<table>
<thead>
<tr>
<th>Sl. No</th>
<th>Name of the Faculty</th>
<th>Qualification</th>
<th>Designation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Dr. Niranjan.N. Chiplunkar</td>
<td>Ph.D.</td>
<td>Professor/Principal</td>
</tr>
<tr>
<td>2.</td>
<td>Dr. Sarojadevi H.</td>
<td>Ph.D.</td>
<td>Professor and HOD</td>
</tr>
<tr>
<td>3.</td>
<td>Dr. K.R. Uday Kumar Reddy</td>
<td>Ph.D.</td>
<td>Professor</td>
</tr>
<tr>
<td>4.</td>
<td>Dr. Sreekantha D. K.</td>
<td>Ph.D.</td>
<td>Professor</td>
</tr>
<tr>
<td>5.</td>
<td>Dr. Neelima B.</td>
<td>Ph.D.</td>
<td>Professor</td>
</tr>
<tr>
<td>6.</td>
<td>Mrs. Sharada U. Shenoy</td>
<td>M.Tech(Ph.D.)</td>
<td>Associate Professor</td>
</tr>
<tr>
<td>7.</td>
<td>Mr. Venugopala P.S.</td>
<td>M.Tech(Ph.D.)</td>
<td>Associate Professor</td>
</tr>
<tr>
<td>8.</td>
<td>Mr. Roshan Fernandes</td>
<td>M.Tech(Ph.D.)</td>
<td>Associate Professor</td>
</tr>
<tr>
<td>9.</td>
<td>Ms. Shalini P.R.</td>
<td>M.Tech(Ph.D.)</td>
<td>Associate Professor</td>
</tr>
<tr>
<td>10.</td>
<td>Mr. Radhakrishna Dodmane</td>
<td>M.Tech(Ph.D.)</td>
<td>Associate Professor</td>
</tr>
<tr>
<td>11.</td>
<td>Mr. Raju K.</td>
<td>M.Tech(Ph.D.)</td>
<td>Associate Professor</td>
</tr>
<tr>
<td>12.</td>
<td>Mr. Pradeep Kanchan</td>
<td>M.Tech(Ph.D.)</td>
<td>Asst.Prof. Gd III</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fulltime - Sponsored</td>
<td></td>
</tr>
<tr>
<td>14.</td>
<td>Mr. Ravi B.</td>
<td>M.Tech(Ph.D.)</td>
<td>Asst.Prof. Gd II</td>
</tr>
<tr>
<td>15.</td>
<td>Mr. Vijaya Murari T.</td>
<td>M.Tech</td>
<td>Asst.Prof. Gd III</td>
</tr>
<tr>
<td>16.</td>
<td>Mr. Chandra Naik</td>
<td>M.Tech</td>
<td>Asst.Prof. Gd II</td>
</tr>
<tr>
<td>17.</td>
<td>Mr. Manjunath Kamath</td>
<td>M.Tech</td>
<td>Asst.Prof. Gd II</td>
</tr>
<tr>
<td>18.</td>
<td>Mrs. Pallavi K. N.</td>
<td>M.Tech</td>
<td>Asst.Prof. Gd II</td>
</tr>
<tr>
<td>19.</td>
<td>Mr. Hemanth Kumar G</td>
<td>M.Tech</td>
<td>Asst.Prof. Gd II</td>
</tr>
<tr>
<td>20.</td>
<td>Mr. Pradeep Nazareth</td>
<td>M.Tech</td>
<td>Asst.Prof. Gd I</td>
</tr>
<tr>
<td>No.</td>
<td>Name</td>
<td>Degree</td>
<td>Position</td>
</tr>
<tr>
<td>-----</td>
<td>-----------------------------</td>
<td>-------</td>
<td>-------------</td>
</tr>
<tr>
<td>21</td>
<td>Mr. Ranjan Kumar H. S.</td>
<td>M.Tech</td>
<td>Asst.Prof. Gd II</td>
</tr>
<tr>
<td>22</td>
<td>Mrs. Anisha P. Rodrigues</td>
<td>M.Tech</td>
<td>Asst.Prof. Gd II</td>
</tr>
<tr>
<td>23</td>
<td>Mrs. Minu P. Abraham</td>
<td>M.Tech</td>
<td>Asst.Prof. Gd II</td>
</tr>
<tr>
<td>24</td>
<td>Mr. Ramesha Shettigar</td>
<td>M.Tech</td>
<td>Asst.Prof. Gd II</td>
</tr>
<tr>
<td>25</td>
<td>Ms. Savitha Shetty</td>
<td>M.Tech</td>
<td>Asst.Prof. Gd I</td>
</tr>
<tr>
<td>26</td>
<td>Ms. Sharmila S. Sequeira</td>
<td>M.Tech</td>
<td>Asst.Prof. Gd I</td>
</tr>
<tr>
<td>27</td>
<td>Mr. Sannidhan M.S.</td>
<td>M.Tech</td>
<td>Asst.Prof. Gd I</td>
</tr>
<tr>
<td>28</td>
<td>Mr. Naveen Chandaverkar</td>
<td>M.Tech</td>
<td>Asst.Prof. Gd I</td>
</tr>
<tr>
<td>29</td>
<td>Mr. Pawan Hegde</td>
<td>M.Tech</td>
<td>Asst.Prof. Gd I</td>
</tr>
<tr>
<td>30</td>
<td>Mrs. Keerthana B. C.</td>
<td>M.Tech</td>
<td>Asst.Prof. Gd I</td>
</tr>
<tr>
<td>31</td>
<td>Mrs. Asmita Poojari</td>
<td>M.Tech</td>
<td>Asst.Prof. Gd I</td>
</tr>
<tr>
<td>32</td>
<td>Mr. Raghunandan K. R.</td>
<td>M.Tech</td>
<td>Asst.Prof. Gd II</td>
</tr>
<tr>
<td>33</td>
<td>Mrs. Shabari Shedthi B.</td>
<td>M.Tech</td>
<td>Asst.Prof. Gd I</td>
</tr>
<tr>
<td>34</td>
<td>Mr. H R Manjunath Prasad</td>
<td>M.Tech(Ph.D.)</td>
<td>Asst.Prof. Gd I</td>
</tr>
<tr>
<td>35</td>
<td>Mr. Krishna Prasad Rao</td>
<td>M.Tech</td>
<td>Asst.Prof. Gd I</td>
</tr>
<tr>
<td>36</td>
<td>Mr. Sunil Kumar Aithal</td>
<td>M.Tech</td>
<td>Asst.Prof. Gd I</td>
</tr>
<tr>
<td>37</td>
<td>Mr. Shashank Shetty</td>
<td>M.Tech</td>
<td>Asst.Prof. Gd I</td>
</tr>
<tr>
<td>38</td>
<td>Mr. Sampath Kini</td>
<td>M.Tech</td>
<td>Asst.Prof. Gd I</td>
</tr>
<tr>
<td>39</td>
<td>Mrs. Divya Jennifer D’Souza</td>
<td>M.Tech</td>
<td>Asst.Prof. Gd I</td>
</tr>
<tr>
<td>40</td>
<td>Mr. Mahesh Kini</td>
<td>M.Tech</td>
<td>Asst.Prof. Gd I</td>
</tr>
</tbody>
</table>
DEPARTMENT: COMPUTER SCIENCE & ENGINEERING

VISION:
To become a hub of academic activities of the Computer Science and Engineering and a center of excellence in the field of computer science education.

MISSION:
To transform the students into computer science graduates who would be ready to take up any challenges in the field of computer science as well informed, attuned, adapted and responsible by imparting the state of the art concepts and technologies.

Programme Educational Objectives (PEOs):
1. Graduates must gain the ability to identify, formulate & solve challenging Computer science and Engg. Problems both theoretically and practically.
2. Graduates must develop professional and communication skills that prepare them for immediate employment or for adapting to emerging trends by engaging in life-long learning in Computer science and related disciplines.
3. Graduates be provided with an educational foundation that prepares them for leadership roles along diverse career paths and work in a team.
4. Graduates must develop an understanding of the social and human context in which their engineering contribution will be utilized.

**Programme Outcomes (POs):**

After successful completion of the BE programme in Computer Science and Engineering discipline, the students are expected to have attained the following abilities.

a. An ability to apply knowledge of mathematics, science and engineering.
b. An ability to design and conduct experiments, as well as to analyze and interpret data.
c. An ability to design a system, component or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability and sustainability.
d. An ability to function on multidisciplinary teams.
e. An ability to identify, formulate and solve engineering problems.
f. An understanding of professional and ethical responsibility.
g. An ability to communicate effectively.
h. The broad education necessary to understand the impact of engineering solutions in global, economic environmental and societal context.
i. A recognition of the need for and an ability to engage in lifelong learning.
j. An ability to apply engineering and project management principles.
k. An ability to use the techniques, skills and modern engineering tools necessary for engineering practice.
l. An ability to understand and apply the concepts of programming and computer design & technology.
## DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
### SCHEME OF TEACHING AND EXAMINATION

#### V Semester

<table>
<thead>
<tr>
<th>Sl. No.</th>
<th>Code</th>
<th>Subject</th>
<th>Theory/Tuto./Prac./ Self Study</th>
<th>Total Hrs./Week</th>
<th>C.I.E.</th>
<th>S.E.E.</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>13CS501</td>
<td>Compiler Design</td>
<td>4+0+2+0</td>
<td>6</td>
<td>50+50</td>
<td>50+50</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>13CS502</td>
<td>Relational Database Management System</td>
<td>4+0+2+0</td>
<td>6</td>
<td>50+50</td>
<td>50+50</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>13CS503</td>
<td>Microprocessors &amp; Peripherals</td>
<td>4+2+2+0</td>
<td>8</td>
<td>50+50</td>
<td>50+50</td>
<td>5</td>
</tr>
<tr>
<td>4</td>
<td>13CS504</td>
<td>Software Engineering.</td>
<td>4+0+0+0</td>
<td>4</td>
<td>50</td>
<td>50</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>13CS505</td>
<td>Operating Systems</td>
<td>4+0+0+0</td>
<td>4</td>
<td>50</td>
<td>50</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>13CS51Y</td>
<td>Elective I</td>
<td>3+2+0+0</td>
<td>5</td>
<td>50</td>
<td>50</td>
<td>3</td>
</tr>
<tr>
<td>7</td>
<td>13IL001</td>
<td>Employability Skill Development</td>
<td>0+1+0+0</td>
<td>1</td>
<td>50</td>
<td>50</td>
<td>0</td>
</tr>
</tbody>
</table>

**TOTAL** | **34** | **34** | **500** | **450** | **26** |
## Elective – I

<table>
<thead>
<tr>
<th>Code</th>
<th>Course Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>13CS511</td>
<td>Advanced Unix Programming</td>
</tr>
<tr>
<td>13CS512</td>
<td>Signals &amp; Systems Theory</td>
</tr>
<tr>
<td>13CS513</td>
<td>Digital Signal Processing</td>
</tr>
<tr>
<td>13CS514</td>
<td>Data Compression</td>
</tr>
<tr>
<td>13CS515</td>
<td>Operations Research</td>
</tr>
<tr>
<td>13CS516</td>
<td>Cloud Computing Architecture (CCV)</td>
</tr>
<tr>
<td>13CS517</td>
<td>Data Mining and Predictive Modeling (BAO)</td>
</tr>
</tbody>
</table>
## DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

### SCHEME OF TEACHING AND EXAMINATION

<table>
<thead>
<tr>
<th>Sl. No.</th>
<th>Code</th>
<th>Subject</th>
<th>Theory/Tuto./Prac./ Self Study</th>
<th>Total Hrs./Week</th>
<th>C.I.E.</th>
<th>S.E.E.</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>13CS601</td>
<td>Computer Graphics &amp; Multimedia</td>
<td>4+0+2+0</td>
<td>6</td>
<td>50+50</td>
<td>50+50</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>13CS602</td>
<td>Computer Networks</td>
<td>4+0+2+0</td>
<td>6</td>
<td>50+50</td>
<td>50+50</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>13CS603</td>
<td>Java, &amp; Internet Technologies</td>
<td>4+0+2+0</td>
<td>6</td>
<td>50+50</td>
<td>50+50</td>
<td>5</td>
</tr>
<tr>
<td>4</td>
<td>13CS604</td>
<td>Software Testing and Automation</td>
<td>3+0+0+0</td>
<td>3</td>
<td>50</td>
<td>50</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>13CS61Y</td>
<td>Elective-II</td>
<td>3+1+0+0</td>
<td>4</td>
<td>50</td>
<td>50</td>
<td>3</td>
</tr>
<tr>
<td>6</td>
<td>13CS62Y</td>
<td>Elective-III</td>
<td>2+1+1+0</td>
<td>4</td>
<td>50+50</td>
<td>50+50</td>
<td>3</td>
</tr>
<tr>
<td>7</td>
<td>13CS605</td>
<td>IGW*</td>
<td>0+3*+0+0</td>
<td>3</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>8</td>
<td>13IL002</td>
<td>Employability Skill Development</td>
<td>0+1+0+0</td>
<td>1</td>
<td>50</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td></td>
<td></td>
<td></td>
<td><strong>33</strong></td>
<td><strong>550</strong></td>
<td><strong>500</strong></td>
<td><strong>24</strong></td>
</tr>
<tr>
<td>Elective –II</td>
<td>Elective-III</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>--------------------------------------</td>
<td>---------------------------------------</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13CS611 Pattern Recognition</td>
<td>13CS621 VLSI Design</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13CS613 Distributed Systems</td>
<td>13CS623 Microcontrollers</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13CS614 Advanced DBMS</td>
<td>13CS623 Embedded and Real Time Systems</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13CS615 Cloud Computing and Infrastructure Management</td>
<td>13CS624 Programming Languages</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13CS616 Big Data</td>
<td>13CS625 Programming Languages</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13CS617 BPM (Business Process Management (CCV))</td>
<td>13CS626 Multicast Communications</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13CS618 Data Warehouse &amp; Multi- dimensional Modeling (BAO)</td>
<td>13CS627 Cloud Deployment Models (CCV)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13CS619 Information and Storage Management</td>
<td>13CS628 Business Intelligence [Cognos BI] (BAO)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

* These classes will be held only during the first week of the semester.

Note: Where ever there is a combined theory and lab, students must score minimum passing marks in each of the component.
COMPILER DESIGN

Sub Code : 14CS501                   Credits : 04
Hrs/Week : 4+0+2+0                   Total Hours : 52

Course Outcomes:
1. Understand different translators, the role of lexical analyzer in the first phase of compiler, working of loader, linkers and their design, the role of syntax analyzer in a compiler, use of Syntax-Directed Translation in parsing and code generation, the use of Basic blocks and Flow graphs in code generation, the use of Syntax tree and DAG in code generation and the working of simple code generator
2. Learn different top down and bottom up parsers and applications of Context free grammars, DFA and NFA
3. Construct LR parsing tables
4. Write lexical analyzer using transition diagrams, write YACC programs, Syntax-Directed Translation / Definition for a given problem and generate intermediate code expressions, different language constructs and simple code generator for a subset of C language

UNIT-I


Lexical Analysis:
Lexical Analysis- The Role of Lexical Analyzer, Input Buffering, Specifications of Tokens, Recognition of Tokens, A Language for Specifying Lexical Analyzer, LEX programming. 11 Hours

UNIT-II
Syntax Analysis:

The Role of the Parser, Context-free Grammars, Top-down Parsing, Bottom-up Parsing, Operator-Precedence Parsing. 11 Hours

UNIT-III

Syntax Analysis:
LR Parsers –Simple LR (SLR), Canonical LR (CLR), Look Ahead LR (LALR) Parsers, YACC programming. 11 Hours

UNIT-IV

Syntax-Directed Translation:
Syntax-Directed definitions, Constructions of Syntax Trees, Bottom-up Evaluation of S-attributed definitions, L-attributed definitions.

Intermediate Code Generation:
Intermediate Languages, Declarations, Assignments, Boolean Expressions. 11 Hours

UNIT-V

Code Generation:

Issues in the design of Code Generator, The Target Machine, Run-time Storage Management, Basic blocks and Flow graphs, Next-use information, A Simple Code Generator, Register Allocation and Assignment, The DAG representation of Basic Blocks.

Code Optimization:
Introduction, The Principle of Optimization, Optimization of Basic Blocks, Loops in flow graphs. 12 Hours

TEXT BOOKS:
2. Leland L Beck, System Software

REFERENCE BOOKS:

3. Internet resources for advanced compilation
4. Language Processing Tools by John R Levine, O’Reilly Publication

COMPILER DESIGN LAB

<table>
<thead>
<tr>
<th>Sub Code</th>
<th>Credits</th>
<th>Hrs/Week</th>
<th>Total Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>14CS501</td>
<td>01</td>
<td>02</td>
<td>52</td>
</tr>
</tbody>
</table>

Course Outcomes:

1. Write LEX programs
2. Develop YACC programs
3. Design and Implement Mini Project

1. Students have to carry out experiments based on LEX and YACC
2. The following mini projects need to be carried out by the students in groups
   a. Implementing the SIC assembler
   b. Writing a mini compiler for a hypothetical language (e.g. a subset of C), generating the 8086 compatible assembly code. This code may be assembled and executed using MASM and debug to check for the correctness.
RELATIONAL DATABASE MANAGEMENT SYSTEM

Sub Code : 14CS502 Credits : 04
Hrs/Week : 4+0+2+0 Total Hours : 52

Course Outcomes:
1. Understand database structures and how they work
2. Design simple database models using Entity-Relationship Modeling
3. Write queries using SQL
4. Construct the stages of database project design
5. Understand the issues associated with Transaction Processing and Recovery

UNIT-I
Introduction to database systems
Introduction, Characteristics of the Database approach, Actors on the scene, Advantages of using the DBMS approach, Data models, Schemes and Instances, Three Schema Architecture and Data Independence.

Entity-Relationship Model
Using High-Level Conceptual Data Models for Database Design; An Example Database Application; Entity Types, Entity Sets, Attributes and Keys; Relationship types, Relationship Sets, Roles and Structural Constraints; Weak Entity Types; Refining the ER Design for the COMPANY Database; ER Diagrams, Naming Conventions and Design Issues.

UNIT-II

Relational Model And Relational Algebra
Relational Model Concepts; Relational Model Constraints and Relational Database Schemas; Update Operations and Dealing with Constraint Violations; Unary Relational Operations: SELECT and PROJECT; Relational Algebra Operations from Set Theory; Binary Relational Operations: JOIN and DIVISION; Additional Relational Operations; Examples of Queries in Relational Algebra; Relational Database Design Using ER- to-Relational Mapping.  

**UNIT-III**

**SQL-The Relational Database Standard**
SQL Data Definition and Data Types, Specifying Basic Constraints in SQL, Schema Change Statements in SQL; Basic Queries in SQL; More Complex SQL Queries; Insert, Delete and Update Statements in SQL; Additional Features of SQL; Views (Virtual Tables) in SQL; Database Programming: Issues and Techniques; Embedded SQL.  

10 Hours

**UNIT-IV**

**Database Design**
Informal Design Guidelines for Relation Schemas; Functional Dependencies; Normal Forms Based on Primary Keys; General Definitions of Second and Third Normal Forms; Boyce-Codd Normal Form; Properties of Relational Decompositions; Algorithms for Relational Database Schema Design; Multivalued Dependencies and Fourth Normal Form; Join Dependencies and Fifth Normal Form.  

10 Hours

**UNIT-V**

**Transaction Management**
The ACID Properties; Transactions and Schedules; Concurrent Execution of Transactions; Lock- Based Concurrency Control; Performance of Locking; Transaction Support in SQL; Introduction to Crash Recovery; 2PL, Serializability and Recoverability; Introduction to Lock Management; Lock Conversions; Dealing with Deadlocks; Specialized Locking Techniques; Concurrency Control without Locking; Introduction

11 Hours

TEXT BOOKS:


REFERENCE BOOK:


************

RDBMS LABORATORY

Sub Code: 13CS502  Credits: 01  
Hours / Week: 02

Course Outcomes:

1. Understand various queries and their execution  
2. Design a new database  
3. Do simple programming using visual basic  
4. Design GUI based database applications using Visual Basic

Student must carry out experiments using some RDBMS package (like MS-SQL/Oracle/DB2..) at the back end and some visual programming tool at the front end (like VB/VC++/D2K..). Typical experiments must include query processing applications. All SQL statements must be made use of by the student in developing the application.

*************

MICROPROCESSORS AND PERIPHERALS

Sub Code: 13CS503
Hours/Week: 4+2+2+0
Credits: 04
Total Hours: 52

Course Outcomes:
1. Understand the architecture of 8086 and addressing modes
2. Write assembly level language programs using the instruction set and assembler directives
3. Develop programs that solve complex arithmetic problems using the arithmetic coprocessor
4. Contrast the 80386/80486 microprocessors with earlier Intel microprocessors
5. Interface keypad, 7-segment displays, stepper motor, logic controller and elevator to the microprocessor

UNIT-I
Microprocessor and 8086 Architecture.
8086 internal architecture
8086 Instruction Descriptions and Assembler Directives
8086 Addressing modes and Instruction Formats

8086 / 8088 Hardware Specifications.
Pin Functions
Basic 8086 Configurations: (Min. and Max. Modes)

8086 Instruction Set

UNIT-II
Programming the Microprocessor
Programming based on instruction set, Modular Programming, Using the Key Board and Video Display, DOS & BIOS interrupts, Data Conversions, Disk files, Example Programs. 

**Memory interface** to 16bit and 32 bit & 64 bit Microprocessors, examples  

**10 Hours**

**UNIT-III**

**Arithmetic Co Processor**

Data formats for the arithmetic co processor, 80 x 87 architecture, Instruction set.  

**Interrupt Controller and DMA Controller**

8259 A Programmable Interrupt Controller, Basic DMA Operation, 8237 DMA Controller  

**10 Hours**

**UNIT-IV**

**Introduction to Advanced Microprocessors**

Salient features of 80186,80286,80386,80486 and Pentium Family processors up to P-IV, special registers, MMX and SSE, Overview of Protected Mode and Address translation.  

**11 Hours**

**UNIT - V**

**Interfacing**

Programmable Parallel Ports and Handshake Input/Output, Interfacing a Microprocessor to Keyboards,7-segment Displays, stepper motor, Logic controller, Elevator, DAC interface for waveform generation, ADC interface.  

**11 Hours**

**TEXT BOOKS:**

1. Microprocessors- Douglas V.Hall, Revised 2nd Edition  
5. 8051 Microcontroller- Kenneth Ayala

****************

MICROPROCESSOR LABORATORY

Sub Code: 13CS503
Credits: 01
Hrs / Week: 02

Course Outcomes:
1. Write Assembly Language programs to solve various problems.
2. Develop programs to interface Intel 8086 microprocessor to external devices like keypad, seven segment display, elevator, stepper motor, ADC, Logic Controller

Student must use MASM/TASM, appropriate linker and debug utility to carry out 8086 assembly language program experiments. Typical programs include
1. Searching
2. Sorting
3. String manipulation
4. usage of Macros and subroutines
5. DOS interrupt usage
6. BIOS interrupt usage
7. Keyboard interface
8. Display interface
9. Logic controller interface
10. DAC interface
11. Stepper Motor control interface
12. Elevator interface
13. Some examples of Real Mode programming in Pentium.
14. Student must be encouraged to do some chip level interfacing with 8086.

SOFTWARE ENGINEERING

Sub Code: 13CS504  Credits : 04
Hrs/Week: 4+0+0+0  Total Hours: 52

Course Outcomes:

1. Understand Basics of Software engineering.
2. Understand the various software production techniques (process models).
3. Understand the project management concepts
4. Analyze the difference between software evolution, software costing and managing people.

UNIT-I
Overview

Requirements
Software Requirements: Functional and Non-functional requirements; User requirements; System requirements; Interface specification; The software requirements document. Requirements Engineering Processes: Feasibility studies; Requirements elicitation and analysis; Requirements validation; Requirements management. 10 Hours
UNIT-II

System models, Project Management
System Models: Context models; Behavioral models; Data models; Object models; Structured methods.
Project Management: Management activities; Project planning; Project scheduling; Risk management. 10 Hours

UNIT-III

Software Design
Architectural Design: Architectural design decisions; System organization; Modular decomposition styles; Control styles.
Object-Oriented design: Objects and Object Classes; An Object-Oriented design process; Design evolution. 11 Hours

UNIT-IV

Development
Rapid Software Development: Agile methods; Extreme programming; Rapid application development.
Software Evolution: Program evolution dynamics; Software maintenance; Evolution processes; Legacy system evolution.

Verification and Validation
Verification and Validation: Planning; Software inspections; Automated static analysis; Verification and formal methods.
Software testing: System testing; Component testing; Test case design; Test automation. 11 Hours

UNIT-V

Management
Managing People: Selecting staff; Motivating people; Managing people; The People Capability Maturity Model.
Software Cost Estimation: Productivity; Estimation techniques; Algorithmic cost modeling, Project duration and staffing. 10 Hours
TEXT BOOKS:

REFERENCE BOOKS:

*********************************************************

OPERATING SYSTEMS

Sub Code: 13CS505  Credits : 04
Hrs/Week: 4+0+0+S*  Total Hours: 52

*Self Study to be exercised under the supervision of course instructor and to be restricted to not more than 10% of the total teaching hours.

Course Outcomes:

1. Understand Basics of Operating Systems and OS Service
2. Know about the Process Scheduling and understanding of Process Synchronization methods
3. Understand deadlocks and management of it
4. To study main, virtual and Secondary memory management aspects are studied and various file system implementations

UNIT - I
**Syllabus of V & VI Semester B.E./Computer Science & Engg.**

**Introduction to Operating Systems, System structures:** Operating System structure; Operating System operations(functions), Operating System Services; User - Operating System interface; System calls; Types of system calls; System programs; Operating System design and implementation; Operating System structure; Virtual machines; Operating System generation; System boot.

**Process Management:** Process concept; Process scheduling; Operations on processes; Inter-process communication.

**Multi-Threaded Programming:** Overview; Multithreading models; Thread Libraries; threading issues. Process - thread comparison.

10 Hours

**UNIT - II**

**Process Scheduling:** Basic concepts; Scheduling criteria; Scheduling algorithms; Multiple-Processor scheduling; thread scheduling.

**Process Synchronization**
Synchronization: The Critical section problem; Peterson’s solution; Synchronization hardware; Semaphores; Classical problems of synchronization; Monitors.

**Deadlocks**
Deadlocks: System model; Deadlock characterization; Methods for handling deadlocks; Deadlock prevention; Deadlock avoidance; Deadlock detection and recovery from deadlock.

10 Hours

**UNIT-III**

**Memory Management:** Memory Management Strategies: Background; Swapping; Contiguous memory allocation; Paging; Structure of page table; Segmentation. Virtual Memory Management: Background; Demand paging; Copy-on-write; Page replacement; Allocation of frames; Thrashing.

10 Hours

**UNIT-IV**
**File System, Implementation of File System**

File System: File concept; Access methods; Directory structure; File system mounting; File sharing; Protection.
Implementing File System: File system structure; File system implementation;
Directory implementation; Allocation methods; Free space management.  

11 Hours

---

**UNIT-V**

**Secondary Storage Structures**, Mass storage structures; Disk structure; Disk attachment; Disk scheduling; Disk management; Swap space management.

**Self Study Component: The Linux & Windows Operating System**: history; Design principles; Kernel modules; Process management; Scheduling; Memory management; File systems, Input and output; Inter-process communication - Comparison.

**Tutorials**: Students may be asked to go through the LINUX kernel code to understand the OS design. Students may be asked to implement a few algorithms related to operating system.  

11 Hours

**Note**: Students must submit a report for the studies conducted & programs implemented at the end of semester.

**TEXT BOOKS**:


**REFERENCE BOOKS**:


*******************

ADVANCED UNIX PROGRAMMING
Sub Code: 13CS511 Credits :03
Hrs/Week: 03 Total Hrs: 39

Course Outcomes:
1. Define and discuss the POSIX standard and different types of files.
2. Explain and apply various APIs for file handling.
3. Understand the representation of a process and its environment and interprocess communication; Study and apply various APIs for process handling
4. Understand the concept of signal and its handling methods. Use the signal handling APIs in programs.
5. Understand the concept of demon process.

UNIT-I

The POSIX standards. File types. General File APIs, File handling programs. 7 Hours

UNIT-II

Makefile – introduction, creation and execution of make file.
THE PROCESS: Introduction, Mechanism for creating process. The UNIX Kernel support for process.

THE ENVIRONMENT OF A UNIX PROCESS: Introduction, main function, Process Termination, Command line arguments, Environment List, Memory layout Of a C program, Memory allocation, Environment variables, functions.  

UNIT-III
Setjmp and longjmp functions, getrlimit, setrlimit
SIGNALS: The UNIX Kernel Support for signals, Signal,  

UNIT-IV
Signal mask, Sigaction, The SIGCHLD Signal and waitpid functions, The sigset jmp and sig long jmp Functions, Kill, Alarm, Interval Timers, POSIX .1b Timers.
DAEMON PROCESSES: Introduction, Daemon Characteristics, Coding Rules.  

UNIT-V
INTERPROCESS COMMUNICATIONS: Overview of IPC Methods, Pipes, popen, Pclose functions, FIFOs, Message Queues, Semaphores, Shared Memory.
SOCKETS: Introduction, functions, Client/Server Message Handling Example.  

TEXT BOOKS:


REFERENCE BOOKS:
3. R. Stones, N. Matthew, Beginning Linux Programming, Wrox publication.

************************

SIGNALS AND SYSTEMS THEORY
Sub Code: 13CS512 Credits :03
Hrs/Week: 03 Total Hrs: 39

UNIT-I

Introduction
Definitions of a signal and a system, classification of signals, basic operations on signals, elementary signals, systems viewed as interconnections of operations, properties of systems. 7 Hours

UNIT-II

Time-domain representations for LTI systems
Convolution, impulse response representation, properties of impulse response representation, differential and difference equation representations, block diagram representations. 8 Hours

UNIT-III

Fourier representation for signals
Introduction, Fourier representations for four signal classes, orthogonality of complex sinusoidal signals, DTFS representations, continuous-time-Fourier-series representations, DTFT and FT representations, properties of Fourier representations. 8 Hours

UNIT-IV

Application of Fourier representations
Frequency response of LTI systems, solution of differential and difference equations using system function, Fourier transform representations for periodic signals, sampling of continuous time signals and signal reconstruction. 8 Hours

UNIT-V

Z-Transforms
Introduction, Z-transform, properties of ROC, properties of Z-transforms, inversion of Z-transforms, transforms analysis of LTI systems, transfer function, stability and causality, unilateral Z-transforms and its application to solve difference equations. 8 Hours

TEXT BOOK:

   Chapters : 1.1 to 1.8, 2.2 to 2.5, 3.1 to 3.6, 4.2 to 4.3, 4.7, 7.1 to 7.6, 7.8

REFERENCE BOOKS:


*****************

DIGITAL SIGNAL PROCESSING

Sub Code : 13CS513  
Credits : 03
Hrs/Week: 03  
Total Hrs: 39

UNIT - I

The Discrete Fourier Transform: Its Properties and Applications


8 Hours

UNIT - II

Efficient Computation of the DFT: Fast Fourier Transform Algs

UNIT - III

Implementation of Discrete-Time Systems

Structures for FIR Systems: Direct-Form Structures, Cascade-Form Structures, Frequency-Sampling Structures, Lattice Structure.

UNIT - IV

Representation of Numbers
Fixed-Point Representation of Numbers, Binary Floating-Point Representation of Numbers, Errors Resulting from Rounding and Truncation.
Quantization of Filter Coefficients: Analysis of Sensitivity to Quantization of Filter Coefficients, Quantization of Coefficients in FIR Filters.

**UNIT - V**

**Digital Signal Processors**
Architecture, features and instructions of Fixed and Floating point Processors. (TMS320c25 and TMS32067)  

**TEXT BOOKS :**

**REFERENCE BOOKS:**
6. Digital Signal Processors: by B.Venkataramini & M.Bhaskar

************************

DATA COMPRESSION

Sub Code: 13CS514
Credits: 03
Hrs/Week: 03
Total Hrs: 39

UNIT-I
Introduction, Lossless Compression -1

Compression techniques; Modeling and Coding.
Mathematical preliminaries for lossless compression: Overview; Basic concepts of Information Theory; Models; Coding; Algorithmic information theory; Minimum description length principle.
Huffman coding: Overview; The Huffman coding algorithm, Minimum variance Huffman codes; Application of Huffman coding for text compression. 7 Hours

UNIT-II
Lossless Compression -2
Dictionary Techniques: Overview; Introduction; Static dictionary; Adaptive dictionary; Applications: UNIX compress, GIF, PNG, V.42.
Lossless image compression: Overview; Introduction; Basics; CALIC; JPEG-LS; Multiresolution approaches; Fascimile encoding: Run-length coding, T.4 and T.6.

UNIT-III

Basics of Lossy Coding
Some mathematical concepts: Overview; Introduction; Distortion criteria; Models. Scalar quantization: Overview; Introduction; The quantization problem; Uniform quantizer; Adaptive Quantization.

Vector Quantization, Differential Encoding
Vector quantization: Overview; Introduction; Advantages of vector quantization over scalar quantization; The LBG algorithm. Differential Encoding: Overview; Introduction; The basic algorithm; Prediction in DPCM; Adaptive DPCM.

UNIT-IV

Some mathematical Concepts, Transform coding
Some mathematical concepts: Linear Systems; Sampling; Discrete Fourier transform; Z-transform. Transform coding: Overview; Introduction; The transform; Transforms of interest; Quantization and coding for transform coefficients.

Subband Coding, Audio Coding
Subband coding: Overview; introduction; Filters; The basic subband coding algorithm; Bit allocation. Audio coding: Overview; introduction; MPEG audio coding.

UNIT-V
Wavelet- Based Compression

Overview; Introduction; Wavelets; Multiresolution and the scaling function; Image compression; JPEG 2000.

Video Compression
Overview; Introduction; Motion compensation; Video signal representation; H.261; Model-based coding; Asymmetric applications; MPEG-1 and MPEG-2; H.263; H.264; MPEG-4 and advanced video coding 8 Hours

TEXT BOOK:
   (Chapters1,2 (excluding 2.2.1 and 2.4.3 ), 3.1, 3.2, 3.2.1, 3.8.2, 5, 7.1 to 7.5, 7.6, 7.6.1, 7.6.2, 8.1 to 8.3, 8.6, 9.1 to 9.5, 10.1 to 10.4, 11.1 to 11.5, 12.6 to 12.9, 13.1 to 13.5, 14.1 to 14.4, 14.9, 15.1 to 15.4, 15.6, 15.9, 16.1 to 16.3, 18.1 to 18.12)

REFERENCE BOOK:

******************
OPERATIONS RESEARCH

Sub Code: 13CS515 Credits : 03
Hrs/Week: 3 Total Hrs: 39

Course Outcomes:
1. To know the basics of OR, modelling and applications of OR.
2. Definition of linear programming model, formulation of linear programming model and application of linear programming model using different techniques.
3. To formulate the problem and solve the problem by using different techniques.
4. Describe mathematical formulation of an Assignment Problem and solve various scenarios by using different methods.
5. Understand and identify the project management techniques

UNIT-I

Introduction

Introduction to OR, nature and meaning, applications, modeling in OR, phases of OR study

Linear Programming

Introduction to Linear Programming through an example, graphical method, formulation of LP model from practical problems, assumptions and properties of linear programming, simplex method 7 Hours

UNIT-II

Revised simplex method, Big M method, 2 phase method, Duality theory, Primal and dual relationship, Dual simplex method 8 Hours

UNIT-III

Transportation Problems:

Special types of main programming, transportation problems, methods to find initial feasible solution and modification to obtain optimal solution (Degeneracy in transportation problems, unbalanced transportation problems 8 Hours

UNIT-IV
Assignment problem

Mathematical formulation of an assignment problem, unbalanced assignment problem, TSP, Hungarian method 8 Hours

UNIT-V

CPM, PERT

Representation of a project by a network, activities and events, starting times, finishing times, floats, slacks, CPM, Idea of crashing probabilistic times and PERT analysis 8 Hours

TEXTBOOKS:

REFERENCE BOOKS:
5. Operation research, Kantiswaroop, Manmohan and Gupta

*******************

CLOUD COMPUTING ARCHITECTURE

Sub Code: 13CS516
Hours/Week: 03
Credits : 3
Total Hours: 39

Objectives:
1. To learn cloud computing delivery model IaaS,
2. To learn cloud computing delivery model PaaS,
3. To learn cloud computing delivery model SaaS

UNIT - I

Unit Title : Overview of delivery models in Cloud Computing
Introduction; Overview of Cloud Computing; Cloud Service models and Cloud Deployment Models; Cloud History – Internet technologies (SOA, Web Services, Web 2.0, mashups), Distributed computing – Utility and Grid Computing, Hardware – VMWare ESXi, Xen, KVM; Virtual Appliances and the open Virtualization format; System Management; Anatomy of Cloud; Benefits of Cloud; Cloud Transformation roadmap; cloud delivery models and their advantages; Cloud computing architecture.  

UNIT-II

Infrastructure as a Service (IaaS)
Introduction to Infrastructure as a Service delivery model, characteristics of IaaS, Architecture, examples of IaaS, Applicability of IaaS in the industry, Comparing ISPs and IaaS, Motivations for renting the infrastructure; IaaS Case studies; IaaS enabling Technology; Trusted cloud; 

UNIT-III

Platform as a Service (PaaS)
Introduction to Platform as a Service delivery model, characteristics of PaaS, patterns, architecture and examples of PaaS, Applicability of PaaS in the industry; Integrated Lifecycle Platform; Anchored Lifecycle platform; Enabling Technologies as a Platform; PaaS – best option or not; 

UNIT-IV

Software as a Service (SaaS)
Introduction to Software as a Service delivery model, characteristics of SaaS, SaaS Origin; Evolvement of SaaS – Salseforce.com’s approach; SaaS Economics and Ecosystem; Types of SaaS Platforms; Architecture, SaaS – Providers; Collaboration as a Service; Enabling and Management tools as a Service; Applicability of SaaS in the industry

UNIT-V

Cloud Computing Reference Architecture (CCRA)
Introduction to Cloud computing reference architecture (CCRA), benefits of CCRA, Architecture overview – The conceptual Reference Model; Cloud Consumer; Cloud provider; Cloud Auditor; Cloud carrier; Scope of control between Provider and Consumer; CCRA Architectural Components – Service deployment, Service Orchestration, Cloud Service Management, Security; Cloud Taxonomy; IBM’s Cloud Computing Reference Architecture (CCRA 2.0) – Introduction, roles, Architectural elements; CCRA evolution; Examples of Cloud Services; versions and application of CCRA for developing clouds.

9 Hours

TEXT BOOKS
• Cloud Computing Architecture by IBM ICE Publication

REFERENCES BOOKS
• Cloud Computing For Dummies (November, 2009), Judith Hurwitz, Robin Bloor, Marcia Kaufman, Fern Halper
• IBM Cloud Computing
• Wikipedia page on Cloud Computing
  http://en.wikipedia.org/wiki/Cloud_computing

******************************
DATA MINING AND PREDICTIVE MODELING AND PREDICTIVE

Sub Code: 13CS517  Credits: 3
Hours/Week: 03  Total Hours: 39

Objectives:
1) To learn, how to develop models to predict categorical and continuous outcomes, using such techniques as neural networks, decision trees, logistic regression, support vector machines, and Bayesian network models.
2) To know the use of the binary classifier and numeric predictor nodes to automate model selection.
3) To advice on when and how to use each model. Also learn how to combine two or more models to improve prediction

Expected Outcome:

At the end of this course student would be able to -
1. Understand the process of formulating business objectives, data selection/collection, preparation and process to successfully design, build, evaluate and implement predictive models for a various business applications.
2. Compare and contrast the underlying predictive modeling techniques.
3. Select appropriate predictive modeling approaches to identify particular cases to progress with.
4. Apply predictive modeling approaches using a suitable package such as SPSS Modeler

UNIT - I

Data Understanding & preparation
Identifying business objectives, Translating business objectives to data mining goals, Reading data from various sources – Database/ Excel/ Text/others, data visualization – tabular & graphic, distributions and summary statistics, field reordering, Reclassify data 6 Hours

UNIT - II

Data Transformations
Data quality issues, Data Audit, anomalies, relationships among variables, Extent of Missing Data, Segmentation, Outlier detection, Variable transformations, Variable derivation, Variable selection, Automated Data Preparation, Combining data files, data restructuring, Aggregation, Duplicates removal, Sampling cases, Data Caching, Partitioning data, Missing Value replacement 6 Hours

UNIT-III

Modeling techniques - I
Partitioning The Data - Training, Validation & Testing, Model selection, Model development techniques - Linear regression, Logistic regression, Discriminant analysis, Bayesian networks, Neural networks, Rule Induction 8 Hours

UNIT-IV

Modeling techniques - II
Support vector machines, Cox regression, Time series analysis, Decision trees, Clustering, Association Rules, Sequence Detection, Which Technique To Use When 8 Hours

UNIT-V

Model evaluation & deployment
Model Validation, Determining Model Accuracy, Rule Induction Using CHAID, Automating Models For Categorical Targets, Automating Models For Continuous Targets, Comparing And Combining Models, Evaluation Charts For Model Comparison, Using Propensity Scores, Meta-Level Modeling, Error Modeling, Deploying Model, Exporting Model Results, Assessing Model Performance, Updating A Model 8 Hours
TEXT BOOKS
• Data Mining Predictive Modeling by IBM ICE Publications

REFERENCES BOOKS
• Bruce Ratner, Statistical and Machine-Learning Data Mining, CRC Press, 2011
• Eric Siegel & Thomas H. Davenport, Predictive Analytics, Wiley Publications, 2013
• James Wu and Stephen Coggeshall, Foundations of Predictive Analytics, CRC Press, 2012

Data Mining and Predictive Modeling Lab (Tutorial)

List of Experiments: (Each of the below Exercises has sub Labs)
Exercise 1: Introduction to the Case Study
Exercise 2: Data Understanding and Preparation
Exercise 3: Partitioning Data

TEXT BOOKS
Data Mining and Predictive Modeling by IBM ICE Publication


COMPUTER GRAPHICS AND MULTIMEDIA

Sub Code: 13CS601  Credits :4
Hours/Week: 04  Total Hours: 52

Course Outcomes:
1. Understand basics graphics and graphic devices.
2. Explain and apply raster graphics algorithms.
3. Explain various transforms (2D and 3D).
4. Design the curves fractal and perform shading and perform visible surface determination.
5. Apply Open GL for graphics programming.

UNIT-II

Introduction:
Introduction to graphics, Raster and random scan displays, video controller, Applications of Computer Graphics.


UNIT-II


Geometrical Transformations (3D): Matrix representation of 3D Transformations, Transformations as change in coordinate system.

UNIT-III

Curves, Fractals and Shading: Polygon surfaces, curved lines and surfaces, Quadratic surfaces, Spline Representation, Bezier & B-Spline Curves & Surfaces, Fractal Geometry methods, Illumination models, Shading models for polygons, surface details and shadows.

UNIT-IV


UNIT-V


TEXT BOOKS:

REFERENCE BOOK:

******************************
COMPUTER GRAPHICS & MULTIMEDIA LABORATORY

Sub Code : 13CS601  Credits : 02
Hours/Week : 02

A. Student has to write and execute programs in C/C++ using OPENGL on Windows/Linux platform to implement a few graphics applications like:

1. Transformations in both 2D and 3D
2. Clipping
3. 3D viewing
4. Hidden line removal
5. Fractal generation

B. Student may also be asked to implement one or two graphics algorithms like Line drawing or Circle drawing or Filling by using only graphic primitives

C. Graphics Mini project implementation using Open GL.

***************************

COMPUTER NETWORKS

Sub Code : 13CS602  Credits : 04
Hours/Week : 04  Total Hrs : 52

Course Outcomes:

1. Understand basic concept of computer network.
2. Explain the network layer and the related issues.
3. Explain the congestion control, prevention methods.
4. Explain different type of networks and protocols.
5. Describe the transport layer protocols.
UNIT-I

Introduction to Computer Networks

UNIT-II

Network Layer (Part-I)
Network layer design issues: Store and Forward packet Switching, Services Provided to the Transport Layer, Implementation of Connectionless Service, Implementation of Connection-Oriented Service, Comparison of Virtual Circuit and Datagram Subnets; Routing algorithms: The Optimality Principal, Shortest Path Routing, Flooding, Distance Vector Routing, Link state Routing, Hierarchical Routing, Broadcast Routing, Multicast Routing, Routing for Mobile Hosts, Routing in Ad hoc Networks. 10 Hours

UNIT-III

Network layer (Part – II)
Congestion Control Algorithms: General Principles of Congestion Control, Congestion Prevention Policies, Congestion Control in Virtual-Circuit Subnets, Congestion Control in Datagram Subnets, Load Shedding, Jitter Control; Quality Of Service: Requirements, Techniques for Achieving Good Quality of Service, Integrated Services, Differentiated Services; 10 Hours
UNIT-IV

Network layer (Part – III)


11 Hours

UNIT-V

The Transport Layer


11 Hours

TEXT BOOKS:

REFERENCE BOOKS:


***************

COMPUTER NETWORKS LABORATORY

Sub Code : 13CS602  Credits : 01
Hours/Week : 02

A. Student must carryout socket programming based experiments for file transfer and message transfer using TCP and UDP
B. Use of any one Network Simulator (Like NS2/Etheral) to simulate some networking scenario and performance evaluation. Tcl/Tk programs also may be included
C. Study of Router/Firewall configuration, Use of Network Monitoring tool, study of Proxy settings

***************
JAVA AND INTERNET TECHNOLOGIES

Sub Code : 13CS603 Credits : 04
Hours/Week : 04 Total Hrs : 52

Course Outcomes:
1. Understand the basic concepts of Java programming.
2. Apply inheritance, package and exception handling in Java programming.
3. Explain the functions for event handling and GUI programming in Java.
4. Apply AWT, file handling and JDBC for writing java program.
5. Understand and apply network programming concepts of Java
6. Design web page based on PHP and Java script.

UNIT-I

Introducing Classes – Class Fundamentals, Declaring Objects, Assigning Object Reference Variables, Introducing Methods, Constructors, this keyword, Method overloading, Using objects as parameters, Argument passing, Returning objects, Access control, static, final, Using command line arguments, variable length arguments.

Inheritance – Inheritance Basics, Using super, creates a Multilevel Hierarchy, When constructors are called? Method Overriding, Using abstract classes, Using final with Inheritance.

Packages and Interfaces – Packages, Access protection, Importing Packages, Interfaces.

Exception Handling – Exception-Handling Fundamentals, Exception Types, Uncaught Exceptions, Using try and catch, multiple catch Clauses, Nested try statements, throw, throws, finally.

12 Hours
UNIT - II

Multithreaded Programming – The Java Thread Model, The Main Thread, Creating a Thread, Creating Multiple Threads, Using isAlive() and join(), Thread Priorities, Synchronization, Inter-thread Communication.


UNIT – III

Swings – Swing key features, MVC connection, components and containers, Event handling, Painting, Simple examples; Exploring Swings, Swings UI components.


File Handling - Serial Access Files, File Methods, Redirection, Command Line Parameters, Random Access Files. 10 Hours

UNIT - IV

Java Database Connectivity (JDBC) - The Vendor Variation Problem, SQL and Versions of JDBC, Creating an ODBC Data Source, Simple Database Access, Modifying the Database Contents, Transactions, Meta Data, Scrollable ResultSets in JDBC 2.0, Modifying Databases via Java Methods.


and IP Addresses, Internet Services, URLs and DNS, TCP, UDP. The InetAddressClass, Using Sockets (TCP and UDP).  

10 Hours

UNIT – V

Java Servlets – Benefits, A simple Java Servlet, Anatomy of a Java Servlet, Reading data from a client, Reading HTTP Request Headers, Sending data to a client and writing the HTTP Response Header, Working with Cookies, Tracking Sessions.

Java Server Pages (JSP) – Installation, JSP Tags, Request String, User Sessions, Cookies, Session objects.  

10 Hours

TEXT BOOKS:
2002

REFERENCE BOOK:

*****************************

JAVA AND INTERNET TECHNOLOGIES LABORATORY

Sub Code : 13CS603  
Credits : 01  
Hrs/Week : 02

1. Java programs that includes each of the topic discusses in the theory.
2. Programs on Java Script, Simple programs and programs to design simple user interface using HTML.
INSTRUCTIONS:
1. In exam two programs may be asked randomly in any combination.
2. Any number of programs can be practiced in lab under each section.
3. Programs should incorporate as many features as possible.
4. Program need may be based on various features of the technology being used.
5. It is mandatory to score minimum marks both in Theory and Lab.

SOFTWARE TESTING AND AUTOMATION

Sub Code     : 13CS604
Credits      : 03
Hrs/Week     : 03
Total Hours: 39

Course Outcomes:

1. Understand and practice designing test cases, instances and scenarios using regular expressions, data structures, algorithms
2. Know to apply performance related issues, debugger related concepts
3. Explain about testing, various test methods, tools and automation

UNIT-I

1. Software practice I:


4 Hours
2. **Software practice II**:

UNIT-II

3. **Software practice III**:

4. **Software practice IV**:
Notation: Formatting data, Regular expressions, Programmable tools, Interpreters, Compilers and Virtual machines, Programs that write programs, Using Macros to generate code.
Debugging: Debuggers, Good Clues, Easy Bugs, No Clues, Hard Bugs, Last Resorts, Non Reproducible bugs, Debugging Tools, Other peoples bugs.

UNIT-III

5. **Software Testing**:
The six essentials of Software testing: The state of the art and state of the practice, The clean sheet approach to getting started, Establishing
UNIT-IV

6. Testing Methods:

Verification testing: Basic verification methods, Getting leverage on verification, Verifying documents at different phases, Getting the best from verification, Three critical success factors for implementing verification, Recommendation.
Validation Testing: Validation overview, validation methods, validation activities, Recommendation strategies for validation testing, Controlling validation costs: Minimizing the cost performing tests, Minimizing the cost of maintaining the tests, Minimizing validation testware development costs, Recommendations, Testing tasks, deliverables and Chronology, Master test planning, Verification testing tasks and deliverables, Validation testing tasks and deliverables. A testing orphan – User manuals, Product release criteria, Summary of IEEE/ANSI test related documents.
Testing Tools: Categorizing testing tools, Tool acquisition, Measurements, Useful and other interesting measures, Recommendations.

UNIT-V

7. Managing Testing Technology & Automation:

Organizational approaches to testing: Organizing and Reorganizing testing, Structural design elements, Approaches to organizing the test functions, Selecting the right approach: Current practices, trends, challenges, GUIs: What is new here, Usage testing, tester to developer ratios, Software measures and practices benchmark study, Getting sustainable gains in place, Getting gains to happen, Getting
help, Follow up, Standards relevant to software engineering and testing, Verification check lists. Automation techniques and tools.  

8 Hours

TEXT BOOKS:
   Chapters 1(except 1.5), 2 (except 2.4&2.5),  3(except 3.5,3.6,3.7),4 (except 4.3,4.4), 5, 7, 8 , 9 (except 9.7)
   Chapters 1 to 15.

REFERENCE BOOKS:

ENTRY EDGE: IMMERSIVE GROUP WORKSHOP (IGW)

Sub Code       : 13CS605 
Duration: 5 Days 
Timings : 9.00 AM to 12.30 PM, 1.15 PM to 4.45 PM

Syllabus
Module 1: Minds-on and hands-on simulation project
• Understanding Task environment – Goals, responsibilities, Task focus
• Working in Teams towards common goals
• Organizational performance expectations–technical and behavioural competencies.

5 Hours

Module 2: Re-enforcement of critical individual skills and behaviours
• Application of individual effectiveness skills in team and organizational context – improving self awareness, goal setting, time management, communication and presentation skills.

5 Hours

Module 3: Etiquettes and Ethics
• Professional etiquettes at workplace – dressing, telephone, e-mail, meeting and general behaviour
• Basic honesty & respect for law / rules
• Conflict of interest
• Use of organizational resources
• Misrepresentation and misappropriation
• Intellectual property
• Whistle blowing

7 Hours

Module 4: Interpersonal Behaviour & relationship skills
• Establishing trust based relationships in team & organizational environment
• Trust equation – credibility, responsiveness, integrity, self-interest

3.5 Hours

Module 5: Dealing with Conflicts
Orientation towards conflicts in team and organizational environment
• Understanding sources of conflicts
• Conflict resolution styles and techniques

3.5 Hours

**Pedagogical tools & techniques used in the workshop**

• Organizational templates for simulating a organizational context-structures, units, roles and activities
• Metaphoric scenarios for simulating real –life tasks and dynamics in a team/project context
• LEGO™ building blocks for simulating last-mile technical activity in teams
• Case studies, Role play scenarios group learning activities, observation and feedback.

**Note:** Evaluation is done and a grade of P (pass) or NP (not pass) is awarded

***************

**PATTERN RECOGNITION**

<table>
<thead>
<tr>
<th>Sub Code</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>13CS611</td>
<td>03</td>
</tr>
<tr>
<td>Hrs/Week</td>
<td></td>
</tr>
<tr>
<td>03</td>
<td>Total Hours: 39</td>
</tr>
</tbody>
</table>

**UNIT-I**

Introduction: Machine Perception, Pattern Recognition systems, Design cycle, learning and adaptation (1.1, 1.3, 1.4, 1.5 of Ref.1)
Bayesian Decision Theory: Introduction, Bayesian Decision theory – continuous features, classifiers, discriminant functions, and decision surfaces, normal density and discriminant functions, Bayes decision theory – discrete features (2.1, 2.2, 2.4, 2.5, 2.6, 2.9 of Ref. 1) 8 Hours

**UNIT-II**
Maximum likelihood and Bayesian parameter estimation: Introduction, maximum likelihood estimation, Bayesian Estimation, Bayesian parameter estimation, problem of dimensionality, sufficient and exponential family, complex analysis & discriminants, (3.1 to 3.8 of Ref.1)  

**8 Hours**

**UNIT-III**

Nonparametric Techniques: Introduction, Density Estimation, Parzen Windows, $k_n$-nearest neighbour estimation, nearest neighbor rule, metrics and nearest-neighbor classification, fuzzy classification, reduced coulomb energy, approximations by series expansions (4.1 – 4.9 of Ref.1)  

**8 Hours**

**UNIT-IV**

Linear discriminant functions: Introduction, linear discriminant functions, generalized linear discriminant functions, minimizing the Perceptron criterion function, relaxation procedures, nonseparable behaviours, minimum squared-error procedures, Ho-Kashyap procedures (5.1 to 5.9 of Ref.1)  

**8 Hours**

**UNIT-V**

Unsupervised learning and clustering: Mixture densities and identifiability, maximum-likelihood estimates, application to normal mixtures, unsupervised Bayesian learning, data decryption and clustering, criterion functions and clustering, hierarchical clustering, on-line clustering. Component analysis, low-dimensional representations and multidimensional scaling (10.1 to 10.14 except 10.8, 10.12 of Ref. 1)

Syntactic pattern Recognition: Overview, qualifying structure in pattern description and recognition, grammar based approach, elements of formal grammar (Chap. 3 of Ref. 2)  

**7 Hours**
TEXT BOOKS:


**********************************

SYSTEM SIMULATION & MODELING

Sub Code : 13CS612
Hrs/Week : 03
Credits : 03
Total Hours: 39

UNIT-I
1. Introduction To Simulation:
When Simulation is the Appropriate Tool; When Simulation Is Not Appropriate; Advantages and Disadvantages of Simulation; Areas of Application; Systems and System Environment; Components of a System; Discrete and Continuous Systems; Model of a System; Types of Models; Discrete-Event System Simulation; Steps in a Simulation Study.

8 Hours

2. General Principles:

UNIT-II

3. Random-Number Generation:
Properties of Random Numbers; Generation of Pseudo-Random Numbers; Techniques for Generating Random Numbers; Tests for Random Numbers.

4. **Random-Variate Generation:**
Inverse Transform technique: Exponential Distribution, Uniform Distribution, Discrete Distributions; Acceptance-Rejection Technique: Poisson Distribution.  

6 Hours

**UNIT-III**

5. **Input Modeling:**
Data Collection; Identifying the distribution with Data; Parameter Estimation; Goodness of Fit Tests; Selecting Input Models without Data; Multivariate and Time-Series Input Models.  

8 Hours

**UNIT-IV**

6. **Verification And Validation Of Simulation Models:**
Model Building, Verification and Validation; Verification of Simulation Models; Calibration and Validation of Models  

8 Hours

**UNIT-V**

7. **Simulation Of Computer Systems:**
Introduction; Simulation Tools; Model Input; High-Level Computer-System Simulation; CPU Simulation; Memory Simulation.  

8 Hours

**TEXT BOOK:**

**REFERENCE BOOKS:**

**********************

**DISTRIBUTED SYSTEMS**

<table>
<thead>
<tr>
<th>Sub Code</th>
<th>Credits</th>
<th>Hrs/Week</th>
<th>Total Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>13CS613</td>
<td>03</td>
<td>03</td>
<td>39</td>
</tr>
</tbody>
</table>

**UNIT-I**

**CHARACTERIZATION OF DISTRIBUTED SYSTEMS:**
Introduction, Examples of distributed systems, Resource sharing and the web, Challenges.

**SYSTEM MODELS:** Introduction, Architectural models, Fundamental mode.  

**UNIT-II**

**INTERPROCESS COMMUNICATION:** Introduction, The API for the internet protocols, External data representation and marshalling, Clint-server communication, Group communication.

**UNIT-III**

**DISTRIBUTED OBJECTS AND REMOTE INVOCATION:**
Introduction, Communication between distributed objects, Remote procedure call, Events and notifications.

**UNIT-IV**

**SECURITY:** Introduction, Overview of security technique cryptographic algorithms, Digital signature, Cryptography progrmatics.
TIME & GLOBAL STATES: Introduction, Clocks, Events, Process states, Synchronizing physical clocks, Global states, Distributed debugging.

UNIT-V
COORDINATION AND AGREEMENT: Distributed mutual exclusion, Elections, Multicast communication. 9 Hours

CORBA CASE STUDY: Introduction, CORBA RMI, CORBA Services.

ADVANCED DBMS

<table>
<thead>
<tr>
<th>Sub Code</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>13CS614</td>
<td>03</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Hrs/Week</th>
<th>Total Hours:</th>
</tr>
</thead>
<tbody>
<tr>
<td>03</td>
<td>39</td>
</tr>
</tbody>
</table>

Course Outcomes:

1. Understand the concepts of indexing and storage.
2. Apply indexing and sorting.
3. Explain the methods of query optimization.
4. Explain the concept and application of object databases.
5. Explain the distributed databases.

UNIT-I
Overview of storage and indexing, disks and files:
Data on external storage; File organizations and indexing; Index data structures; Comparison of file organizations; Indexes and performance tuning. Memory hierarchy; RAID; Disk space management; Buffer manager; Files of records; Page formats and record formats.

Tree structured indexing: Intuition for tree indexes; Indexed Sequential Access Method (ISAM); B+ trees - Search, Insert, Delete, Duplicates; B+ trees in practice. 8 Hours
UNIT-II

Hash based indexing: Static hashing; Extendible hashing, Linear hashing, comparisons.
External Sorting: When does a DBMS sort data? A simple two-way merge sort; External merge sort, Using B+ trees for sorting.
Evaluating Relational Operators: The Selection operation; General selection conditions; The Join operation; The Projection operation; The Set operations; Aggregate operations; The impact of buffering.

8 Hours

UNIT-III

Physical Database Design and Tuning: Introduction; Guidelines for index selection, examples; Clustering and indexing; Indexes that enable index-only plans; Overview of database tuning; Choices in tuning the conceptual schema; Choices in tuning queries and views; Impact of concurrency; DBMS benchmarking.

7 Hours

UNIT-IV

Object Databases:
Concepts for Object Databases: Overview of Object-Oriented Concepts, Object Identity, Object Structure, and Type Constructors, Encapsulation of Operations, Methods, and Persistence, Type and Class Hierarchies and Inheritance, Complex Objects;
Object Database Standards, Languages, and Design: Overview of the Object Model of ODMG, The Object Definition Language ODL, The Object Query Language OQL, Overview of the C++ Language Binding, Object Database Conceptual Design;
Object-Relational and Extended-Relational Systems: Overview of SQL and its Object-Relational features, Object-Relational Features of Oracle 8.

8 Hours
UNIT-V

Distributed Databases:
Distributed Database concepts; Data Fragmentation, Replication, and Allocation Techniques for Distributed Database Design; Types of Distributed Database Systems; Query Processing in Distributed Databases; Overview of Concurrency Control and Recovery in Distributed databases; Distributed databases in Oracle.

Security and Authorization:

TEXT BOOKS:


REFERENCE BOOKS:


CLOUD COMPUTING AND INFRASTRUCTURE MANAGEMENT

<table>
<thead>
<tr>
<th>Sub Code</th>
<th>Credits</th>
<th>Hrs/Week</th>
<th>Total Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>13CS615</td>
<td>03</td>
<td>03</td>
<td>39</td>
</tr>
</tbody>
</table>
Course Outcomes:
1. Understand the concept of cloud computing business need and various networking methods.
2. Explain and implement virtualization at various levels
3. Explain the infrastructure management for cloud environment.
4. Explain the security in application to cloud.

UNIT-I
Eras of computing, Parallel vs. Distributed Computing, Elements of Parallel Computing- (What is parallel computing, hardware architecture for Parallel processing, approaches to parallel programming, levels of parallelism, Laws of caution). Elements of Distributed Computing- (General concepts and definitions, components of a distributed system, Architectural styles for distributed computing, models for inter-process communication, Technologies for distributed computing-Remote procedure call, Service oriented computing).

8 Hours

UNIT-II
Classic data center, its elements, challenges and benefits. Data center management Steps in transitioning to cloud- consolidation, automation, IT as a service.

Cloud computing Architecture: - Introduction, Cloud reference models- (Architecture, Infrastructure/Hardware as a service, Platform as a service, Software as a service), Types of cloud – (Public Clouds, Private Clouds, Hybrid Clouds, Community Clouds), Economics of cloud, Open challenges.

6 Hours

UNIT-III
Virtualization: – Introduction, characteristics of virtualized environments, taxonomy of virtualization technique- (execution of virtualization, other types of virtualization-Compute, Storage, Network,
Desktop, Application). Virtualization and cloud computing, Pros and Cons of virtualization, Technology examples- XEN, VMware, Microsoft Hyper-V.  

UNIT-IV


UNIT-V


Text Books:


References:


BIG DATA

Sub Code : 13CS616
Credits : 03
Hrs/Week : 03
Total Hours: 39

UNIT-I

UNDERSTANDING BIG DATA:

What is big data, why big data, convergence of key trends, unstructured data, industry examples of big data, web analytics, big data and marketing, fraud and big data, risk and big data, credit risk management, big data and algorithmic trading, big data and healthcare, big data in medicine, advertising and big data, big data technologies, introduction to Hadoop – open source technologies – cloud and big data

UNIT-II

NOSQL DATA MANAGEMENT:

Introduction to NoSQL, Aggregate data models: aggregates, key-value and document data models, Relationships, graph databases, schemaless databases, Distribution models: sharding, master-slave replication, peer-peer replication, sharding and replication, MapReduce: partitioning and combining -- Composing Map-Reduce Calculations.

UNIT-III

BASICS OF HADOOP:

Data format, Analyzing data with Hadoop, Scaling out, Hadoop streaming, Hadoop pipes. Design of Hadoop distributed file system (HDFS), HDFS concepts, Hadoop I/O, data integrity, compression, serialization

UNIT-IV
MAPREDUCE APPLICATIONS:
MapReduce workflows, unit tests with MRUnit, test data and local tests, anatomy of MapReduce job run, job scheduling, shuffle and sort

7 Hours

UNIT-V

HADOOP RELATED TOOLS:

Introduction to Hbase: The Dawn of Big Data, the Problem with Relational Database Systems, Introduction to Hive, data types and file formats, HiveQL data definition, HiveQL data manipulation, HiveQL queries. 8 Hours

TEXT BOOKS:


REFERENCE BOOKS:

1. Alex Holmes , “Hadoop in Practice“

2. Lars George, "HBase: The Definitive Guide", O'Reilley, 2011

BUSINESS PROCESS MANAGEMENT

Sub Code : 13CS617
Credits : 03
Hrs/Week : 03
Total Hours: 39

Objectives:

The course enables students –

- To Understand concepts of Business Process Management (Process Modeling & Analysis using BPM tools)

Outcomes:

The students will be able to -

- Gain knowledge of Business Process Management concepts

UNIT - I

Introduction to BPM


7 Hours

UNIT - II

BPM Life Cycle Methodology


UNIT - III

Business Process Management Overview


UNIT - IV
Creating User Interfaces
Creating user interfaces, Coaches - Difference between Coaches and Heritage Coaches. Developing reusable Coach Views - Coach Views, Templates, Stock controls - Button, Checkbox, Date Time Picker, Horizontal Section, Output Text, Select, Table Tabs, Text, Vertical Section, Stock content controls, Document List - Document Viewer. Advanced items for Coach Views - Content box, Custom HTML, Boundary events. Binding views with data - Defining Coach View behavior. Calling Ajax services from Coach Views, Example, Accessing a child Coach view, Building Coaches - Coach View API Reference. Architecting complex process applications - Designing process interactions for business users, configuring a role-based business user interface. Developing flexible and efficient process applications, setting up collaboration features for business users, enabling task management, integrating with other systems, creating outbound integrations, Integration Service implementations, IBM Case Manager Integration Service implementations - Using a Web Service Integration step in an integration service, Using IBM Business Process Manager SQL Integration services. Creating inbound integrations - Building a sample inbound integration. Posting a message to IBM Business Process Manager Event Manager, Understanding the message structure, Passing complex variable types to Undercover Agents, Passing IBM BPM Structured types, Passing Record type, Passing Date/Time types, Passing Boolean type, Passing Map type, Passing XmlDocument type, Passing XMLElement type, Passing XMLNodeList type, Passing ANY type, Publishing IBM Business Process Manager Web Services - Web services, compatibility, Configuring conditional activities, Globalization. 8 Hours

UNIT - V
Dashboards and Reports

Business value, Solution overview - Solution architecture, IBM products used in the solution, Software development roles that are associated with the solution, Product-specific roles that are associated with the solution,
Usage scenarios. IBM Solution for Collaborative Lifecycle Management - InfoSphere Data Architect, WebSphere Operational Decision Management, Business Process Manager Advanced, Integration. Designing process interactions for business users, Factors affecting BPEL process interactions - Interaction style, BPEL process type, WSDL operation type, Service endpoint resolution. Developing flexible and efficient process applications - Enabling processes for tracking and reporting, Racking IBM Business Process Manager performance data. Defining reports in Process Designer (deprecated), Defining a custom layout Process Designer for reports (deprecated), IBM Business Monitor dashboards - Overview, Get the spreadsheets, Define metrics, Define KPIs, Define reports, Generate a monitor model using the CSV tool, Deploy your monitor model, Send events to the monitor model, View the dashboards. 8 Hours

TEXT BOOK-
1. Business Process Management (IBM ICE Publication)

BUSINESS PROCESS MANAGEMENT LAB EXERCISES

- Creating a business process definition (BPD)
- Creating a participant group
- Creating a user attribute definition
- Modelling process execution paths using sequence flow
- Implementing activities
- Configuring an activity for multi-instance looping
- Setting up a routing policy
- Modelling subprocess activities
- Building services
- Adding a decision service to a process
- Creating rules using the rule editor
- Testing a decision service
DATA WAREHOUSE & MULTIDIMENSIONAL MODELING

Sub Code : 13CS618
Credits : 02
Hrs/Week : 03
Total Hours: 39

Objectives:

The course enables students –
1. Understand the fundamentals of Data Warehousing
2. Learn modelling of datawarehousing
3. Understand the concepts of Multi-Dimensional Modeling and learn the Methodology
4. Learn Non-Temporal Design of R-OLAP
5. Learn Non-Temporal Design of M-OLAP.

Outcomes:

The students will be able to –
1. Gain fundamental concepts of data warehousing
2. Develop a model for datawarehousing
3. Do multidimensional modelling of datawarehousing.
4. Design R-OLAP
5. Design M-OLAP

UNIT - I

Introduction to Data Warehousing

Data Warehouse Architectures, A perspective on decision support applications. 7 Hours

UNIT - II
Data Warehousing and Modeling

An Introduction to Data Warehouse Modeling, Differentiating the Warehousing model from the OLTP model, Warehouse Modeling Approaches, OLAP – OnLine Analytical Processing, Basic OLAP Operations. 8 Hours

UNIT - III

Multi-Dimensional Modeling – Methodology

Requirement Analysis, Requirements modeling, Terminologies in Multidimension Model, Multi-Dimensional Model Structures, Solution Validation Techniques, Detailed Dimension Modeling. 8 Hours

UNIT – IV

Non-Temporal Design - R-OLAP-I

R-OLAP and its design techniques, Design techniques of an R-OLAP System, Dimension-Oriented Design techniques, Fact-oriented Design Techniques. 8 Hours

UNIT - V

Non-Temporal Design - R-OLAP-II

Utilize Cubing Services to improve R-OLAP and M-OLAP performance, Cubing Services performance and scalability, Scalability, Cubing Services security, Role-based security in Cubing Services. 8 Hours

Non-Temporal Design - M-OLAP

IBM Cognos Architecture, Sparse and Dense Dimensions – with Hyperion Essbase, MOLAP characteristics, Online Data Analysis MOLAP and ROLAP.

TEXT BOOK-
Data Warehouse & Multidimensional Modeling (IBM ICE Publication)
DATA WAREHOUSE & MULTIDIMENSIONAL MODELING LAB

Sub Code : 13CS618
Credits : 01
Hrs/Week : 02

- Introduction to the Case Study
- Business Requirements for Rental and Sales Analysis
- Business Requirements for Working Shifts
- Business Requirements for Customers
- Build a snowflake model for the Customer dimension.
- Build a Customer dimension table with the same information content as the previously developed snowflake model.
- Using any design techniques you know, build an optimum model for the Customer dimension, taking all the available statistics and association properties into account.

**************

INFORMATION AND STORAGE MANAGEMENT

Sub Code : 13CS619
Credits : 03
Hrs/Week : 03
Total Hours: 39

Objectives:

1. To understand the concept of server centric architecture and storage centric architecture.
2. To understand the architecture of intelligent disk subsystems.
3. To compare the advantages and disadvantages of different RAID levels.
4. To compare the techniques of instant copies, remote mirroring.
5. To understand the layered design of Fibre Channel Protocol Stack
6. To compare Fibre channel SAN, NAS, iSCSI storage technologies
7. To understand various storage virtualization levels.
UNIT - I

INTRODUCTION: Server Centric IT Architecture and its Limitations; Storage – Centric IT Architecture and its advantages; Case study: Replacing a server with Storage Networks;

INTELLIGENT DISK SUBSYSTEMS: Architecture of Intelligent Disk Subsystems; Hard disks and Internal I/O Channels, JBOD. 7 Hours

UNIT – II

INTELLIGENT DISK SUBSYSTEMS: Storage virtualization using RAID, different RAID levels; Caching; Intelligent Disk Subsystems, Instant copies, Remote mirroring, LUN masking, Availability of Disk Subsystems

I/O TECHNIQUES: The Physical I/O path from the CPU to the Storage System; SCSI, Fiber Channel Protocol Stack; Fiber Channel SAN; 8 Hours

UNIT-III

FILE SYSTEM AND NAS:
Local File Systems; Network file Systems and file servers; Shared Disk file systems; Comparison of fiber Channel SAN, NAS AND iSCSI SAN.

STORAGE VIRTUALIZATION: Virtualization in the I/O Path; Limitations and requirements. 8 Hours

UNIT-IV

STORAGE VIRTUALIZATION: Definition of Storage virtualization; Implementation Considerations; Storage virtualization on Block or file level; Storage virtualization on various levels of the storage Network; Symmetric and Asymmetric storage virtualization in the Network
UNIT-V


Storage Sharing: Disk storage pooling, Dynamic tape library sharing, Data sharing.

Availability of Data: Failure of an I/O bus, Failure of a server, Failure of a disk subsystem

Failure of virtualization in the storage network Failure of a data centre based upon the case study ‘protection of an important database’.

Adaptability and Scalability of IT Systems: Clustering for load distribution, Web architecture, Web applications based upon the case study ‘travel portal’

8 Hours

TEXT BOOK:


REFERENCE BOOKS:


VLSI DESIGN

Sub Code : 13CS621
Credits : 03
Hrs/Week : 03
Total Hours: 39

UNIT - I

8 Hours

UNIT - II

**LOGIC GATES, COMBINATIONAL LOGIC NETWORKS**

Introduction, Combinational Logic Functions, Static Complementary gates, Wires and delay, switch logic, layout design methods, Simulation, Combinational Network Delay, Crosstalk, Power Optimization, Switch Logic Networks, Combinational Logic Testing  
8 Hours

UNIT - III

**Sequential machines, Subsystem Design**

Sequential Machines: Introduction , latches and flipflops, sequential systems and clocking disciplines, Sequential System Design, Power Optimization, Design Validation, Sequential Testing, Sub-system Design: Introduction, Subsystem Design Principles, Combinational Shifters adders, High-Density Memory, FPGAs, PLAs,  
8 Hours

UNIT - IV
Floor Planning, Architectural Design, Chip Design


UNIT-V

CAD, Design Modeling


TEXT BOOKS:

2. VLSI CAD – Niranjan N. Chiplunkar & Manjunath Kothari, PHI Learning, 2011

********************************

MULTI-CORE ARCHITECTURE AND PROGRAMMING

Sub Code : 13CS622  
Credits : 03  
Hrs/Week : 03  
Total Hours: 39

Course Outcomes:

1. Understand the concept of multi core architecture
2. Design parallel program using the multi thread concept.
3. Explain and implement program using Openmp.
4. Explain the optimization of computer for parallel program
UNIT-I

Introduction to Multi-core Architecture: Introduction, Moore’s law, Amdhal’s law, Gustafson’s law, Motivation for Multi-core processors, Types and levels of parallelism, Flynn’s classification of multi-processors, Introduction to parallelization and vectorization: Data dependencies, SIMD technology, Hardware Multithreading vs. Software multi threading, Hyper threading, SMT, Case Study of multi-core processors: Intel, AMD, IBM/Sony 8 Hours

UNIT-II

Concepts and Design of Parallel and Thread Programming: Definition of thread and process, Parallel programming models, Parallel Programming constructs: Synchronization, Deadlock, Critical sections, Threading APIs- Win 32, POSIX threads. 7 Hours

UNIT-III

Parallel Programming: MPI Model: Collective communication, Data decomposition, Communicators and topologies, point-to-point communication, MPI Library, OpenMP: Directives and clauses, environment variables, Programs using OpenMP and MPI. Introduction to Intel TBB, Thread-Safeness, Cache related issues 7 Hours

UNIT-IV

Multithreaded Program Debugging: Benchmarks, performance tools, VTune Performance analyzer, thread checker, thread profiler, hotspots, performance issues in algorithms, branch misprediction, cache organization, cache loads, efficiency, hardware and software prefetch. 7 Hours

UNIT-V
**Compiler Optimizations and Parallel Algorithms:** Compilers for High performance Computing, compiler optimization, code and loop optimization, scalar and vector processing, temporal and spatial locality- matrix multiplication example. OS support to multi-core architectures. Parallel algorithms study and analysis- The Sieve of Eratosthenes, Floyd’s algorithm, Matrix-Vector multiplication, Monte Carlo methods, Matrix Multiplication, Parallel Quicksort Algorithm  

9 Hours

**TEXT BOOKS & REFERENCE BOOKS:**

1. Multicore programming- Increasing performance through software multithreading.-- Shameem Akhter and Jason Roberts, Intel press
3. Parallel Programming in C with MPI and OpenMP by Michael J. Quinn, Tata McGraw-Hill Edition
4. www.openmp.org
5. www tutorials on introduction to parallel computing

***************

**MICROCONTROLLERS**

<table>
<thead>
<tr>
<th>Sub Code</th>
<th>Credits</th>
<th>Total Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>13CS623</td>
<td>03</td>
<td>39</td>
</tr>
<tr>
<td>Hrs/Week</td>
<td>03</td>
<td></td>
</tr>
</tbody>
</table>

**UNIT - I**

Differences between Microprocessors and Microcontrollers, RISC and CISC CPU architectures, Harvard and Von-Neumann CPU architectures, Commercially available 8, 16 and 32 bit Microcontrollers from Intel, ARM and others.
Definition of Embedded system, Use of Microcontrollers in Embedded system, High level software development life cycle. ARM processor Architecture overview.  

**UNIT - II**

Introduction to ARMv7-M ISA and basic Programmer’s model; ARM/Thumb assembly instructions, addressing modes, processor mode, register set, basic assembly instructions, MDK-ARM simulator, Assembly language programming.  

**UNIT - III**

C programming overview, I/O, stack, subroutine, logical and shift operation, KEIL utility, debugging in MDK-ARM simulator, Arithmetic operations, ARM prediction and condition execution, timers, pointers and advanced debugging, Finite State Machines, I/O synchronization, stack frames, FSMs in C.  

**UNIT - IV**

Cortex-M3 exception and interrupt handling, Systick timer and periodic interrupts, Memory mapped peripherals, LCD interface, MDK-ARM C programming, Mixing C and assembly, UART, SPI, I2C, CAN, DAC and ADC on ARM MCU, Review of threads and thread communication.  

**UNIT - V**

Hands on sessions to carry out any 10 of the following exercises using simulator and ARM Kit STM32L-Discovery:
a. Digital I/O, MCU pin direction, and logical functions, written in assembly and simulation
b. LEDs and switching, written in assembly & simulation
c. Use switches and LEDs, and control LED intensity using switches written in C and simulation
d. Traffic Light Controller with bits, written in assembly and simulation
e. Reset system using watchdog timer in case of error.
f. Simple FSM simulator in C
g. MCU bring-up and initialization, M3 vector table, setting up dummy handlers
h. LCD device driver and test, written in assembly and C (simulation and on board)
i. UART echo test (simulation and on board), and control LED intensity based on UART parameters (simulation and on board)
j. Display temperature on PC over UART (simulated and board)
k. Sample sound and plot amplitude vs. time on PC
l. Sample sound over microphone and display intensity through LEDs
m. FIFO queue, threads, mixture of assembly and C (simulated and board)
n. Real-time Position Monitor, ADC, interrupts, LCD, mixture of assembly and C (simulated and board)
o. Digital Piano or pacemaker using a DAC, C (simulated and board)
p. Generate RT clock using timers and output time over UART (simulation and board) **10 Hours**

**TEXT BOOKS AND RESOURCE BOOKS:**

EMBEDDED AND REAL TIME SYSTEMS

Sub Code : 13CS624  Credits : 03
Hrs/Week : 03  Total Hours: 39

Course Outcomes:
1. Understand the concept of embedded system and its blocks.
2. Understand the application of times and controllers in embedded system design.
3. Explain the concept of RTOS.
4. Explain the Intel ATOM architecture and its interface.

Course Objectives:
To give an overview of Embedded systems, typical building blocks of Embedded system and Software aspects of embedded applications including features of Real time operating systems. Course gives an overview of Intel ATOM processor, highlighting its low power features which are desirable in low power embedded system design. Students will also work on Intel ATOM boards and interface the same to some peripheral devices – thus getting a feel of building typical embedded system.

UNIT – I

Embedded system definition, characteristics, design metrics; Processor, IC and design technologies; Embedded system examples, Digital Camera
building blocks, Combinational and sequential building blocks. Use of DSP Processors, SoCs and Microcontrollers in embedded systems. Overview of 8051 microcontroller.  

UNIT - II
Timers, ADCs, Keypad controllers, LCD controllers, stepper motor and DC motor control, Custom Single Purpose processor design examples: GCD Generator, 4 bit multiplier, Communication bridge. Memory – Composing memory, memory hierarchy and Cache memory, interfacing- Serial, Parallel and Wireless Protocols.  

UNIT - III
Introduction to Real – Time Operating Systems, features, Examples of RTOS, typical RTOS functions. Interrupt handling and latency, Shared data problems, Tasks and Task States, Task scheduling, Inter-task communication and synchronization, Semaphores, Message Queues, Mailboxes and Pipes, Reentrant functions, Typical software architectures, Embedded Software development and testing tools, JTAG debugger, typical system boot flow diagram  

UNIT - IV
Intel ATOM Processor Architecture, Platform architecture and Micro architecture details, Overview of Assembly language programming of ATOM Processor, Low power issues of ATOM processor, ATOM processor series.  

UNIT - V
Intel ATOM Processor kit details, I/O options available, Keyboard and Mouse interface, GPS, GSM and RFID interface – Hands On, Overview of Device drivers.
TEXT BOOKS & RESOURCE BOOKS :

1. Frank Vahid and Tony Givargis, Embedded Systems Design – A unified Hardware/Software Introduction, John Wiley, 2002 (Chapter 1, 2, 4)

2. David E.Simon, An Embedded Software Primer, Pearson Education Asia, First Indian Reprint 2000. (Chapter 6, 7, 8, 9)


4. Lori Matassa and Max Domeika “Break away with Intel Atom Processors: A guide to Architecture Migration” Intel Press, 2010 (Chapter 3, selected topics of Chapter 4 & 5)


6. Intel Websites

7. NPTEL videos on Embedded Systems

8. Lab Manual of “Embedded system lab with ATOM Kit & Interfaces”

*******************************

PROGRAMMING LANGUAGES

<table>
<thead>
<tr>
<th>Sub Code</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>13CS625</td>
<td>03</td>
</tr>
<tr>
<td>Hrs/Week</td>
<td>03</td>
</tr>
</tbody>
</table>

UNIT-I

INTRODUCTION; NAMES, SCOPE, AND BINDINGS – 1: Language design; Programming language spectrum; Why study
programming languages? Compilation and interpretation; Programming environments. Names, scope, and bindings: Concept of binding time; Object lifetime and storage management; Scope rules and implementing scope. The binding of reference environments; Binding within a scope; Separate compilation.  

7 Hours

UNIT – II
Control Flow: Expression evaluation, Structured and unstructured flow; Sequencing; Selection; Iteration; Recursion; DATA TYPES – 1: Type systems; Type checking; Records and variants; Arrays.

8 Hours

UNIT – III

DATA TYPES - 2: Strings; Sets; Pointers and recursive types; Lists; Files, and Input/Output; Equality testing and assignment.
Subroutines and Control Abstraction - 1: Review of stack layout; Calling, sequences; Parameter passing; Generic subroutines and modules; Exception handling.

7 Hours

UNIT – IV

CONTROL ABSTRACTION – 2; DATA ABSTRACTION, OBJECT ORIENTATION: Control abstraction – 2: Coroutines. Data Abstraction, Object Orientation: Object oriented programming; Encapsulation and Inheritance; Multiple inheritance;

8 Hours

UNIT - V

FUNCTIONAL LANGUAGES, LOGIC LANGUAGES, SCRIPTING

FUNCTIONAL LANGUAGES : Functional Languages: Origins; Concepts; An overview of scheme; Evaluation order revisited; Higher-order functions; Functional
programming in perspective. Logic Languages: Concepts; Prolog; Logic programming in perspective. Scripting Languages: Common characteristics.

TEXT BOOK:

REFERENCE BOOKS:

******************************

MULTICAST COMMUNICATIONS

Sub Code : 13CS626
Credits : 03
Hrs/Week : 03
Total Hours: 39

UNIT - I

The basics of group communications: Types of communications; Multicast vs Unicast; Scalability; Applications of group communication; characteristics of groups; Special aspects of group communication.

7 Hours

UNIT - II

Multicast Routing: Basic Routing algorithms; Group dynamics; scoping and multicast address allocation; Concepts of multicast routing; Multicast routing on the internet.

8 Hours

UNIT - III
Multicast in ATM networks: The switching technology ATM; ATM multicast. Transport protocols: UDP; XTP. 8 Hours

UNIT - IV
Transport protocols: MTP; RMP; LBRM; SRM; RMTP 8 Hours

UNIT - V
Mbone- The Multicast Backbone of the Internet: Mbone architecture; Mbone applications; Mbone Tools; Outlook; Multicast Routing and Mobile Systems. 8 Hours

TEXT BOOK:

CLOUD DEPLOYMENT MODELS

<table>
<thead>
<tr>
<th>Sub Code</th>
<th>13CS627</th>
<th>Credits</th>
<th>03</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hrs/Week</td>
<td>02</td>
<td>Total Hours: 39</td>
<td></td>
</tr>
</tbody>
</table>

Objectives:
The course enables students –
1. To learn Public cloud deployment model
2. To learn Private cloud deployment model
3. To learn Hybrid cloud deployment model.

Outcomes:
The students will be able to -

1. Understands Cloud deployment models in details e.g., Public, Private and Hybrid.

UNIT-I

Cloud Computing Platform Overview


8 Hours

UNIT-II

Private Cloud Deployment Model


8 Hours

UNIT-III

Public Cloud Deployment Model

What is a Public Cloud?, Illustration of Public Cloud, Why Public Cloud, Advantages of Public Cloud, Limitations of Public Cloud, Low degree of security and control, Lack of control on infrastructure, configuration, Network latency and accessibility concerns, Highest long term cost, Public v/s Private, Journey into Public Cloud, Revisit the idea of adopting public cloud, Cloud vendor selection, Migrating to Cloud, Cloud vendor selection, SLA – Service Level Agreements,
Credits/Compensation terms, Credit process, Disaster recovery plan, Exclusions, Security and Privacy, Periodic upgrade and maintenance, Data location and Jurisdiction, Pricing and Measurability, Interoperability and Lock-in, Exit process/Termination policies, Proven track record, Public cloud vendors, Case studies.  

**8 Hours**

**UNIT-IV**

**Hybrid Cloud Deployment Model**

What is a Hybrid Cloud?, Why Hybrid Cloud, Illustration of Hybrid Cloud, Advantages of Hybrid Cloud, Challenges of Hybrid Cloud, Develop and manage hybrid workloads, Developing applications for hybrid cloud, Develop applications using PaaS, Managing hybrid workloads, Journey into Hybrid Cloud. Step 1: Assess current IT infrastructure and business, Step 2: Explore cloud computing, Step 3: Create cloud deployment strategy plan, Step 4: hybrid cloud implementation.  

**8 Hours**

**UNIT-V**

**Cloud Computing Platform Lab**

OpenStack Introduction, OpenStack Architecture, Lab Environment, Hardware requirements, Software requirements, High level overview of setup.  

**7 Hours**

**TEXT BOOK-**

Cloud Deployment Model (IBM ICE Publication)

**REFERENCES AND URLS**

1. Cloud Computing For Dummies (November, 2009), Judith Hurwitz, Robin Bloor, Marcia Kaufman, Dr. Fern Halper
2. Developing and Hosting Applications on the Cloud (July, 2012), Alex Amies, Harm Sluiman, Qiang Guo Tong, Guo Ning Liu
http://en.wikipedia.org/wiki/Cloud_computing

*****************************

LAB COMPONENTS

Cloud Deployment Model Lab Exercises
Open Stack Installation
Familiarize with OpenStack dashboard
Deploy a virtual machine instance
Deploy a Linux VM from an ISO Image

Deploy a VM from an image snapshot
User and Project management
Common Cloud Management tasks
Adding a new compute note
Overview of Nagios
Overview of Openstack CLI

*****************************

BUSINESS INTELLIGENCE [COGNOS BI]

Sub Code : 13CS628  Credits : 02
Hrs/Week : 03  Total Hours: 39

Objectives:

The course enables students –
1. Learn the basics of Business Intelligence.
2. Learn dashboards design by utilizing key performance indicators that managers can use to improve day-to-day business operations.
3. To learn how to plan and implement BI development projects.
4. To know the administrative and deployment scenarios & issues in BI space.
Outcomes:
The students will be able to –
1. Understand & appreciate the use of analytical skills and business principles in operational and strategic decision-making by means of BI.
2. Design and develop dashboards.
3. Learn the best practices to work on BI projects.
4. Use IBM Cognos BI tool to develop, implement and administrate wide range of BI artifacts.

UNIT-I
Introduction to Business Intelligence
Business Intelligence (BI), Scope of BI solutions and their fitting into existing infrastructure, BI Components and architecture, BI Components, Future of Business Intelligence, SaaS and Cloud computing techniques.

UNIT-II
Business Intelligence (Continued)
Functional areas of BI tools, End user assumptions, setting up data for BI, Data warehouse, OLAP and advanced analytics, supporting the requirements of senior executives including performance management, Glossary of terms and their definitions specific to the field of BI and BI systems.

UNIT-III
Elements of Business Intelligence Solutions
Business Query and Reporting, Reporting, Dashboards and Scorecards Development, Development, Scorecards, Metadata models, Automated Tasks and Events, Mobile Business Intelligence, Software development kit (SDK).

UNIT-IV
Building BI Project

UNIT-V
Report Authoring
Building Reports, Building a Report, Drill-up, Drill-down Capabilities.

9 Hours

BI Deployment, Administration and Security

Centralized versus Decentralized Architecture, Phased and Incremental BI road map, Setting early expectations and measuring the results, EPM (Enterprise performance Management), End-User Provisos, OLAP Implementation, Implementation, Data Warehouse Architecture, Predictive Analysis, Text Mining, Authentication, Authorization, Access Permissions, Group and Roles, Single Sign-on (SSO), Data Backup and Restoring.

TEXT BOOK-
• Business Intelligence (IBM ICE Publication)

*******************

BUSINESS INTELLIGENCE [COGNOS BI] (BAO) LAB

Sub Code : 13CS629 Credits : 01
Hrs/Week : 02 Total Hours: 39

Business Intelligence Lab Exercises :

- Overview of BI Tool – Cognos Report Studio
- Authoring Reports
- List, Crosstab and Chart Reports
- Grouping and Summarizing data
- Filter, Sort and Calculation

*******************

EMPLOYABILITY SKILL DEVELOPMENT
UNIT - I

Analytical Aptitude Skill: concept of analytical skill, definition-logical thinking and testing of Analytical Aptitude

UNIT - II

Quantitative Aptitude skill-Concept-definition-Preliminary requirement for development of quantitative skill- testing of quantitative skill.

UNIT - III

Verbal and ability skill – Knowledge and Vocabulary and grammar-comprehension-Verbal Reasoning skill

REFERENCE BOOKS:
**Examination pattern:**
This course is a mandatory learning course without credit. Continuous internal examination (CIE) consists of 2 internal exams (20 marks each) and tasks (10 marks). There is no semester end examination (SEE). The student will be awarded PP or NP grade as per autonomous regulations.