
Message Integrity and Message Authentication
Message Integrity, Random Oracle Model, Message Authentication.

Digital Signature: Comparison, Process, Services, Attacks of Digital Signature, Digital Signature Schemes.

SLE: Variations and Applications

9 Hours
UNIT 6

**Entity Authentication:** Introduction, Passwords, Challenge-Response, Biometrics.

**Key Management:** Symmetric-Key Distribution, Kerberos, Symmetric-key Agreement.

**SLE:** Public-Key Distribution.

8 Hours

**TEXT BOOK**


**REFERENCE BOOKS**


MANAGEMENT AND ENTREPRENEURSHIP (4:0:0)

Sub Code: CS0424  
CIE: 50%  
Hrs/week: 04  
SEE: 50%  
See Hrs: 3 Hrs  
Max Marks: 100

COURSE OUTCOME

On successful completion of the course the students will be able to

1. Provide the information about management as science, art, administration and as a key approach in doing a assigned work.

2. Explain planning and organizing, with planning students learn its importance and hierarchy and with organizing students learn about departmentation, committees, centralization and decentralization.

3. Explain the importance of staffing directing and controlling, meaning and steps in controlling.

4. Identifying who is an entrepreneur, his functions and types and identifying based on geography.

5. Provide knowledge of small scale industry and institutional support Prepare project Project Selection; Project Report; Need and Significance of Report

UNIT 1

Management: Introduction - Meaning - nature and characteristics of Management, Scope and functional areas of management - Management as a science, art or profession Management & Administration - Roles of Management, Levels of Management,

SLE : Development of Management Thought - early management approaches - Modern management approaches, case studies on administration and management.

UNIT 2

Planning: Nature, importance and purpose of planning process - Objectives - Types of plans (Meaning only) - Decision making -
Importance of planning - steps in planning & planning premises - Hierarchy of plans.

Organizing: Nature and purpose of organization - Principles of organization - Types of organization - Departmentation - Committees - Centralization Vs Decentralization of authority and responsibility - Span of control - MBO and MBE (Meaning only)

SLE: Case studies on planning and organizing 9 hours

UNIT 3

SLE: Case studies on staffing, directing & controlling 10 hours

UNIT 4
Entrepreneur: Meaning of Entrepreneur; Evolution of the Concept, Functions of an Entrepreneur, Types of Entrepreneur, Intrapreneur - an emerging Class. Concept of Entrepreneurship - Evolution of Entrepreneurship, Development of Entrepreneurship; Stages in entrepreneurial process; Role of entrepreneurs in Economic Development; Entrepreneurship in India; Entrepreneurship - its Barriers.

SLE: Case studies on entrepreneurs and overcoming barriers 8 Hours

UNIT 5
Small Scale Industry: Definition; Characteristics; Need and rationale: Objectives; Scope; role of SSI in Economic Development. Advantages of SSI Steps to start an SSI - Government policy towards SSI; Different Policies of S.S.I.; Government Support for S.S.I. during 5 year plans, Impact of Liberalization, Privatization, Globalization on S.S.I., Effect of WTO/GATT Supporting Agencies of Government for S.S.I., Meaning; Nature of Support; Objectives; Functions; Types of Help; Ancillary Industry and Tiny Industry (Definition only).
Institutional Support: Different Schemes; TECKSOK; KIADB; KSSIDC; KSIMC; DIC Single Window Agency: SISI; NSIC; SIDBI; KSFC.

SLE: case studies on setting up of small scale industries and institutional support 10 Hours

UNIT 6

Preparation Of Project: Meaning of Project; Project Identification; Project Selection; Project Report; Need and Significance of Report; Contents; formulation; Guidelines by Planning Commission for Project report; Network Analysis; Errors of Project Report; Project Appraisal. Identification of Business Opportunities: Market Feasibility Study; Technical Feasibility Study; Financial Feasibility Study & Social Feasibility Study

SLE: Case study on preparation of the project report and feasibility study 8 Hours

TEXT BOOKS:

1. *Entrepreneurship development text and cases*, B Janakiram, M Rizwana.


REFERENCE BOOKS:


5. Web Sites for the Institutions listed in the Unit 5 on Institutional Support
PROJECT PHASE -I (0:0:4) (2 Credits)

Sub code : CS0205
Hrs / week : 04

Course outcome:

On successful completion of the course the students will be able to:

1. Identify a real world engineering problem and formulate it.
2. Outline a software project plan to check the feasibility of the solution in terms of both time and cost and carry out Analysis.
3. Evaluate the available tools by Literature survey and adapt it to develop a suitable design.
SEMINAR (0:0:2) (1 Credits)

Sub code : CS0110

Course outcome

On successful completion of the course the students will be able to

1. Identify current trends in a specific area of interest.
2. Identify real world issues by conducting literature survey of the area.
3. Understand and interpret the results of technical work as indicated by the literature.
4. Demonstrate both technical report writing and presentation skills.
5. Summarize effectively, so as to improve both oral and written skills.
PROJECT PHASE-II (0:0:12) (6 Credits)

Sub code : CS0601

Course outcomes

On successful completion of the course the students will be able to:

1. Implement the proposed design of phase-I.
2. Compute the results obtained from the implementation.
3. Verify and validate the obtained results using various test cases.
4. Demonstrate and present the project.
5. Prepare a detailed technical report of the project work carried out.
6. Publish the work in a reputed journal/conference.
PAPER PRESENTATION IN JOURNALS AND CONFERENCE (0:0:2) (1 Credit)

Sub code : CS0113

Hrs/week : 2

Course outcome

On successful completion of the course the students will be able to:

1. Demonstrate the working of new model/method/algorithm in conference.

2. Produce the technical work and results through journals.
Course Outcome:
On successful completion of the course, the students will be able to
1. Identify the various terminologies and therole of operating systems in the computers
2. Analyze problems and solve them using flowcharts & algorithms.
3. Illustrate the concepts of operators, decision making and branching.
4. Demonstrate the use of arrays, structures and strings.
5. Implement user defined functions.
6. Understand the Basics of Python Programming.

UNIT 1

Introducing Computer Systems:
The Computer, defined, Computers for individual users, Computers for organizations, The part of a computer system, The Information processing cycle, Essential computer hardware.

Processing Data:
Transforming Data into Information: How computers represent data? How computers process data? Factors affecting processing speed.

Using Operating Systems:
SLE: Importance of Android OS, Examples of Magnetics and optical storage devices

UNIT 2

Algorithms and Flowcharts:
Algorithms, Flowcharts, Writing algorithms and drawing flowcharts for simple exercises.

Introduction to C
Constants, Variables, and Data types:
Structure of C, Characters set, Ctokens, Keywords and Identifiers, Constants, Variables, Datatypes, Declaration of Variables.

Operators and Expressions:
Arithmetic operators, Relational operators, Logical operators, Assignment operators, Increment and decrement operators, conditional operator, Bitwise operators, Special operators, Arithmetic
expressions, Evaluation of expressions, Precedence of Arithmetic operators, Type conversions in expressions, Operator precedence and associativity.

SLE: Identification of latest browsers used in different devices (computer and mobile). 9hrs

UNIT – 3

Managing Input and Output Operations:
Reading a character, Writing a character, Formatted Input, Formatted Output.

Decisionmaking and Branching:
Decisionmaking with if statement, Simple if statement, The if...else statement, Nesting of if else statements, The else ... if ladder, The switch statement, the ?: operator, The Goto statement.

Decisionmaking and Looping:
The while statement, the do...statement, the For statement, Jumps in Loops.
SLE: Writing Set of Programs on above concept. 9hrs

UNIT – 4

Arrays:
One-dimensional Arrays, Declaration of one-dimensional Arrays, Initialization of one-dimensional Arrays, Two-dimensional Arrays, Initializing two-dimensional Arrays

Structures:
Defining a Structure, Declaring Structure Variables, Accessing Structure Variables, Structure Initialization.

Strings:
Introduction to Strings, Declaring and initializing, Reading a String, Writing a String, String Functions (strstr, strcpy, strcmp, strlen)
SLE: List applications of arrays and structures. Writing Programs on above concept 9hrs

UNIT-5

defined Functions:
Need for User-defined Functions, A multi-function Program, Elements of User-defined Functions, User Definition of Functions, Return Values and their Types, Function Calls, Function Declaration, Parameter Passing technique-calling by value and calling by reference, Category of Functions.
SLE: Writing Set of Programs on above concept. 8hrs

UNIT – 6

Python Basics:
Our First Python Script, Comments, Basic Data type, Identifiers, Keywords, Variables, Basic Input and Output.

Python Control Structures
Getting Started with Programs, Decisions, Loops, Terminating Control .
TEXTBOOKS:

REFERENCE BOOKS:
1. *Introduction to Computer Science*, ITLEducation Solutions Ltd.
COMPUTER PROGRAMMING LABORATORY (0:0:3)

Sub code : CS0101
Hrs/week : 03
Max. Marks: 50

On Successful completion of this course, the students will be able to

1. Demonstrate the hardware components, DOS commands, mechanics of creating a Word document, the use of basic functions and formulas in Microsoft Excel and the basic mechanics of creating a PowerPoint presentation.
2. Apply and practice logical ability to solve the problems.
3. Understand C programming development environment, compiling, debugging, linking and executing a program using the development environment.
4. Analyzing the complexity of problems, modularize the problems into small modules and then convert them into programs.
5. Understand and apply the in-built functions and customized functions for solving the problems.
6. Understand and apply the pointers, memory allocation techniques and use of files for dealing with variety of problems.

PART – A

1. a. Identify the hardware components of a computer
   b. Execute basic DOS commands.
2. Create a document using a suitable word processing package, with at least three paragraphs and perform the following operations:
   1. Set left margin 1” and right margin 0.75”
   2. Centre the heading and make it bold. Increase the font size
   3. Underline the specified words in the document and change them to italics
   4. Conduct spell check and correct them suitably
   5. Demonstrate use of numbering and bullets
   6. Exchange paragraphs 2 and 3 using cut and paste facility
   7. Put suitable headers and footers
   8. Count the number of words and lines and demonstrate the use of drawing tools
   9. Include suitable logo/emblem/symbol
   10. Include the a table
3. Generate pay slip for different categories of employees (example: manager, assistant manager, supervisor, clerk etc) using spreadsheet which includes Basic pay, DA, HRA and other allowances along with deductions like PF, Professional Tax, and other deductions. DA and HRA are calculated in terms of percentage which keeps varying. Use formula while performing calculations. Each column and row should display total. Sort the name list in ascending order. Draw the bar chart to describe the DA percentage for different types of employees.
4. Create a PowerPoint presentation consisting of  
   Master slide generation  
   Minimum 8 slides  
   Information about your college  
   Include animation, Hyperlinks, Insert Header and Footer, Insert files, images, tables and Charts. Slide show features.  

5. Write a C program to find and output all the roots of a given quadratic equation, for non-zero coefficients. (Using if...else statement)  

6. Write a C program to simulate a simple calculator that performs arithmetic operations like addition, subtraction, multiplication, and division only on integers. Error message should be reported, if any attempt is made to divide by zero. (Using switch statement)  

7. Write a C program to reverse a given six digit integer number and check whether it is a palindrome or not. Output the given number with suitable message. (Using while statement)  

8. Write a C program to generate and print first 'N' Fibonacci numbers. (Using do-while statement)  

9. Write a C program to find the GCD and LCM of two integers and output the results along with the given integers. Use Euclid's algorithm. (Using looping constructs)  

10. Write a C program to find whether a given number is prime or not. Output the given number with suitable message. (Using for statement)  

PART - B  

11. Write a C program to input N real numbers in ascending order into a single dimension array. Conduct a binary search for a given key integer number and report success or failure in the form of a suitable message.  

12. Write a C program to input N integer numbers into a single dimension array. Sort them in ascending order using bubble sort technique. Print both the given array and the sorted array with suitable headings.  

13. Write a C program to read two matrices A(M x N) and B(P x Q) and perform addition. Output the input matrices and the resultant matrix with suitable headings and format. (Using two dimension arrays where array size M, N, P, Q ≤ 3)  

14. Write C user defined functions:  
   1. To input N integer numbers into a single dimension array.  
   2. To conduct a linear search.  

Using these functions, write a C program to accept the N integer numbers & given key integer number and conduct a linear search. Report success or failure in the form of a suitable message.  

15. Write C user defined functions:  
   - To input N integer numbers into a single dimension array.  
   - To sort the integers numbers in descending order using selection sort technique.  
   - To print the single dimension array elements.  

Using these functions, write a C program to input N integer numbers into a single dimension array, sort them in descending order, and print both the given array & the sorted array with suitable headings.  

16. Write C user defined functions:  
   To input N real numbers into a single dimension array.  
   Compute their mean.  
   Compute their variance.
Compute their standard deviation.
Using these functions, write a C program to input N real numbers into a single dimension array, and compute their mean, variance & standard deviation. Output the computed results with suitable headings.

17. Write a C-program that reads a string from keyboard and determines whether the string is a palindrome or not using:
   a. String functions
   b. user-defined functions (without C-library functions)

18. Write C user defined functions:
   - To read the elements of a given matrix of size M x N
   - To print the elements of a given matrix of size M x N
   - To compute the product of two matrices

   Using these functions, write a C program to read two matrices A (M x N) and B (P x Q) and compute the product of A and B after checking compatibility for multiplication. Output the input matrices and the resultant matrix with suitable headings and format. (Using two dimension arrays where array size M, N, P, Q ≤ 3)

19. a. Write a C program to create an integer variable, Assign value to the variable, create a pointer to this variable and print the value of the variable and address of the variable using pointer.
    b. Write a C program to create a structure called Employee with members Name, Job, Salary. Create a structure variable. Accept the input values for the structure members at run time. Suitably display the same.

20. Write a C program to create a C file called student.rec and store information about a student, in terms of his NAME, USN, BRANCH, and SEMESTER. Read the stored contents from the file and display on the screen.
ENGLISH ENHANCEMENT COURSE

Sub Code: EN0201
Hrs / Week: 02

CIE: 50% Marks
SEE: 50% Marks
Max Marks: 50

Course Prerequisites:
None

Course Outcomes:
Upon successful completion of this course, the student will be able to:
1. Conceptualize, Design and Develop good Presentations using technology. Will be innovative and creative
2. Develop an Inquisitive nature. Assimilate and voice their opinion with confidence and precision
3. Interpret different accents and speak accurately to a global audience
4. Apply and analyze new concepts in communication through self and peer appraisal for becoming successful professionals

Course Content:

Unit I: 8 Hours
- Language and Communication basics - LSRW
- Learning with VTU software – E Client
- Face to face and back to back communication
- Grammar concepts through common errors
- Presentation skills – Verbal and Visual
- Letter Writing

Unit II: 6 hours
- Group Discussion and Writing task
- Seek information on the given topic, assimilate and present / share opinions, facts, ideas etc within the given parameters. Topics will pertain to current issues
- Paraphrasing – Rewriting the given passage in own words without changing the content and meaning
- Writing task – Analyzing and Interpreting the data

Unit III: 8 hours
- S-T-A-R and Listening
- S-T-A-R : ( Speak – Transcribe – Analyze - Record ) – learn to speak in comprehensible accent, diction, without MTI and in correct grammar
- Listening skills : Understand different accents and develop a neutral comprehensible accent
Unit IV:
- Writing task
- Writing short passages
- Presentation skills and Grammar – through writing

TEXT BOOKS:
1. A Mirror of Common Errors by Ashok Kumar Singh, Publisher – Students’ Friends
2. English Grammar by Wren and Martin
3. I too had a dream – Verghese Kurien

REFERENCES:
1. King’s English – The first encyclopedia of English Language, Publishers – Addone
2. News Papers – English dailies
3. IELTS books
4. Film CDs of UGC – Role Model series
5. Internet sources

Note:
Handouts, Questionnaires and materials will be provided
DATA STRUCTURES WITH ‘C’ (4:0:0)

Sub code : CS0404
Hrs/week : 04
SEE Hrs : 03 Hours

CIE : 50% Marks
SEE : 50% Marks
Max. Marks: 100

Prerequisite: Programming in ‘C’

Course Outcome

On Successful completion of the course, the students will be able to:

1. Illustrate basic operations to be performed using pointers and dynamic memory allocations.
2. Demonstrate Abstract Data Types using an Arrays and recursion.
3. Illustrate various operations of different types of Queues.
4. Explain the basic operations of linked list and its implement using dynamic variables.
5. Demonstrate the operations and applications of Binary trees.
6. Apply different sorting and searching methods.

UNIT – 1

Pointer: Understanding pointers, Pointer variables, Accessing address of a variable, Initialization of pointer variables, Accessing variables through pointers, Chain of pointers, Pointer Arithmetic and arrays, Pointer increments and scale factor, Array of pointers, Pointer and character string, Pointers as function argument, Function returning pointers, Pointers to function, Pointers and structures. Dynamic memory allocation: malloc, calloc, free, realloc.

SLE: Pointer expressions, realloc.

9 Hours
UNIT – 2


SLE: Prefix, recursive chains 9 Hours

UNIT – 3

Queue: The Queue and its representation, Circular Queue, Priority Queue

SLE: Applications of Queues. 7 Hours

UNIT – 4

Linked Lists: Inserting and Removing nodes from a list, Implementation of stacks, getnode and freenode operations, Linked implementation of queues, Linked list as a data structure, Example of list operations, Header nodes, List in C, Array implementation of lists, Limitations of array implementation, Allocating and freeing dynamic variables, Queues as lists in C, Examples of list operations in C, Circular lists, Stack as a circular lists, Double linked list. Linked lists using dynamic variables.

SLE: Linked lists using dynamic variables, Primitive operations on circular lists, Queue as a circular list. 10 Hours

UNIT – 5


SLE: An Example: The Huffman algorithm, C Representation of trees, Tree traversal. 9 Hours
UNIT- 6

Sorting: Binary tree Sort, Heap sort, Insertion Sort, Shell Sort and Merge Sort.

Searching: Sequential Search, Binary Searches, Hashing-Hash Functions, Collisions, Collision resolution.

SLE: Radix Sorts, Separate chaining.

8 Hours

TEXT BOOK:


REFERENCE BOOKS:

Course Outcomes

On successful completion of the course the students will be able to:

**CO1**: Write programs that apply advanced C concepts like pointers, structures, dynamic memory allocation.

**CO2**: Implement stack data structure and use it for solving programming problems.

**CO3**: Implement queue data structure and use the same in solving real world problems.

**CO4**: Implement different types of linked lists and use them to develop applications.

**CO5**: Implement binary tree data structure and develop programs that use trees to store and access data efficiently.

**Syllabus Outline**: Students have to implement lab programs based on the following topics: Advanced used of pointers, Implementation and application of data structures: Stacks, Queues, Linked lists, Binary Trees.
OBJECT ORIENTED PROGRAMMING WITH C++ (4:0:0)

Sub code : CS0405  
Hrs / week : 04  
SEE Hrs : 3 Hrs

CIE : 50 %  
SEE : 50 %  
Max. Marks : 100

Prerequisite: Computer concepts and C Programming.

Course Outcome

On successful completion of the course the students will be able to

1. Explain principles of Object Oriented Programming and write simple C++ programs using class and object.
2. Apply advanced features of C++ classes for programming.
3. Make use of overloading for designing classes that are syntactically resembling built-in data types.
4. Build reusable and extensible classes using inheritance and virtual functions.
5. Apply templates and exception handling for implementing highly generic and robust C++ programs.
6. Develop applications using advanced C++ features such as STL and RTTI.

UNIT 1:


Classes and Objects(Introduction): C Structures revisited, Specifying a Class, Defining member functions, A C++ Program with class.

Self Learning Exercise: Dynamic Initialization of variables, Scope resolution operator. 8 Hrs
UNIT 2:

Classes and Objects (Advanced): Relationship of Structure, Union and Class in C++, Friend Functions, Friend Classes, Inline Functions- Defining Inline Functions Within a Class, Constructors and Destructors, Parameterized Constructors- Constructors with One Parameter: A Special Case, Static Class Members- Static Data Members and Static Member Functions, When Constructors and Destructors Are Executed, Nested Classes, Local Classes, Passing Objects to Functions, Returning Objects, Object Assignment.

Arrays, pointers, References, and the Dynamic Allocation Operators: ‘this’ Pointer, References – Reference Parameters, Passing References to Objects and Returning References, C++’s Dynamic Allocation Operators- Initializing Allocated Memory, Allocating Arrays and Allocating Objects.

Self Learning Exercise: Arrays of Objects- Creating Initialized vs. Uninitialized Arrays, Pointers to Objects.

10 Hrs

UNIT 3:

Function Overloading, Copy Constructor, and Default Arguments: Function Overloading, Copy Constructor, Default Function Arguments, Default Argument vs. Overloading.

Operator Overloading: Creating a Member Operator Function- Creating Prefix and Postfix Forms of the Increment and Decrement Operators, Operator Overloading Using a Friend Function – Using a Friend to Overload ++ or --, Friend operator Functions Add Flexibility, Overloading << and >>.

Self Learning Exercise: Overloading Some Special Operators([ ], ( ), ->) , Overloading new and delete

8 Hrs

UNIT 4:

Inheritance: Base-Class Access Control, Inheritance and protected Members- Protected Base-Class Inheritance, Inheritance Multiple Base Classes, Constructors, Destructors, and Inheritance- When Constructors and Destructors Are Executed, Passing Parameters to Base-Class Constructors. Granting Access, Virtual Base Classes.

Virtual Functions and Polymorphism: Virtual Functions- Calling a Virtual Function Through a Base-Class Reference, The Virtual Attribute vs. Inherited, Virtual Functions Are Hierarchical, Pure Virtual Functions- Abstract Classes, Using Virtual Functions, Early vs. Late Binding.

Self Learning Exercise: Granting access (Explicitly granting access to base class members in derived)

8 Hrs
UNIT 5


Exception Handling: Exception Handling Fundamentals- Catching Class Types, Using Multiple Catch Statements.

C++ File I/O: <fstream> and the File Classes, Opening and Closing a File, Reading and Writing Text Files.

Self Learning Exercise: Handling Derived-Class Exceptions, Exception Handling Options. Compacting an Array 10 Hrs

UNIT 6:

Run-Time Type ID and the Casting Operators: Run-Time Type Identification (RTTI), typeid Applied to Template Classes, Casting Operators: dynamic_cast, const_cast, static_cast, reinterpret_cast.


Self Learning Exercise: List (Basics of list container class) 8 Hrs

TEXT BOOKS:


REFERENCES:

3. WEBSITE: http://www.cplusplus.com/
OBJECT ORIENTED PROGRAMMING LAB (0:0:3)

Sub code : CS0103  

Hrs/week : 03

Course Outcome

On successful completion of the course the students will be able to

1. Understand object-oriented concepts and how they are supported by C++ and Implementation issues related to object-oriented techniques.

2. Demonstrate the ability to analyze, use, and create functions, classes, to overload operators, and use inheritance and Pointers when creating or using classes and create templates

3. Demonstrate the ability to understand and use Exception handling and file handling mechanism.

4. Design and write programs that make appropriate use of advanced object-oriented facilities common to many object-oriented languages such as classes, message passing, overloading and inheritance.
OPERATING SYSTEMS (4:0:0)

Sub code : CS0413
Hrs / week : 04
SEE Hrs : 3 Hrs
CIE : 50 %
SEE : 50 %
Max. Marks : 100

Course Outcome
On successful completion of the course the students will be able to

1. Explain the differences between processes and threads and analyze issues of scheduling user-level processes/threads.
2. Identify the issues in inter-process communication
3. Analyze the design issues of memory management
4. Identify the design and implementation issues in virtual memory management
5. Explain the concepts of file system and its implementation
6. Explain the concepts of deadlock and analyze how they can be managed/avoided

UNIT 1
INTRODUCTION, PROCESSES AND THREADS: WHAT IS AN OPERATING SYSTEM?: The Operating system as an extended machine, The Operating system as a resource manager.
SLE: Thread scheduling

9 Hours
UNIT 2
INTERPROCESS COMMUNICATION
Race conditions, Critical regions, Mutual exclusion with busy waiting, Sleep and wakeup, Semaphores, Mutexes, Monitors, Message passing. CLASSICAL IPC PROBLEMS: The dining philosophers problem, The readers and writers problem.
Memory Management: No Memory Abstraction, A Memory Abstraction: Address Spaces - The Notion of an Address Space, Swapping, Managing Free Memory.
SLE: Barriers

UNIT 3
SLE: Comparison of Page Replacement Algorithms

UNIT 4
DESIGN ISSUES FOR PAGING SYSTEMS: Local versus Global allocation policies, Load control, Page size, Separate instruction and data spaces, Shared pages, Virtual memory interface.
SLE: Separation of policy and mechanism.
UNIT 5
SLE: Defragmenting Disks

9 Hours

UNIT 6
DEADLOCKS: INTRODUCTION TO DEADLOCKS: Conditions for deadlock, Deadlock modeling. THE OSTRICH ALGORITHM, DEADLOCK DETECTION AND RECOVERY: Deadlock detection with one resource of each type, Deadlock detection with multiple resource of each type, Recovery from deadlock, DEADLOCK AVOIDANCE: Resource trajectories, Safe and Unsafe States, The Banker's algorithm for a single resource, The banker's algorithm for multiple resources, DEADLOCK PREVENTION: Attacking the mutual exclusion condition, Attacking the Hold and Wait condition, Attacking the No Preemption condition.
SLE: Attacking the Circular Wait condition

9 Hours

TEXT BOOK:

REFERENCE BOOKS:


DATABASE LABORATORY (0:0:3)

Sub code: CS0107 Hrs/week: 03

Course Outcome:

At the end of this course the student will be able to

1. Explain his/her knowledge of database technology, its importance, its architectures, and the central role Database technology plays in Information Systems.

2. Apply appropriate development methodologies of data analysis, design and use appropriate modeling techniques for databases.

3. Demonstrate query facilities to formulate queries and manipulate the database e.g. Structured Query Language (SQL)

4. Appreciate the issues underlying database implementation in any database

5. Concentrate on a methodology for good database design and practical experience in designing and implementing standalone database system.
SOFTWARE ENGINEERING (4:0:0)

Sub code: CS0417
Hrs/week: 04
SEE Hrs: 03 Hours
CIE : 50 %
SEE : 50 %
Max. Marks : 100

Course Outcome
On successful completion of the course the students will be able to
1. Describe the fundamental knowledge of Software Engineering Process.
2. Discuss the development guidelines for Agile View of Process and System Engineering.
3. Determine the key steps in gathering requirement for a problem and building an Analysis Model.
4. Provide the set of design principles and practices to build a high quality system.
5. Validate System functions using various Testing Skills
6. Identify an effective estimation model for Software Project Management.

UNIT 1
SLE: Software development life cycle.

9 Hours
UNIT 2
SLE: Computer Based Systems.

8 Hours

UNIT 3
Requirement Engineering: Requirement Engineering Tasks, Requirement Engineering Process, Eliciting Requirements, Developing USE-CASE.

SLE: Validating requirements.

9 Hours

UNIT 4

SLE: The Design Model.

8 Hours
UNIT 5

**Testing Tactics:** Software Testing Fundamentals, Black Box & White Box Testing, Basis Path Testing, Control Structure Testing, Black Box Testing.

**SLE:** Validating Testing.

9 Hours

UNIT 6
**Project Management:** Project Management Spectrum, People, Product, Process, Project.


**SLE:** Resources

9 Hours

**TEXT BOOK**

**REFERENCE BOOKS**
COMPUTER NETWORKS (4:0:0)

Sub code: CS0418
Hrs / week: 04
SEE Hrs: 03 Hours

CIE: 50 %
SEE: 50 %
Max. Marks: 100

Prerequisite: Data Communications and Networking

Course Outcomes:

On successful completion of the course the students will be able to

1. Discuss the network layer services and learn the concepts of IP address, Classes, IPv4 Datagram.
2. Learn the auxiliary protocol in IPV4 and working of different Unicast Routing algorithms and protocols.
3. Discuss the concepts and description of IPV6 and various Multicast routing protocols.
4. Identify the functions of the Transport layer and analyze the performances of UDP and TCP protocols.
5. Explain the working of SCTP and congestion control TCP.
6. Distinguish between various application layer protocols and analyze the need for information on DNS.

UNIT 1


9 hours
UNIT 2
SLE: Path Vector Routing Algorithm

UNIT 3
SLE: Transition from IPv4 to IPv6

UNIT 4
SLE: UDP Applications
UNIT 5
SLE: Tahoe TCP

UNIT 6
SLE: Web documents

TEXT BOOK:


REFERENCE BOOKS:

WEB TECHNOLOGIES AND APPLICATIONS (4:0:0)

Sub code : CS0450
Hrs / week : 04
SEE Hrs : 03 Hours
CIE : 50 %
SEE : 50 %
Max. Marks: 100

Prerequisite: Object Oriented Programming Concepts

Course outcome
On successful completion of the course the students will be able to

1. Explain the importance of java in the internet, its difference compared with other languages and cite examples using data types, variables, operators, control statements, arrays and vectors
2. Compare and demonstrate the use of different category of classes, Recognize group of classes with different visibility control to create package; identification of errors and handling them using exceptional handling;
3. Analyze working of I/O operations, compare functions of string and stringbuffer class
4. Prepare web pages using HTML/XHTML and CSS.
5. Creation of dynamic web pages using CGI script. Illustrate how to create database using MYSQL and link them with perl script.
6. Demonstrate how to embed php into HTML and how to link php with MYSQL to create dynamic pages

UNIT 1
The Java Language – 1: The Genesis of Java: Java's Lineage, The creation of Java, Why Java is important to the Internet? Java's Magic, Java features; An overview of Java Programming: Object oriented programming, simple java program, Lexical issues; Data types, Variables and Arrays: Simple types, Literals,
Variables, Arrays, Operators, Control Statements, Introduction to Java Classes: Classes, Object, Methods, Constructors, The this keyword, Garbage collection, The Finalize() Method, A Stack Class.

SLE: Type conversions and Casting

UNIT 2

The Java Language - 2: Methods and Classes; Inheritance; Inheritance Basic, Using Super, Creating Multilevel hierarchy, When Constructors are Called, Method Overriding, Dynamic Method Dispatch, Using Abstract Classes, Using Final, the Object Class; Java Packages and Interfaces; Exception Handling; Multithreading Programming.

SLE: Creation of User defined package

9 Hours

UNIT 3


SLE: StringBuffer Class.

9 Hours

UNIT 4

XHTML, XML and CSS: XHTML: Basic syntax; Standard XHTML document structure; Basic text markup, Images; Hypertext Links; Lists; Tables; Forms; Frames; CSS: Introduction; Levels of style sheets; Style specification formats; Selector forms; Property value forms; Font properties; List properties; Color; Alignment of text; The Box model; Background images; The and tags; Conflict resolution. XML: Introduction; Syntax; Document structure;
Document Type definitions; Namespaces; XML schemas; Displaying raw XML documents; Displaying XML documents with CSS.

SLE: Creation of Web Pages

8 Hours

UNIT 5

CGI and MYSQL: Introduction to Apache Web Server-Introduction, Starting, stopping and Restarting Apache, Apache Log Files, MYSQL: Introduction, Tutorial, DBI, Table Joins, CGI - Introduction, A Simple CGI Program, what can go wrong in the program? CGI.pm: Introduction, CGI.pm HTML Shortcuts, Information received by the CGI program, Form Widget Methods, Security consideration, A note on die().

SLE: Loading and Dumping a Database, Apache Configuration

9 Hours

UNIT 6

PHP and MYSQL: PHP Introduction, Embedding PHP into HTML, Configuration, Quick examples, Language Syntax, Built-IN PHP Functions, PHP and MySQL.

SLE: Connecting PHP to Mysql

8 Hours

TEXT BOOKS


3. Open Source Web Development with Lamp, James Lee and Brent Ware, Pearson Education, 2009
REFERENCE BOOKS


WEB TECHNOLOGIES AND APPLICATIONS LABORATORY (0:0:3)

Sub code: CS0112

Hrs/week: 03

Course Outcome:

At the end of this course the student will be able to

1. Implement programs on Arrays, Stack, Class and Methods

2. Demonstrate importance of Interface, Packages, Multithreading, Exception handling

3. Explore working of I/O operations, compare functions of string and stringbuffer class

4. Prepare web pages using HTML/XHTML and CSS.

5. Design of dynamic web pages using CGI script with connection to MYSQL

6. Show how to embed php into HTML and how to link php with MYSQL
Course Outcomes:

At the end of this course the student will be able to

1. Implement the basic Concepts of Networking like
   (i) Usage of different devices.
   (ii) Connecting systems.
   (iii) Use any packet analyzer to determine the various Networking factors.
   (iv) Prepare a write up about the CSLAB NETWORK Set-up.

2. Develop Networks using NS2 or equivalent simulators in world scenarios in a project-based approach.
INTRODUCTION TO C# PROGRAMMING AND .NET CONCEPTS (4:0:0)

Sub code : CS0426
Hrs/week : 04
SEE Hrs : 03 Hours
CIE : 50% Marks
SEE : 50% Marks
Max. Marks : 100

Course Outcome

On successful completion of the course the students will be able to

1. Explain the basics of .Net platform and the role of base class libraries, role of common intermediate language and namespaces and also to Define and deploy the different command line compiler options.
2. Discuss the fundamentals of c# and to build the basic c# program using different constructs.
3. Review the basic pillars of object oriented programming concepts and apply the exception handling technique to handle different types of errors.
4. Provide the knowledge about basics of object lifetime and to define the use of interfaces and collections.
5. Explain the callback interfaces, delegates, advanced C# keywords and its implementation.
6. Build and deploy local and shared assemblies.

UNIT 1
THE PHILOSOPHY OF .NET: Understanding the Previous State of Affairs, The .NETSolution, The Building Block of the .NET Platform (CLR,CTS, and CLS), The Role of the .NET Base Class Libraries, What C# Brings to the Table, An Overview of .NET Binaries ( aka Assemblies ), the Role of the Common Intermediate Language , The Role of .NET Type Metadata, The Role of the Assembly Manifest, Compiling CIL to Platform
Specific Instructions, Understanding the Common Type System, Intrinsic CTS Data Types, Understanding the Common Languages Specification, Understanding the Common Language Run-time A tour of the .NET Namespaces, Increasing Your Namespace Nomenclature.


SLE: Deploying the .NET Run-time, An Interesting Aside The System. Environment Class.

9 Hours

UNIT 2

SLE: Programming using basic constructs of c# 9 Hours
UNIT 3
OBJECT- ORIENTED PROGRAMMING WITH C#: Forms
Defining of the C# Class, Definition the "Default Public Interface" of a Type, Recapping the Pillars of OOP, The First Pillars: C#’s Encapsulation Services, Pseudo- Encapsulation: Creating Read-Only Fields, The Second Pillar: C#’s Inheritance Supports, keeping Family Secrets: The "Protected" Keyword, Nested Type Definitions, The Third Pillar: C #’s Polymorphic Support, Casting Between. EXCEPTIONS : Ode to Errors, Bugs, and Exceptions, The Role of .NET Exception Handling, the System. Exception Base Class, Throwing a Generic Exception, Catching Exception, CLR System – Level Exception (System. System Exception), Custom Application-Level Exception (System. System Exception), Handling Multiple Exception, The Family Block, the Last Chance Exception Dynamically Identifying Application – and System Level Exception Debugging System.
SLE: Exception Using VS.NET

8 Hours

UNIT 4
INTERFACES AND COLLECTIONS: Defining Interfaces Using C# Invoking Interface Members at the object Level, Exercising the Shapes Hierarchy, Understanding Explicit Interface Implementation, Interfaces As Polymorphic Agents, Building Interface Hierarchies, Implementing, Implementation, Interfaces Using VS .NET, understanding the IConvertible Interface, Building a Custom Enumerator (IEnumerable and Enumerator), Building Cloneable objects (ICloneable), Building Comparable Objects (I Comparable), Exploring the system. Collections Namespace,Building a Custom Container (Retrofitting the Cars Type).
UNIT 5

CALLBACK INTERFACES, DELEGATES, AND EVENTS, ADVANCED TECHNIQUES: Understanding Callback Interfaces, Understanding the .NET Delegate Type, Members of System. Multicast Delegate, The Simplest Possible Delegate Example, Building More a Elaborate Delegate Example, Understanding Asynchronous Delegates, Understanding (and Using) Events.

The Advances Keywords of C#, A Catalog of C# Keywords Building a Custom Indexer, A Variation of the Cars Indexer Internal Representation of Type Indexer. Using C# Indexer from VB .NET. Overloading operators, The Internal Representation of Overloading Operators, interacting with Overload Operator from Overloaded- Operator- Challenged Languages, Creating Custom Conversion Routines, Defining Implicit Conversion Routines, The Internal Representation of CustomsConversion Routines.

SLE: Building callback interfaces, delegates and events using VS.NET

9 Hours

UNIT 6

UNDERSTANDING .NET ASSEMBLIES: Problems with Classic COM Binaries, A Overview of .NET Assembly, Building a Simple File Test Assembly, A C# . Client Application, A Visual Basic .NET Client Application, Cross Language Inheritance, Exploring the CarLibrary’s, Manifest, Exploring the CarLibrary’s Types, Building the Multifile Assembly, Using Assembly, Understanding Private Assemblies, Probing for Private Assemblies (The Basics), Private A Assemblies XML Configurations Files, Probing for Private Assemblies (The Details), Understanding Shared Assembly, Understanding Shared Names, Building a Shared
Assembly, Understanding Delay Signing, Installing/Removing Shared Assembly, Using a Shared Assembly.

**SLE:** Building a custom assemblies.

8 Hours

**TEXT BOOKS**


**REFERENCE BOOKS:**

INTRODUCTION TO ANDROID PROGRAMMING
(4:0:0)

Sub code : CS0443
Hours / week : 04
SEE Hours : 3 Hours

CIE : 50 %
SEE : 50 %
Max. Marks : 100

Pre-requisite: DBMS, Java Programming

Course outcomes:
On successful completion of the course the students will be able to
1. Understand the basic versions, the evolution of Android as a Mobile OS, create simple apps and apply different styles.
2. Experiment with the User Interface of Android.
3. Manipulate the UI with different common elements.
4. Apply the knowledge to a real time Internet download application.
5. Able to learn how to persist data using preferences.
6. Apply the knowledge of DBMS to persist data.

UNIT 1

59
Sophisticated Progress Dialog, Linking Activities Using Intents, Resolving Intent Filter, Collision, Returning Results from an Intent, Passing Data Using an Intent Object, Fragments, Adding Fragments Dynamically, Life Cycle of a Fragment, Interactions between Fragments, Calling Built-In Applications Using Intents, Understanding the Intent Object, Using Intent Filters, Adding Categories, Displaying Notifications
SLE: Designing a Android Application

9 Hours

UNIT 2
GETTING TO KNOW THE ANDROID USER INTERFACE:
Understanding the Components of a Screen, Views and View Groups, Linear Layout, Absolute Layout, Table Layout, Relative Layout, Frame Layout, Scroll View, Adapting to Display Orientation, Anchoring Views, Resizing and Repositioning, Managing Changes to Screen Orientation, Persisting State Information during Changes in Configuration, Detecting Orientation Changes, Controlling the Orientation of the Activity, Utilizing the Action Bar, Adding Action Items to the Action Bar, Customizing the Action Items and Application Icon, Creating the User Interface Programmatically, Listening for UI Notifications, Overriding Methods Defined in an Activity, Registering Events for Views. DESIGNING YOUR USER INTERFACE WITH VIEWS:
SLE: Designing User Interface

9 Hours
UNIT 3
SLE: Designing Menus in Android
8 Hours

UNIT 5
USING INTERNET RESOURCES: Downloading and Parsing Internet Resources, Connecting to an Internet Resource, Parsing XML Using the XML Pull Parser, Creating an Earthquake Viewer, Using the Download Manager, Downloading Files, Customizing Download Manager Notifications, Specifying a Download Location, Cancelling and Removing Downloads, Querying the Download Manager, Using Internet Services, Connecting to Google App Engine, Best Practices for Downloading Data Without Draining the Battery
SLE: Using Internet Resources to develop Android Applications
9 Hours

UNIT 5
FILES, SAVING STATE, AND PREFERENCES:
Earthquake Viewer, Persisting the Application Instance State, Saving Activity State Using Shared Preferences, Saving and Restoring Activity Instance State, Using the Lifecycle Handlers, Saving and Restoring Fragment Instance State, Using the Lifecycle Handlers, Including Static Files as Resources, Working with the File System, File-Management Tools Using Application-Specific Folders to Store Files, Creating Private Application Files, Using the Application File Cache, Storing Publicly Readable File

SLE: Creating Private Application Files 9 Hours

UNIT 6
DATABASES AND CONTENT PROVIDERS: Introducing Android Databases, SQLite Databases, Content Providers, Introducing SQLite, Content Values and Cursors, Working with SQLite Databases, Introducing the SQLiteOpenHelper, Opening and Creating Databases Without the SQLiteOpenHelper, Android Database Design Considerations, Querying a Database, Extracting Values from a Cursor, Adding, Updating, and Removing Rows, Inserting Rows, Updating Rows, Deleting Rows, Creating Content Providers, Registering Content Providers, Publishing Your Content Provider's URI Address, Creating the Content Provider's Database, Implementing Content Provider Queries, Content Provider Transactions, Storing Files in a Content Provider, A Skeleton Content Provider Implementation, Using Content Providers, Introducing the Content Resolver, Querying Content Providers, Querying for Content Asynchronously Using the Cursor Loader, Introducing Loaders, Using the Cursor Loader, Adding, Deleting, and Updating Content, Inserting Content, Deleting Content, Updating Content, Accessing Files Stored in Content Providers, Creating a To-Do List Database and Content Provider
SLE: Usage of SQLite Databases 8 Hours
TEXT BOOKS
1. Wei-Meng Lee, Beginning Android 4 Application Development, Wrox, Wiley India Edition. (Chapters:1,2,3,4,5)
2. Reto Meier, Professional Android 4 Application Development, Wrox, Wiley India Edition (Chapters:6,7,8)

REFERENCE BOOKS:
2. Android Programming: The Big Nerd Ranch Guide (Big Nerd Ranch Guides)
3. Paul Deitel, Harvey Deitel, Abbey Deitel, and Michael Morgano, Android for Programmers: An App-Driven Approach
PYTHON PROGRAMMING (3:0:2)

Sub Code : CS0452
Hours/Week : 05
SEE Hours : 03 Hours

Prerequisite: Computer Programming Concepts

Course Outcomes:
On Successful Completion of the Course, the students will be able to:

1. Solve problems by applying basic Python programming.
2. Use Numeric types, String and Lists in Python for solving application – specific programs.
3. Use Dictionaries and Tuples for organizing and processing data in Python programs.
4. Analyze the use of unanimous functions to reduce the code length.
5. Develop and manipulate persistent files and databases using Python programs.
6. Develop user-defined classes in Python, instantiate and use them in programs.

UNIT 1
Introduction:
Why do people use Python? Who uses Python today? What can I do with Python? What are Python's technical strengths?
Variable names and keywords, Operators and operands, Expressions and statements, Interactive mode and script mode, Order of operations, String operations, Comments.

Variables, expressions and statements: Variables, expressions and statements: Values and types, Variables, Variable names and keywords, Operators and operands, Expressions and statements, Interactive mode and script mode, Order of operations, String operations, Comments.
Functions: Function calls, Type conversion functions, Math functions, Composition, Adding new functions, Definitions and uses, Flow of execution, Parameters and arguments, Variables and parameters are local, stack diagrams, Fruitful functions and void functions.

Conditionals, recursion and iteration:
Modulus operator, Boolean expressions, Logical operators, Conditional execution, Alternative execution, Chained conditionals, Nested conditionals, Recursion, Stack diagrams for recursive functions, Infinite recursion, Keyboard input, The while statement, The for loop
SLE: break, continue, pass, and the Loop else

UNIT 2
Numeric Types:
Numeric type basics, Numbers in action – Display formats, Division, Integer Precision, Complex numbers, Hex, Octal, Binary, Bitwise Operations, Built-in Numeric tools, Other Numeric Types – Decimal Type, Fraction Type, Sets, Booleans

Strings:
Introduction, function: len, Traversal with a while/for loop, String slices, Strings are immutable, Searching, Looping and counting, String methods, The in operator, String comparison.
Lists: Introduction, Lists are mutable, Traversing a list, List operations, List slices, List methods, Map, filter and reduce, Deleting elements, Lists and strings, Objects and values.
SLE: Aliasing, List arguments

UNIT 3
Dictionaries:
Introduction, Dictionary as a set of counters, Looping and dictionaries, Reverse lookup, Dictionaries and lists, Memos, Global variables, Long integers.
Tuples:
Introduction: Tuples are immutable, Tuple assignment, Tuples as return values, Variable-length argument tuples, Lists and tuples, Dictionaries and tuples.
SLE: Comparing tuples, Sequences of sequences.

UNIT 4
Python Essentials:
Iterations and Comprehensions, The lambda function, The map function, The reduce function, The filter function. Importing User-defined modules, Command-line arguments, Positional arguments, Keyword arguments, Functions as 'First Class' Citizens, Generators, Uses of __ and __ in Names, Handling errors with try and except.
SLE: Decorators, Solving problems using the above features.

UNIT 5
Files:
Persistence, Reading and writing, Format operator, Filenames and paths, Catching exceptions, Databases, Pickling.
SLE: Pipes, Writing modules.

UNIT 6
Classes and objects: User-defined types, Attributes, Example: Rectangle, Instances as return values, Objects are mutable, Copying. Classes, functions and methods: Time example, Pure functions, Modifiers, Object-oriented features, Printing objects, examples, The init method, The __str__ method, Operator overloading.
Combining Classes: Inheritance, Composition, Polymorphism
SLE: Type-based dispatch, Future Directions.

10 Hours
TEXT BOOKS:
1. Learning Python, Mark Lutz, O'Reilly Media, 5th Ed.
2. Think Python - How to Think Like a Computer Scientist, Allen Downey, Green Tea Press, 2nd Edition
   (2.0.17)

REFERENCE BOOKS:

WEB LINKS:
1. Official Python Documentation: https://docs.python.org
2. Full Stack Python: https://www.fullstackpython.com/
3. Think Python codebase: http://thinkpython.com/code
BIG DATA ANALYTICS(3:0:2)

Sub code : CS0444
Hours / week : 05
SEE Hours : 3 Hours
CIE : 50 %
SEE : 50 %
Max. Marks : 100

Course Outcome
On successful completion of the course the students will be able to
1. Overview of Big Data and Related Technologies
2. Analyze Technologies for Handling Big Data and Hadoop Ecosystem
3. Acquire clear understanding of Map Reduce Fundamentals and HBase
4. Acquire clear understanding of NoSQL Data Management
5. Acquire a clear understanding of Analytics and Big Data
6. Analyze the various Analytical Approaches Tools to Analyze Data and Exploring R

UNIT 1
SLE: Business Intelligence, Preventing Fraud Using Big Data Analytics

8 Hours

UNIT 2
Computing Technology for Big Data, Hadoop Ecosystem, Hadoop Distributed File System, HDFS Architecture, Features of HDFS, MapReduce, Features of MapReduce, Hadoop YARN.

SLE: HBase, Hive, Pig, Sqoop, Flume

UNIT 3

SLE: Installation of HBase

8 hours

UNIT 4
NoSQL Data Management: Introduction to NoSQL, Characteristics of NoSQL, Evolution of Databases, Aggregate Data Models, Key Value Data Model, Document Databases, Relationships, Graph Databases, Schema-Less Databases, Materialized Views, Distribution Models, Sharding, MapReduce Partitioning and Combining, Composing MapReduce Calculations.

SLE: CAP Theorem

8 hours

UNIT 5
UNIT 6
SLE: Installing R, R Studio

TEXTBOOK:
1. Big Data: Black Book, DT Editorial Services, Wiley India Pvt Ltd, 2015 Edition (Chapters 1, 3, 4, 5, 16, 17, 18, 19)

REFERENCE BOOKS:
2. Big Data Analytics with R and Hadoop, Vignesh Prajapati, -Packt Publishing 2013


6. Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data EMC Education Services, Wiley India Pvt Ltd
Prerequisite: File Systems

Course Outcome:

On successful completion of the course the students will be able to
1. Describe different database models and creation of MySQL database.
2. Design the database using Entity Relationship model and Normalization.
3. Write the SQL query to interact with the database and illustrate transaction concepts.
4. Create triggers, constraints and objects in MySQL.
5. Demonstrate the use of Indexes and Views.
6. Differentiate different types of failures and recovery techniques.

UNIT 1

Introduction: Advantages of using a DBMS approaches, Definition of schema, data model and instances, Three-schema architecture and data independence, Different Database Models, Interfacing with a Database, The Mechanics, Disk versus Main-Memory,

An Example Database: The Everest Books Database

Relational Databases (with a MySQL Flavor): MySQL Database System, Database Organization, Creating and using the Database,

SLE: Steps in Designing a Database.

9 Hours
UNIT 2
Manipulating the Database: Example Tables, Relational Algebra.

Database Design 1: The Entity-Relationship Model- Entity, Entity Sets, Attributes, Relationships, Constraints on Relationship types, Weak entity type, Notations for ER Diagrams, Modeling The Everest Books' Database

Database Design 2: Definition of Functional Dependencies, Normal Forms Based on Primary Keys - 1NF, 2NF, 3NF, BCNF SLE: Ad Hoc Database Design

9 Hours

UNIT 3
SQL: Basics, Data Definition Language, Data Manipulation Language, Stored Routines, Formatting and SQL.
Transactions: Informally Speaking, Transactions in SQL, Transaction Semantics, Serializability, Locks, SQL Isolation Levels, Transactions in MySQL
SLE: Orders, Invoices & Reports

8 Hours

UNIT 4
Constraints: Constraints in SQL, Constraint Check Time.
Triggers: Triggers in SQL.
Objects: Objects in SQL, Referencing Objects
SLE: Comparing Objects.

9 Hours

UNIT 5
Indexes: How does an Index Work?, Types of Indexes, Different Types of Indexes, Hash Indexes, Indexes in MySQL,
Views: Two Implementations of Views, Views in MySQL, Examples.
SLE: Some Thoughts on Indexing. 9 Hours

UNIT 6
Spatial Databases: MySQL, Example
Logs and Recovery: Failure Types, Logs, Log Details, Some Log Characteristics, Database Recovery, Database Recovery Example, MySQL.
SLE: Spatial Databases: Example 8 Hours

TEXT BOOKS:

2. Fundamentals of Database Systems, Elmasri and Navathe, Addison-Wesley, 5th Edition, 2007. (1.1, 1.6, 2.2, 3.3, 3.4.3, 3.5, 3.7.1, 10.2.1, 10.3, 10.5)

REFERENCE BOOKS:

VII SEMESTER

CLOUD COMPUTING (3:0:2)

Sub code : CS0437
Hrs/week : 05
SEE Hrs : 3 Hrs

CIE : 50%
SEE : 50%
Max Marks : 100

Course Outcome
On successful completion of the course the students will be able to

1. Overview of Cloud Computing and various distributed system models with enabling technologies.
3. Acquire the clear understanding of Virtual Machines and Virtualization of Clusters.
4. Acquire the basic knowledge of Cloud Platform Architecture over Virtualized Data Centers.
5. Acquire the clear understanding of Service-Oriented Architectures for Distributed Computing.
6. Develop various applications using Cloud Programming and Software Environments.

UNIT 1

SLE: Role of Open Standards in Cloud Computing.  8 Hours

UNIT 2

SLE: Tianhe-1A.  8 Hours

UNIT 3
Virtual Machines and Virtualization of Clusters and Data Centers: Implementation Levels of Virtualization, Virtualization Structures/Tools and Mechanisms, Virtualization of CPU, Memory, and I/O Devices, Virtual Clusters and Resource Management, Virtualization for Data-Center Automation.

SLE: Cloud OS for Virtualized Data Centers.  10 Hours

UNIT 4
SLE: Google App Engine.

9 Hours

UNIT 5
Service-Oriented Architectures for Distributed Computing: Services and Service-Oriented Architecture, Message-Oriented Middleware, Portals and Science Gateways, Discovery, Registries, Metadata, and Databases, Workflow in Service-Oriented Architectures.

SLE: Semantic Web and Grid

9 Hours

UNIT 6

SLE: Installation of Open stack.

8 Hours

Textbooks:
1. Cloud Computing Bible by Barrie Sosinsky, Wiley India (Chapter 1).
2. Distributed And Cloud Computing, Hwang, Kai; Fox, Geoffrey C; Dongarra, Jack J. ELSEVIER INDIA PVT. LTD (Chapter 1, 2, 3, 4, 5, 6)
Reference Books:
1. **Cloud Computing for Dummies** by Judith Hurwitz, R. Bloor, M. Kanfman, F. Halper (Wiley India Edition).
2. **Cloud Security** by Ronald Krutz and Russell Dean Vines, Wiley-India.
4. **Google Apps** by Scott Granneman, Pearson
CRYPTOGRAPHY (3:2:0)

Sub code : CS0421
Hrs/week : 05
SEE Hrs : 3 Hrs

CIE : 50%
SEE : 50%
Max Marks : 100

Course Outcome
On successful completion of the course the students will be able to
1. Explain the foundations of Cryptography and network security. Critical evaluation of the risks and threats to networked computers. Demonstration of the various mechanisms to protect data and their limitations.
2. Demonstrate detailed knowledge of the role of encryption to protect data using DES.
3. Demonstrate advanced encryption technique like AES.
4. Familiar with cryptographic techniques for secure (confidential) communication of two parties over an insecure (public) channel using public key algorithms.
5. Verify the integrity of the messages transmitted via an insecure channel and unique identification of the originator of any message using MAC.
6. Explain the different key management and digital signature systems.

UNIT 1
Introduction to Symmetric-Key Encipherment: Security Goals, Cryptographic Attacks, Services and mechanism, Techniques.
Traditional Symmetric-Key Ciphers: Introduction, Substitution Ciphers, Transposition Ciphers.
Mathematics of Cryptography: Integer arithmetic, Modular arithmetic, Matrices, Linear Congruence.

SLE: Stream and Block Ciphers,

9 Hours
UNIT 2
Introduction to Modern Symmetric-Key Ciphers: Modern Block Ciphers, Modern Stream Ciphers.

Data Encryption Standard (DES): Introduction, DES Structure (overview only), Security of DES.

SLE: Multiple DES-Convensional Encryption Algorithms. 9 Hours

UNIT 3
Advanced Encryption Standard (AES): Introduction, Transformations, Key expansion, The AES Ciphers, Examples, Analysis of AES.

Encipherment Using Modern Symmetric-Key Ciphers: Use of Modern Block Ciphers (overview only).

SLE: Use of Stream Ciphers (overview only). 8 Hours

UNIT 4
Asymmetric-Key Cryptography: RSA Cryptosystems, Elliptic curve crypto systems (overview only).

SLE: ElGamal Cryptosystem. 9 Hours

UNIT 5
**Digital Signature**: Comparison, Process, Services, Attacks of Digital Signature, DigitalSignature Schemes.

**SLE**: Variations and Applications.  
9 Hours

**UNIT 6**


**Key Management**: Symmetric-Key Distribution, Kerberos, Symmetric-key Agreement.

**SLE**: Public-Key Distribution.  
8 Hours

**Note**: Implementation of the concepts taught in the theory classes will be done in Tutorial sessions.

**Text Book**


**Reference Books**

MANAGEMENT AND ENTREPRENEURSHIP (4:0:0)

Sub Code : CS0424
Hrs/week : 04
See Hrs : 3 Hrs

CIE : 50%
SEE : 50%
Max Marks: 100

Course Outcome
On successful completion of the course the students will be able to

1. Provide the information about management as science, art, administration and as a key approach in doing a assigned work.

2. Explain planning and organizing, with planning students learn its importance and hierarchy and with organizing students learn about departmentation, committees, centralization and decentralization.

3. Explain the importance of staffing directing and controlling, meaning and steps in controlling.

4. Identifying who is an entrepreneur, his functions and types and identifying based on geography.

5. Provide knowledge of small scale industry and institutional support.

6. Prepare project Selection; Project Report; Need and Significance of Report.

UNIT 1
Management: Introduction: Meaning - nature and characteristics of Management, Scope and functional areas of management - Management as a science, art or profession Management & Administration - Roles of Management, Levels of Management.

SLE: Development of Management Thought - early management approaches - Modern management approaches, case studies on administration and management.

7 Hours
UNIT 2
Planning: Nature, importance and purpose of planning process - Objectives - Types of plans (Meaning only) - Decision making - Importance of planning - steps in planning & planning premises - Hierarchy of plans.
Organizing: Nature and purpose of organization - Principles of organization - Types of organization - Departmentation - Committees - Centralization Vs Decentralization of authority and responsibility - Span of control - MBO and MBE (Meaning only).
SLE: Case studies on planning and organizing.

UNIT 3
Staffing: Nature and importance of Staffing - Process of Selection & Recruitment Directing & Controlling: Meaning and nature of directing - Leadership styles, Motivation Theories, Communication-Meaning and importance - Coordination, meaning and importance and Techniques of Coordination, Meaning and steps in controlling - Essentials of a sound control system - Methods of establishing control.
SLE: Cases studies on staffing, directing & controlling.

UNIT 4
Entrepreneur: Meaning of Entrepreneur; Evolution of the Concept, Functions of an Entrepreneur, Types of Entrepreneur, Entrepreneur - an emerging Class. Concept of Entrepreneurship- Evolution of Entrepreneurship, Development of Entrepreneurship; Stages in entrepreneurial process; Role of entrepreneurs in Economic Development; Entrepreneurship in India; Entrepreneurship - its Barriers.
SLE: Case studies on entrepreneurs and overcoming barriers.
UNIT 5
Small Scale Industry: Definition; Characteristics; Need and rationale: Objectives; Scope; role of SSI in Economic Development. Advantages of SSI Steps to start an SSI - Government policy towards SSI; Different Policies of S.S.I.; Government Support for S.S.I. during 5 year plans, Impact of Liberalization, Privatization, Globalization on S.S.I., Effect of WTO/GATT Supporting Agencies of Government for S.S.I., Meaning; Nature of Support; Objectives; Functions; Types of Help; Ancillary Industry and Tiny Industry (Definition only).

Institutional Support: Different Schemes; TECKSOK; KIADB; KSSIDC; KSIMC; DIC Single Window Agency; SISI; NSIC; SIDBI; KSFC.

SLE: Case studies on setting up of small scale industries and institutional support.

10 Hours

UNIT 6
Preparation Of Project: Meaning of Project; Project Identification; Project Selection; Project Report; Need and Significance of Report; Contents; formulation; Guidelines by Planning Commission for Project report; Network Analysis; Errors of Project Report; Project Appraisal. Identification of Business Opportunities: Market, Feasibility Study; Technical Feasibility Study; Financial Feasibility Study & Social Feasibility Study.

SLE: Case study on preparation of the project report and feasibility study.

8 Hours

Text Books
1. Entrepreneurship development text and cases, B Janakiram, M Rizwana.

Reference Books
5. Web Sites for the Institutions listed in the Unit 5 on Institutional Support.
PROJECT PHASE -I (0:0:4) (2 Credits)

Sub Code : CS0205  Hrs/week : 04

Course outcome
On successful completion of the course the students will be able to:

1. Identify a real world engineering problem and formulate it.
2. Outline a software project plan to check the feasibility of the solution in terms of both time and cost and carry out Analysis.
3. Evaluate the available tools by Literature survey and adapt it to develop a suitable design
SEMIRAN (0:0:2) (1 credits)

Sub code: CS0110

Course outcome

On successful completion of the course the students will be able to

1. Identify current trends in a specific area of interest.
2. Identify real world issues by conducting literature survey of the area.
3. Understand and interpret the results of technical work as indicated by the literature.
4. Demonstrate both technical report writing and presentation skills.
5. Summarize effectively, so as to improve both oral and written skills.
PROJECT PHASE-II (0:0:12) (6 credits)

Sub code: CS0601

Sub code: CS0601

Hrs/week: 12

Course outcomes

On successful completion of the course the students will be able to:

1. Implement the proposed design of phase-I.

2. Compute the results obtained from the implementation.

3. Verify and validate the obtained results using various test cases.

4. Demonstrate and present the project.

5. Prepare a detailed technical report of the project work carried out.

6. Publish the work in a reputed journal/conference.
Paper Presentation in Journals and Conference
(0:0:2)(1 credits)

Sub code: CS0113

Hrs/week: 2

Course outcomes

On successful completion of the course the students will be able to:

1. Demonstrate the working of new model/method/algorithm in conference.
2. Produce the technical work and results through journals.
ELECTIVES

EMBEDDED SYSTEMS (3:0:2)

Sub code : CS0428  
Hrs / week : 05  
SEE Hrs : 03 Hours  

CIE : 50 % Marks  
SEE : 50 %Marks  
Max. Marks : 100

Course Outcome

On successful completion of the course the students will be able to

1. Describe the fundamentals of Embedded Systems.
2. Develop necessary skills to understand and design an embedded system application.
3. Identify the challenges of Concurrent Process and its solutions.
4. Compare the software Architectures and design an embedded system application using an RTOS.
5. Relate the tasks with inter task communication primitives like, semaphores and message queues and design the hardware along with the choice of the RTOS.
6. Describe the Embedded Software Development Tools and Model the requirements of an application as a set of tasks.

UNIT 1

Custom single-purpose processor: Hardware; Introduction, Combinational Logic, Sequential Logic, Custom Single-Purpose Processor Design, RT-level custom single-purpose processor design; Optimizing custom single-purpose processors: optimizing the FSM, Optimizing the data-path, optimizing the FSM.

SLE : Optimizing the Original Program.

9 Hours
UNIT 2
Timers, counters, and watchdog timers.
State machine models: Introduction; finite-state machines (FSM); Finite-state machines with data path model (FSMD); Using state machines: Describing a system as a state machine, Comparing state machine and sequential program models, Capturing a state machine model in a sequential programming language; Hierarchical/Concurrent state machine model (HCFSM) and the State charts language; Program state machine model (PSM); The role of an appropriate model and language.
SLE: An introductory example, A basic state machine model.

UNIT 3
Concurrent process models: Concurrent processes: Process create and terminate, Process suspend and resume, Process join; Communication among processes: Shared memory, Message passing; Synchronization among processes: Condition variables, Monitors.
Interrupts: The Shared Data Problem; Interrupt latency.
SLE: Interrupt Basics.

UNIT 4
Survey of Software Architecture: Round Robin with Interrupts, Function Queue Scheduling Architecture; Real Time Operating System Architecture, Selecting architecture.
Introduction to RTOS: Tasks and Task States, Tasks and Data, Semaphores and shared data.
SLE: Round Robin.

UNIT 5
Operating Systems Services: Message Queues, Mailboxes, and Pipes; Timer Functions; Events; Memory Management, Interrupt Routines in an RTOS environment. Basic Design
SLE: Saving Power.

UNIT 6
Embedded Software Development Tools: Host and Target Machines, Linker/Locator for Embedded Software, getting embedded software into the Target System
An example system: What the program does
SLE: Environment in which the program operates.

TEXT BOOKS
1. Embedded System Design: A Unified Hardware/Software Introduction - Frank Vahid, Tony Givargis, John Wiley & Sons, Inc. 2002 (Articles: 2.4, 2.5, 2.6; 3.8; 4.2)
2. An Embedded Software Primer - David E. Simon: Pearson Education, 1999. (Chapters: 4, 5, 6, 7, 8, 9, 10 and 11)

REFERENCE BOOKS


Internet of Things (3:0:2)

Sub code : CS0454
Hrs/week : 05
SEE Hrs : 03 Hours

CIE : 50 % Marks
SEE : 50 % Marks
Max. Marks : 100

Course Outcome
On successful completion of the course the students will be able to

1. Explain the Design of IoT.
2. Differentiate IoT, M2M and Describe System management using NETCONF-YANG.
3. Describe IoT Platforms Design Methodology.
4. Develop Python Programs for IoT.
5. Explain the Design of Raspberry Pi.
6. Describe the usage of IoT in physical servers and Cloud.

UNIT-1
Introduction to Internet of Things: Introduction, physical design of IoT, logical design of IoT-functional blocks, communicational models, communication APIs, IoT enabling technologies

SLE: IoT levels, deployment templates 8 Hours

UNIT-2
IoT and M2M:
Introduction, M2M, Difference between IoT and M2M, SDN and NFV for IoT.

IoT System Management with NETCONF-YANG:
Need for IoT Systems Management, Simple Network Management Protocol, Network operator Requirements, NETCONF, YANG
UNIT-3
IoT Platforms Design Methodology:
Introduction, IoT Design Methodology, Motivation for Python
SLE: IoT system for Weather Monitoring
9 Hours

UNIT-4
Hardware and Software for IoT: Logical design using Python—data types & data Structures, control flow, functions, modules, packages, file handling, classes, Python packages of Interest for IoT
SLE: Date/Time Operations
9 Hours

UNIT-5
IoT Physical Devices & Endpoints:
IoT Device, Exemplary Device Raspberry Pi, Board, Linux on Raspberry Pi, Raspberry Pi Interfaces, Programming Raspberry Pi with python.
SLE: Other IoT Devices
8 Hours

UNIT-6
IoT Physical Servers & Cloud Offerings:
Introduction to Cloud Storage Models & Communication APIs, WAMP-AutoBahn for IoT, Xively Cloud for IoT, Python Web Application Framework—Django
SLE: IoT Printer
9 Hours
TEXT BOOK


REFERENCE BOOKS


2. The Internet of Things: Key Applications and Protocols, Olivier Hersent, David Boswarthick, Omar Elloumi, 2012.
WIRELESS AND MOBILE NETWORKS (4:0:0)

Sub code : CS0451
Hrs/week : 04
SEE Hrs : 3 Hours
CIE : 50%
SEE : 50%
Max Marks : 100

Course Outcome: On successful completion of the course the students will be able to
1. Describe an advanced element of learning in the field of wireless communication.
2. Outline the basic concepts of wireless connectivity and mobile computing.
4. Have a working understanding of the characteristics and limitations of mobile hardware devices including their user-interface modalities.
5. Gain knowledge about different mobile platforms and application development.
6. Understand and analyse how Internet Protocols are used for mobile devices and how they are applied differently compared to static devices.

UNIT 1
SLE: Comparison of Common Wireless Communications Systems, VoLTE, usage in India and its difference with normal voice calls.

8 Hours
UNIT 2
Mobility and Handoff management: Handoff, Mobility management, Roaming management, Strategies for handoff detection, MAHO NCHO, MCHO, channel assignment, radio link transfer, hard handoff and soft handoff
SLE: Ongoing call termination vs New call blocking – precedence during handoff and implementation issues involved

10 Hours

UNIT 3:
Number portability and CDPD: Mobile Computing fundamental challenges, Mobile Number Portability-types, architecture and implementation, fixed networks and mobile networks, Signaling relay approaches, SRF and Cellular Digital Packet Data CDPD
SLE: Mobile number portability in India, VoIP service for mobile networks and H.323

8 Hours

UNIT 4
GSM and GPRS: GSM architecture, Location tracking and call setup, security, GPRS Services and architecture, classes of GPRS devices, GPRS state model and radio resource and multiple access management, sharing of radio channels between voice and data, USSD
SLE: Phone theft security and its relation with GSM network

10 Hours

UNIT 5
Mobile OS: Mobile Devices – Special Constrains & Requirements – Commercial Mobile Operating Systems, Palm OS, Win CE and Android OS architectures, Dalvik Virtual Machine working and its differences with JVM, SDKs
SLE: Wireless local loop (WLL) and challenges to Mobile Computing

8 Hours
INTRODUCTION TO DATA MINING (2:0:0)

Sub code : CS0201
Hours/week : 02
SEE Hours : 02 Hours
CIE : 50% Marks
SEE : 50% Marks
Max. Marks: 50

Course Outcome
On successful completion of the course the students will be able to
1. Understand Data Mining concepts and applications of Data Mining Applications.
2. Understand Data Warehouse Implementation.
3. Discuss Data Preprocessing Techniques.

UNIT 1
Introduction to Data Mining: Motivation and importance, What is Data Mining, Relational Databases, Data Warehouses, Transactional Databases, Advanced Database Systems and Advanced Database Applications, Data Mining Functionalities, Interestingness of a pattern Classification of Data Mining Systems.
SLE: Major issues in Data Mining.
9 Hours

UNIT 2
Data Warehouse and OLAP Technology for Data Mining: What is a Data Warehouse? Multi-Dimensional Data Model, Data Warehouse Architecture, Data Warehouse Implementation, Development of Data Cube Technology.
SLE: Data Warehousing to Data Mining
9 Hours

UNIT 3
Data Preprocessing: Why Pre-process the Data? Data Cleaning, Data Integration and Transformation Data Reduction, Discretization.
SLE: Concept Hierarchy Generation

8 Hours

TEXTBOOK
1. Data Mining Concepts and Techniques, Jiawei Han and Micheline Kamber, Morgan Kaufman Publications

REFERENCE BOOKS
1. Introduction to Data Mining, Adriaan, Addison Wesley Publication
2. Data Mining Techniques, A.K.Pujari, University Press