B.Sc. INFORMATION TECHNOLOGY
CHOICE BASED CREDIT SYSTEM (CBCS)

Syllabus
2017-2018

DEPARTMENT OF COMPUTER SCIENCE,
APPLICATIONS & INFORMATION TECHNOLOGY
FACULTY OF ARTS, SCIENCE AND HUMANITIES

KARPAGAM ACADEMY OF HIGHER EDUCATION
(Deemed to be University)
(Established under section 3 of UGC Act, 1956)
Pollachi Main Road, Eachanari (Post), Coimbatore- 641021, Tamil Nadu, India
Phone: 0422 – 2980011 – 15 Fax No: 0422 – 2980022-23
Email: info@karpagam.com Web: www.kahedu.edu.in
பத்தி - I காற்று பாதுகாப்பில் (2016-2017)

16LAU101: கூடுதல் காற்று 4-H,4-C (இந்தியில் ஆறாவது பிறகு நடவடிக்கை)

அப்து - I : தகவல் நேரடையில்: (10 மணிக்கோட்டை)

குறுக்கு : செயல்பாடு பார்வையளிதல் - வழக்கியாக - பார்வையளிதல் கலைக்குடும்ப.
இல்லாது விளக்கம்: கூடுதல் பார்வை விளக்கம் பார்வையளிதல் - குறுக்குகள் இல்லாது விளக்கம்.

மாணவியல்: கூடுதல் பார்வையளிதல் - வழக்கியாக - பார்வையளிதல்

வேளாண்மை: கூடுதல் காற்று பாதுகாப்பில் - வழக்கியாக - பாதுகாப்பில்

அப்து - II : அழுத் திட்டமிட்டம்: (10 மணிக்கோட்டை)

காற்று பாதுகாப்பு நேரடையில்: 1-50 மணிக்கோட்டை

தவிர்ந்து: மறுபாதுகாப்பு, பாதுகாப்புப்பாதுகாப்பு - 20 மணிக்கோட்டை

முடிக்கும் நேரடை: 5 மணிக்கோட்டை

அப்து - III : திறன்மிட்டம்: (10 மணிக்கோட்டை)

சுருக்கம்: 1-26 மணிக்கோட்டை

தவிர்ந்து: பாதுகாப்பு - 2 மணிக்கோட்டை

முடிக்கும் நேரடை: 9 மணிக்கோட்டை

அப்து - IV : செயல்பாடு: (10 மணிக்கோட்டை)

1. கூடுதல் ஆறாவது பாதுகாப்பு - வழக்கியாக
2. கற்றுவாக்கு - அ. இல்லாது கருத்துறுப்புக் காற்று
3. மாணவியல் - இல்லாது கருத்துறுப்பு
4. ஆறாவது ஆறாவது ஆறாவது - மாணவியல்
5. பாதுகாப்பு - இல்லாது கருத்துறுப்பு

அப்து - V : விளக்கங்கள்: (8 மணிக்கோட்டை)

1. பாயுமறையில் மாணவியல் (செயல்பாடு, மாணவியல், தவிர்ந்து, சுருக்கம்)
2. டெக்னோஜிகல்
3. மேலும் பார்வையளிதல்

மேலும்: கற்றுவாக்கு - காற்று வழ. செயல்பாடு பாதுகாப்பில் மாணவியல் வழக்கியாக

SCOPE: C is a general-purpose programming language. It is designed for developing system software, portable application software. Despite its low-level capabilities, the language was designed to encourage cross-platform programming. Also this course introduces the concepts of Object Oriented Programming language.

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

UNIT-I
Introduction to C and C++:
History of C and C++, Overview of Procedural Programming and Object-Orientation Programming, Using main() function, Compiling and Executing Simple Programs in C++.

Data Types, Variables, Constants, Operators and Basic I/O: Declaration, Defining and Initializing Variables, Scope of Variables, Using Named Constants, Keywords, Data Types, Casting of Data Types, Operators, Using Comments in programs, Character I/O, Formatted and Console I/O, Using Basic Header Files.


UNIT-II
Functions and Arrays: Utility of functions, Call by Value, Call by Reference, Functions returning value, Void functions, Inline Functions, Return data type of functions, Functions parameters, Differentiating between Declaration and Definition of Functions, Command Line Arguments/Parameters in Functions, Functions with variable number of Arguments. Creating and Using One Dimensional Arrays, Use Various types of arrays, Two-dimensional Arrays, Introduction to Multi-dimensional arrays.
UNIT-III
Derived Data Types (Structures and Unions): Understanding utility of structures and unions, Declaring, initializing and using simple structures and unions, Manipulating individual members of structures and unions, Array of Structures, Individual data members as structures, Passing and returning structures from functions, Structure with union as members, Union with structures as members.

Pointers and References in C++: Understanding a Pointer Variable, Simple use of Pointers, Pointers to Pointers, Pointers to structures, Passing pointers as function arguments, Returning a pointer from a function, using arrays as pointers, Passing arrays to functions.

UNIT-IV
File I/O, Preprocessor Directives: Opening and closing a file, Reading and writing Text Files, Using put(), get(), read() and write() functions, Random access in files, Understanding the Preprocessor Directives, Macros.

UNIT-V

Inheritance, Polymorphism and Exception Handling: Introduction to Inheritance, Polymorphism, Basics Exceptional Handling.

Suggested Readings:


WEB SITES

2. http://www2.its.strath.ac.uk/courses/c/
5. www.cplusplus.com/
SCOPE

Computer System Architecture deals with the architecture of computer systems with its various processing units and also the performance measurement of the computer system. This course is designed to provide a comprehensive introduction to digital logic design leading to the ability to understand number system representations, binary codes, binary arithmetic and Boolean algebra, and its relevance to digital logic design.

OBJECTIVE

- To enable the students to gain knowledge on the architecture of modern computer.
- To understand how computer stores positive and negative numbers and to perform arithmetic operation of positive and negative numbers.
- To learn Cache memory and its importance

UNIT -I Introduction

Logic gates, Boolean algebra, circuit simplification, combinational circuits: Adders and Subtractors –Multiplexers and De multiplexers – Encoders and Decoders- sequential circuits: Flip Flop’s, registers, counters and memory units.

UNIT -II Data Representation and Basic Computer Arithmetic

Number systems, complements, fixed and floating point representation, character representation, addition, subtraction, magnitude comparison, multiplication and division algorithms for integers

UNIT –III Basic Computer Organization and Design

Computer registers, bus system, instruction set, timing and control, instruction cycle, memory reference, input-output and interrupt, Interconnection Structures, Bus Interconnection design of basic computer.

UNIT-IV Central Processing Unit

Register organization, arithmetic and logical micro-operations, stack organization, micro programmed control. Instruction formats, addressing modes, instruction codes, machine language, assembly language, input output programming, RISC, CISC architectures, pipelining and parallel architecture.

UNIT –V Memory and Input-Output Organization

Suggested Readings:

SCOPE
This course provides student with a comprehensive study of the fundamentals of computers.

OBJECTIVE
- To know the basic concept of computers
- To understand the concept of flowcharts
- To able to understand the programming logic concepts

UNIT – I

UNIT – II

UNIT – III
Flowcharts: Types of Flowcharts, Advantage and Disadvantage of Flowchart. Pseudocode: Definition and Its Characteristics Control Statements Basics of Programming Language: Usage of Character Set, Meaning of Keywords and Identifiers, Role of Data Types, Constants and Variables.

UNIT – IV
Importance of Casting, Different Types of Operators and their Precedence, Expressions, Conditional Statements (One-Way, Two-Way and Multi-Way Conditional), Looping Statements (For, While, do-while), Usage of Exit, continue, Break and Goto Statement.

UNIT – V
Arrays: Arrays, One dimensional array, Various operatio on Array (Inserting of Elements, Deleting of Element, Rotating List, Sorting, Searching, Merging Etc) and Two dimensional arrays (Matrix Addition, Transpose of Matrix, Matrix Multiplication), Modular programming and its features.
Suggested Readings:

1. WAP to display Fibonacci series (i) using recursion, (ii) using iteration

2. WAP to compute the sum of the first n terms of the following series
   \[ S = 1 + 1/2 + 1/3 + 1/4 + \ldots \]

3. Write a program to find whether a given no. is prime or not. Use the same to generate the prime numbers less than 100.

4. WAP to print the sum and product of digits of an integer.

5. WAP to reverse a number.

6. Write a program that checks whether a given string is Palindrome or not.

7. WAP to print a triangle of stars as follows (take number of lines from user):

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

8. WAP to perform following actions on an array entered by the user:
   i) Calculate and print the sum and average of the elements of array
   ii) Print the maximum and minimum element of array

9. Write a program that swaps two numbers using pointers.

10. Write a program which takes the radius of a circle as input from the user, passes it to another function that computes the area and the circumference of the circle and displays the value of area and circumference from the main() function.

11. Write a program to perform operations on strings.
12. Write a program to perform following Matrix operations (2-D array implementation):
a) Sum b) Difference c) Product

13. Create a structure Student containing fields for Roll No., Name, Class, Year and Total Marks. Create 10 students and store them in a file.

14. Create the Person class. Create some objects of this class (by taking information from the user). Inherit the class Person to create two classes Teacher and Student class. Maintain the respective information in the classes and create, display and delete objects of these two classes (Use Runtime Polymorphism).

15. Copy the contents of one text file to another file using command line argument.
Semester – I

17ITU112 COMPUTER SYSTEM ARCHITECTURE - PRACTICAL  3H – 2C

Instruction Hours / week: L: 0 T: 0 P: 3  Marks:  Int : 40 Ext : 60  Total: 100

1. Verification of Logic Gates
2. Code converters
3. Realization of Multiplexer using basic gates
4. Encoder and Decoder
5. Realization Half and Full adders
6. Realization of Subtractor
7. Realization of Parity generator
8. Flip-Flop Circuits
9. Digital to analog Converters
10. Demonstrate a Basic Arithmetic Computing operations
Semester – I

17ITU113 OFFICE AUTOMATION - PRACTICAL 4H – 2C

| Instruction Hours / week: L: 0 T: 0 P: 4 | Marks: Int : 40 Ext : 60 | Total: 100 |

Practical exercises based on MS Office/ Open Office tools using document preparation and spreadsheet handling packages.

**MS Word**

1. Prepare a *grocery list* having four columns (Serial number, The name of the product, quantity and price) for the month of April, 06.
   - Font specifications for Title (Grocery List): 14-point Arial font in bold and italics.
   - The headings of the columns should be in 12-point and bold.
   - The rest of the document should be in 10-point Times New Roman.
   - Leave a gap of 12-points after the title.

2. Create a *telephone directory*.
   - The heading should be 16-point Arial Font in bold
   - The rest of the document should use 10-point font size
   - Other headings should use 10-point Courier New Font.
   - The footer should show the page number as well as the date last updated.

3. Design a *time-table form* for your college.
   - The first line should mention the name of the college in 16-point Arial Font and should be bold.
   - The second line should give the course name/teacher’s name and the department in 14-point Arial.
   - Leave a gap of 12-points.
   - The rest of the document should use 10-point Times New Roman font.
   - The footer should contain your specifications as the designer and date of creation.

4. BPB Publications plans to release a new book designed as per your syllabus. Design the *first page of the book* as per the given specifications.
   - The title of the book should appear in bold using 20-point Arial font.
   - The name of the author and his qualifications should be in the center of the page in 16-point Arial font.
   - At the bottom of the document should be the name of the publisher and address in 16-point Times New Roman.
   - The details of the offices of the publisher (only location) should appear in the footer.
5. Create the following one page documents.
   a. Compose a note inviting friends to a get-together at your house, Including a list of things to bring with them.
   b. Design a certificate in landscape orientation with a border around the document.
   c. Design a Garage Sale sign.
   d. Make a sign outlining your rules for your bedroom at home, using a numbered list.

6. Create the following documents:
   (a) A newsletter with a headline and 2 columns in portrait orientation, including at least one image surrounded by text.
   (b) Use a newsletter format to promote upcoming projects or events in your classroom or college.

7. Convert following text to a table, using comma as delimiter
   Type the following as shown (do not bold).
<table>
<thead>
<tr>
<th>Color, Style, Item</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blue, A980, Van</td>
</tr>
<tr>
<td>Red, X023, Car</td>
</tr>
<tr>
<td>Green, YL724, Truck</td>
</tr>
<tr>
<td>Name, Age, Sex</td>
</tr>
<tr>
<td>Bob, 23, M</td>
</tr>
<tr>
<td>Linda, 46, F</td>
</tr>
<tr>
<td>Tom, 29, M</td>
</tr>
</tbody>
</table>

8. Enter the following data into a table given on the next page.

<table>
<thead>
<tr>
<th>Salesperson</th>
<th>Dolls</th>
<th>Trucks</th>
<th>Puzzles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kennedy, Sally</td>
<td>1327</td>
<td>1423</td>
<td>1193</td>
</tr>
<tr>
<td>White, Pete</td>
<td>1421</td>
<td>3863</td>
<td>2934</td>
</tr>
<tr>
<td>Pillar, James</td>
<td>5214</td>
<td>3247</td>
<td>5467</td>
</tr>
<tr>
<td>York, George</td>
<td>2190</td>
<td>1278</td>
<td>1928</td>
</tr>
<tr>
<td>Banks, Jennifer</td>
<td>1201</td>
<td>2528</td>
<td>1203</td>
</tr>
<tr>
<td>Atwater, Kelly</td>
<td>4098</td>
<td>3079</td>
<td>2067</td>
</tr>
<tr>
<td>Pillar, James</td>
<td>5214</td>
<td>3247</td>
<td>5467</td>
</tr>
<tr>
<td>York, George</td>
<td>2190</td>
<td>1278</td>
<td>1928</td>
</tr>
<tr>
<td>Banks, Jennifer</td>
<td>1201</td>
<td>2528</td>
<td>1203</td>
</tr>
<tr>
<td>Atwater, Kelly</td>
<td>4098</td>
<td>3079</td>
<td>2067</td>
</tr>
</tbody>
</table>

Add a column Region (values: S, N, N.S,S,S) between the Salesperson and Dolls columns to the given table Sort your table data by Region and within Region by Salesperson in ascending order:
In this exercise, you will add a new row to your table, place the word Total at the bottom of the Salesperson column, and sum the Dolls, Trucks, and Puzzles columns.
9. Wrapping of text around the image.

10. Following features of menu option must be covered

   FILE Complete menu
   EDIT Complete menu
   VIEW Complete menu
   INSERT Complete menu
   FORMAT Complete menu
   TABLE Complete menu
   WINDOW Complete menu
   HELP Complete menu
   TOOLS All options except Online collaboration, Tools on Macro, Templates

**MS Excel**

1. Enter the Following data in Excel Sheet

<table>
<thead>
<tr>
<th>State</th>
<th>Qtr1</th>
<th>Qtr2</th>
<th>Qtr3</th>
<th>QTR4</th>
<th>Qtr Total</th>
<th>Rate</th>
<th>Amount</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delhi</td>
<td>2020</td>
<td>2400</td>
<td>2100</td>
<td>3000</td>
<td>15</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Punjab</td>
<td>1100</td>
<td>1300</td>
<td>1500</td>
<td>1400</td>
<td>20</td>
<td></td>
<td></td>
</tr>
<tr>
<td>U.P.</td>
<td>3000</td>
<td>3200</td>
<td>2600</td>
<td>2800</td>
<td>17</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Haryana</td>
<td>1800</td>
<td>2000</td>
<td>2200</td>
<td>2700</td>
<td>15</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rajasthan</td>
<td>2100</td>
<td>2000</td>
<td>1800</td>
<td>2200</td>
<td>20</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**TOTAL**

**AVERAGE**

(a) Apply Formatting as follow:

i. Title in TIMES NEW ROMAN

ii. Font Size - 14

iii. Remaining text - ARIAL, Font Size -10

iv. State names and Qtr. Heading Bold, Italic with Gray Fill Color.

v. Numbers in two decimal places.

vi. Qtr. Heading in center Alignment.

vii. Apply Border to whole data.

(b) Calculate State and Qtr. Total
(c) Calculate Average for each quarter

(d) Calculate Amount = Rate * Total.

2. Given the following worksheet

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Roll No.</td>
<td>Name</td>
<td>Marks</td>
</tr>
<tr>
<td>2</td>
<td>1001</td>
<td>Sachin</td>
<td>99</td>
</tr>
<tr>
<td>3</td>
<td>1002</td>
<td>Sehwag</td>
<td>65</td>
</tr>
<tr>
<td>4</td>
<td>1003</td>
<td>Rahul</td>
<td>41</td>
</tr>
<tr>
<td>5</td>
<td>1004</td>
<td>Sourav</td>
<td>89</td>
</tr>
<tr>
<td>6</td>
<td>1005</td>
<td>Har Bhajan</td>
<td>56</td>
</tr>
</tbody>
</table>

Calculate the grade of these students on the basis of following guidelines:

<table>
<thead>
<tr>
<th>If Marks</th>
<th>Then Grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt;= 80</td>
<td>A+</td>
</tr>
<tr>
<td>&gt;= 60 &lt; 80</td>
<td>A</td>
</tr>
<tr>
<td>&gt;= 50 &lt; 60</td>
<td>B</td>
</tr>
<tr>
<td>&lt; 50</td>
<td>F</td>
</tr>
</tbody>
</table>

3. Given the following worksheet

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Salesman</td>
<td>Sales in (Rs.)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>No.</td>
<td>Qtr1</td>
<td>Qtr2</td>
<td>Qtr3</td>
<td>Qtr4</td>
</tr>
<tr>
<td>3</td>
<td>S001</td>
<td>5000</td>
<td>8500</td>
<td>12000</td>
<td>9000</td>
</tr>
<tr>
<td>4</td>
<td>S002</td>
<td>7000</td>
<td>4000</td>
<td>7500</td>
<td>11000</td>
</tr>
<tr>
<td>5</td>
<td>S003</td>
<td>4000</td>
<td>9000</td>
<td>6500</td>
<td>8200</td>
</tr>
<tr>
<td>6</td>
<td>S004</td>
<td>5500</td>
<td>6900</td>
<td>4500</td>
<td>10500</td>
</tr>
<tr>
<td>7</td>
<td>S005</td>
<td>7400</td>
<td>8500</td>
<td>9200</td>
<td>8300</td>
</tr>
<tr>
<td>8</td>
<td>S006</td>
<td>5300</td>
<td>7600</td>
<td>9800</td>
<td>6100</td>
</tr>
</tbody>
</table>
Calculate the commission earned by the salesmen on the basis of following Candidates:

<table>
<thead>
<tr>
<th>If Total Sales</th>
<th>Commission</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt; 20000</td>
<td>0% of sales</td>
</tr>
<tr>
<td>&gt; 20000 and &lt; 25000</td>
<td>4% of sales</td>
</tr>
<tr>
<td>&gt; 25000 and &lt; 30000</td>
<td>5.5% of sales</td>
</tr>
<tr>
<td>&gt; 30000 and &lt; 35000</td>
<td>8% of sales</td>
</tr>
<tr>
<td>&gt;= 35000</td>
<td>11% of sales</td>
</tr>
</tbody>
</table>

The total sales is sum of sales of all the four quarters.

4. A company XYZ Ltd. pays a monthly salary to its employees which consists of basic salary, allowances & deductions. The details of allowances and deductions are as follows:

**Allowances**
- HRA Dependent on Basic
  - 30% of Basic if Basic <=1000
  - 25% of Basic if Basic >1000 & Basic<=3000
  - 20% of Basic if Basic >3000
- DA Fixed for all employees, 30% of Basic
- Conveyance Allowance Rs. 50/- if Basic is <=1000
  - Rs. 75/- if Basic >1000 & Basic<=2000
  - Rs. 100 if Basic >2000
- Entertainment Allowance NIL if Basic is <=1000
  - Rs. 100/- if Basic > 1000

**Deductions**
- Provident Fund 6% of Basic
- Group Insurance Premium Rs. 40/- if Basic is <=1500
  - Rs. 60/- if Basic > 1500 & Basic<=3000
  - Rs. 80/- if Basic >3000

Calculate the following:
Gross Salary = Basic + HRA + DA + Conveyance + Entertainment
Total deduction = Provident Fund + Group Insurance Premium
Net Salary = Gross Salary – Total Deduction.

5. Create Payment Table for a fixed Principal amount, variable rate of interest and time in the format below:

<table>
<thead>
<tr>
<th>No. of Installments</th>
<th>5%</th>
<th>6%</th>
<th>7%</th>
<th>8%</th>
<th>9%</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>XXXX</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
</tr>
<tr>
<td>4</td>
<td>XXXX</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
</tr>
<tr>
<td>5</td>
<td>XXXX</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
</tr>
<tr>
<td>6</td>
<td>XXXX</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
<td>XXX</td>
</tr>
</tbody>
</table>

6. Use an array formula to calculate Simple Interest for given principal amounts given the rate of
6. Use an array formula to calculate Simple Interest for given principal amount. Given the Rate of Interest and time.

<table>
<thead>
<tr>
<th>Rate of Interest</th>
<th>8%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time</td>
<td>5 Years</td>
</tr>
<tr>
<td>Principal</td>
<td>Simple Interest</td>
</tr>
<tr>
<td>1000</td>
<td>?</td>
</tr>
<tr>
<td>18000</td>
<td>?</td>
</tr>
<tr>
<td>5200</td>
<td>?</td>
</tr>
</tbody>
</table>

7. The following table gives year wise sale figure of five salesmen in Rs.

<table>
<thead>
<tr>
<th>Salesman</th>
<th>2000</th>
<th>2001</th>
<th>2002</th>
<th>2003</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1</td>
<td>10000</td>
<td>12000</td>
<td>20000</td>
<td>50000</td>
</tr>
<tr>
<td>S2</td>
<td>15000</td>
<td>18000</td>
<td>50000</td>
<td>60000</td>
</tr>
<tr>
<td>S3</td>
<td>20000</td>
<td>22000</td>
<td>70000</td>
<td>70000</td>
</tr>
<tr>
<td>S4</td>
<td>30000</td>
<td>30000</td>
<td>100000</td>
<td>80000</td>
</tr>
<tr>
<td>S5</td>
<td>40000</td>
<td>45000</td>
<td>125000</td>
<td>90000</td>
</tr>
</tbody>
</table>

(a) Calculate total sale year wise.
(b) Calculate the net sale made by each salesman
(c) Calculate the maximum sale made by the salesman
(d) Calculate the commission for each salesman under the condition.
   (i) If total sales > 4,00,000 give 5% commission on total sale made by the salesman.
   (ii) Otherwise give 2% commission.
(e) Draw a bar graph representing the sale made by each salesman.
(f) Draw a pie graph representing the sale made by salesman in 2000.

8. Enter the following data in Excel Sheet

**PERSONAL BUDGET FOR FIRST QUARTER**

**Monthly Income (Net): 1,475**

<table>
<thead>
<tr>
<th>EXPENSES</th>
<th>JAN</th>
<th>FEB</th>
<th>MARCH</th>
<th>QUARTER TOTAL</th>
<th>QUARTER AVERAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rent</td>
<td>600.00</td>
<td>600.00</td>
<td>600.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Telephone</td>
<td>48.25</td>
<td>43.50</td>
<td>60.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Utilities</td>
<td>67.27</td>
<td>110.00</td>
<td>70.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Credit Card</td>
<td>200.00</td>
<td>110.00</td>
<td>70.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Oil</td>
<td>100.00</td>
<td>150.00</td>
<td>90.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AV to Insurance</td>
<td>150.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cable TV</td>
<td>40.75</td>
<td>40.75</td>
<td>40.75</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Monthly Total

Calculate Quarter total and Quarter average.
(a) Calculate Monthly total.
(b) Surplus = Monthly income - Monthly total.
(c) What would be total surplus if monthly income is 1500.
(d) How much does telephone expense for March differ from quarter average.
(e) Create a 3D column graph for telephone and utilities. (f) Create a pie chart for monthly expenses.

9. Enter the following data in Excel Sheet

TOTAL REVENUE EARNED FOR SAM’S BOOKSTALL

<table>
<thead>
<tr>
<th>Publisher name</th>
<th>1997</th>
<th>1998</th>
<th>1999</th>
<th>2000</th>
<th>total</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Rs.1000.00</td>
<td>Rs.1100.00</td>
<td>Rs.1300.00</td>
<td>Rs.800.00</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>Rs.1500.00</td>
<td>Rs.700.00</td>
<td>Rs.1000.00</td>
<td>Rs.2000.00</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>Rs.700.00</td>
<td>Rs.900.00</td>
<td>Rs.1500.00</td>
<td>Rs.600.00</td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>Rs.1200.00</td>
<td>Rs.500.00</td>
<td>Rs.200.00</td>
<td>Rs.1100.00</td>
<td></td>
</tr>
<tr>
<td>E</td>
<td>Rs.800.00</td>
<td>Rs.1000.00</td>
<td>Rs.3000.00</td>
<td>Rs.560.00</td>
<td></td>
</tr>
</tbody>
</table>

(a) Compute the total revenue earned.
(b) Plot the line chart to compare the revenue of all publisher for 4 years.
(c) Chart Title should be "Total Revenue of sam’s Bookstall (1997-2000)"
(d) Give appropriate categories and value axis title.

10. Generate 25 random numbers between 0 & 100 and find their sum, average and count. How many no. are in range 50-60.
SCOPE

The study creates awareness among the people to know about various renewable and nonrenewable resources of the region, enables environmentally literate citizens (by knowing the environmental acts, rights, rules, legislation, etc.) to make appropriate judgments and decisions for the protection and improvement of the earth.

OBJECTIVES

- Creating the awareness about environmental problems among people.
- Developing an attitude of concern for the environment.
- Motivating public to participate in environment protection and improvement.


Unit II: Natural Resources - Renewable and Non-renewable Resources: Natural resources and associated problems. Forest resources, Water resources, Mineral resources, Food resources, Energy resources, Land resources: Use and over-utilization, exploitation. Role of an individual in conservation of natural resources. Equitable use of resources for sustainable lifestyles. Ill-effects of fire works.


Unit IV: Environmental Pollution - Definition, Causes, effects and control measures of Air pollution, Water pollution, Soil pollution, Marine pollution, Noise pollution, Thermal pollution, Nuclear hazards, Solid waste management: Causes, effects and control measures of urban and industrial wastes. Role of an individual in prevention of pollution. Pollution case studies. Disaster management: Foods, earthquake, cyclone and landslides.


**Suggested Readings**


பதிப்பு - 1 கல்வி மாணவி (2016 - 2017)

16LAU201 : கல்வி 2 மாணவி கல்வி
(இன்றிய அறிவியல் பட்ட முறைகளவுற்று)

அண்ணல் - I : பட்ட விளக்கம்
(10 மாணிக்குறிகளில்)

வெளிப்பாடு, வணிகம் விளக்கம் - அறிவுப் பொருள், வணிகம், மற்றும்:
1. வணிகம் - விளக்க வோட்டம், முதலாமார் வோட்டம்.
2. வணிகம் - விளக்க வோட்டம், தொடர்புப் பொருள்: 10 மாணிக்குறிகள்.

அண்ணல் - II : கல்வி விளக்கம்
(15 மாணிக்குறிகளில்)

எள்ளத்துறை விளக்கம், தொடர்புப் பொருள்

குறிப்பிட்டு

முதலில் : பேரும் எள்ளத் - 110

அறிவியல்: கணினியியல் கல்வி - ஆர்த்திச்-
69

கல்வி விளக்கத்தில்:

ஏனெனில்: பட்டத்துறை-
171

பேரும் பற்றி: விளக்கத்துறை, தொடர்பு - 27

வணிகம்: வணிகம் கணினியியல் தொடர்புத் துறைப்பின் -
6, வணிகம் தொடர்பு-7, வணிகம் தொடர்பு-
9, வணிகம் தொடர்பு-10, வணிகம் தொடர்பு-11.

ஆர்த்திச்: ஆர்த்திச் விளக்கம் விளக்கத்துறை-36

அறிவியல்: அறிவியல் விளக்கம் விளக்கம் - ஆர்த்திச்-48

பேரும்: பேரும் பற்றி, பேரும் தொடர்பு-பேரும்-192
உ.  புதுப்பிட்டியல்

முதுகுருகங்காணங்கள் - புதுப்பிட்டியலியின் சிறப்பு

முக்குச் சிறப்புகள் - 'சித்ரநச்சிய சாகுர்காதல்' எழுந்தொடரளி, 'இன்னுறுத்தல்' எழுந்தோடு மோதலியல் விளக்களள் 218-249.

முக்குச் சிறப்புகள் - 'சித்ரநச்சிய சாகுர்காதல்' எழுந்தோடு மோதலியல் விளக்களள் 286-295.

உண்டு - III : காப்பியம் (6 முறைகளும்)

சிறப்புகள்:

முக்குச் சிறப்புகள் முறை: (21-29) - காப்பியமும் சிறப்பு:

'சாகுர்காதல்' எழுந்தோடு மோதலியல், 'சாகுர்காதல் மோதலியல்' எழுந்தோடு மோதலியல் விளக்களள்.

குறிப்பிட்டுடன் (207-234) - வெளிய விளக்களள் முக்குச் சிறப்புகள் காப்பியமும் சிறப்பு:

'சாகுர்காதல்' எழுந்தோடு மோதலியல், 'சாகுர்காதல் மோதலியல்' எழுந்தோடு மோதலியல் விளக்களள்.

முக்குச் சிறப்புகள் முறை: (482-485) - வெளிய விளக்களள் காப்பியமும் சிறப்பு:

'சாகுர்காதல்' எழுந்தோடு மோதலியல், 'சாகுர்காதல் மோதலியல்' எழுந்தோடு மோதலியல் விளக்களள்.

முக்குச் சிறப்புகள் முறை: புதுப்பிட்டியலியின் முழுமையில் சிறப்பு:

'சாகுர்காதல்' எழுந்தோடு மோதலியல், 'சாகுர்காதல் மோதலியல்' எழுந்தோடு மோதலியல் விளக்களள்.

முக்குச் சிறப்புகள்: 'சாகுர்காதல்' எழுந்தோடு மோதலியல், 'சாகுர்காதல் மோதலியல்' எழுந்தோடு மோதலியல் விளக்களள்.

உண்டு - IV : துறைகள் (10 முறைகளும்)

1. மொழிபொருள் ஆய்வுக் - மொழி. அல்லது
2. குறிப்பிட்டு - வகுப்பு
3. துறைகள் - துறைணைப்புணர்வு
4. துறைகள் - துறைணை
அலுவல்- V: ியேறிப்புமறைந்தி

(7 மத்தியிருந்து)

பல்லவியல் நுழைவு வருடம் (மேல், கல்வியுடன், நூற்றாண்டு முறையில்)

இயேறிப்புமறைந்தி

பல்லவியல் நுழைவு வருடம் - குறிப்பிட்டு. குறிப்பிட்டு நுழைவு வருடம் நூற்றாண்டு முறையில் இயேறிப்புமறைந்தி.
### Semester – II

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>17ENU201</td>
<td>ENGLISH</td>
<td>4H – 4C</td>
</tr>
</tbody>
</table>

**Instruction Hours / week:** L: 4 T: 0 P: 0  
Marks: Int : 40 Ext : 60  
Total: 100

**Objectives:** To train students in acquiring proficiency in English by reading different genres in literature and learning grammar  
To provide aesthetic pleasure through literature

#### UNIT - I : PROSE
1. Morals in the Indian Context - Francis Nicholas Chelliah  
2. How Comic Books help us to relive our Childhood - Benoit Peeters  
3. Let’s Do What India Needs From Us - Dr.A.P.J. Abdul Kalam

#### UNIT - II : POEM
1. The Stolen Boat - William Wordsworth  
3. The Sailor - Safaa Fathy

#### UNIT - III : SHORT STORIES
1. Rapunzel - Brothers Grimm  
2. The Romance of a Busy Broker - O.Henry  
3. The Nightingale and the Rose - Oscar Wilde.

#### UNIT - IV
**GRAMMAR**  
1. Tenses  
2. Auxiliaries (Primary and Modal)  
3. Articles  
4. Tag Questions

#### UNIT - V
**FUNCTIONAL ENGLISH**
1. Filling the blanks with the suitable form of verb in a conditional sentence.  
2. Dialogue Writing  
3. Changing positive to negative without altering the meaning  
4. Fill in the blank with suitable modal  
5. Framing a question to a statement  
6. Rewrite the sentences changing the underlined word as directed

**Prescribed Text:** Reminisce, Published by the Department of English, Karpagam University.

**Suggested Reading:** Hewings Martin, 1999 Advanced English Grammar, Cambridge University Press.
SCOPE
This course offers an introduction to the Java programming language. This course covers the basic topics considered are programs and program structure in general, and Java syntax, data types, flow of control, classes, methods, objects, arrays, exception handling, recursion, and graphical user interfaces (GUIs).

OBJECTIVES
- know the java path setting and programming techniques
- basic java programming and Applet programming
- understand the fundamental of Packages and aITUess modifiers and interface in java
- understand the fundamental of Exception Handling and AWT component and AWT classes

UNIT-I
Introduction to Java  Java Architecture and Features, Understanding the semantic and syntax differences between C++ and Java, Compiling and Executing a Java Program, Variables, Constants, Keywords Data Types, Operators (Arithmetic, Logical and Bitwise) and Expressions, Comments, Doing Basic Program Output, Decision Making Constructs (conditional statements and loops) and Nesting, Java Methods (Defining, Scope, Passing and Returning Arguments, Type Conversion and Type and Checking, Built-in Java Class Methods),

UNIT-II
Arrays, Strings and I/O Creating & Using Arrays (One Dimension and Multi-dimensional), Referencing Arrays Dynamically, Java Strings: The Java String class, Creating & Using String Objects, Manipulating Strings, String Immutability & Equality, Passing Strings To & From Methods, String Buffer Classes. Simple I/O using System.out and the Scanner class, Byte and Character streams, Reading/Writing from console and files. Object-Oriented Programming Overview Principles of Object-Oriented Programming, Defining & Using Classes, Controlling aITUess to Class Members, Class Constructors, Method Overloading, Class Variables & Methods, Objects as parameters, final classes, Object class, Garbage Collection.

UNIT-III
Inheritance, Interfaces, Packages, Enumerations, Autoboxing and Metadata
Inheritance: (Single Level and Multilevel, Method Overriding, Dynamic Method Dispatch, Abstract Classes), Interfaces and Packages, Extending interfaces and packages, Package and Class Visibility, Using Standard Java Packages (util, lang, io, net), Wrapper Classes, Autoboxing/Unboxing, Enumerations and Metadata.
UNIT-IV
Exception Handling, Threading, Networking and Database Connectivity
Exception types, uncaught exceptions, throw, built-in exceptions, Creating your own exceptions; Multi-threading: The Thread class and Runnable interface, creating single and multiple threads, Thread prioritization, synchronization and communication, suspending/resuming threads. Using java.net package, Overview of TCP/IP and Datagram programming. Accessing and manipulating databases using JDBC.

UNIT-V
Java Applets: Introduction to Applets, Writing Java Applets, Working with Graphics, Incorporating Images & Sounds. Event Handling Mechanisms, Listener Interfaces, Adapter and Inner Classes. The design and Implementation of GUIs using the AWT controls, Swing components of Java Foundation Classes such as labels, buttons, textfields, layout managers, menus, events and listeners; Graphic objects for drawing figures such as lines, rectangles, ovals, using different fonts. Overview of servlets.

Suggested Readings:

WEB SITES
1. java.sun.com/docs/books/tutorial/
2. www.en.wikipedia.org/wiki/Java
3. www.java.net/
17ITU202  DISCRETE STRUCTURES  

Instruction Hours / week: L: 4 T: 0 P: 0  Marks: Int : 40 Ext : 60  Total: 100

SCOPE
It exposes the students to study entities such as sets, relations, graphs, and trees. These entities act as very fundamental representations useful in a broad spectrum of applications across the length and breadth of computer science.

OBJECTIVE
This course provides a deep knowledge to the learners to develop and analyze algorithms as well as enable them to think about and solve problems in new ways. By the completion of the course students should be able to express ideas using mathematical notation and solve problems using the tools of mathematical analysis.

UNIT I
Sets: Introduction, Sets, finite and infinite sets, uncountably infinite sets, functions, relations, properties of binary relations, closure, partial ordering relations, counting, Pigeonhole principle, Permutation and Combination, Mathematical Induction, Principle of inclusion and Exclusion.

UNIT II
Growth of Functions: Asymptotic Notations, Summation formulas and properties, Bounding Summations, approximation by Integrals

UNIT III
Recurrences: Recurrence relations, generating functions, linear recurrence relations with constant coefficients and their solution, Substitution Method, recurrence trees, Master theorem.

UNIT IV
Graph Theory: Basic terminology, models and types, multigraphs and weighted graphs, graph representation, graph isomorphism, connectivity, Euler and Hamiltonian Paths and circuits, Planar graphs, graph coloring, trees, basic terminology and properties of trees, introduction to Spanning trees

UNIT V
Prepositional Logic: Logical Connectives, Well-formed Formulas, Tautologies, Equivalences, Inference Theory.
Suggested Readings

17ITU203  FUNDAMENTALS OF WEB DESIGNING AND INTERNET

Instruction Hours / week: L: 4 T: 0 P: 0  Marks: Int : 40 Ext : 60  Total: 100

SCOPE
- To gain the skills and project-based experience needed for entry into web design and development careers.
- Develop awareness and appreciation of the many ways that people access the web, and will be able to create standards-based websites that can be accessed by the full spectrum of web access technologies.

OBJECTIVE
- This course in curriculum is an introduction to the design, creation, and maintenance of web pages and websites.
- This course enables students to understand web page site planning, management and maintenance
- How to critically evaluate website quality, learn how to create and maintain quality web pages learn to create and manipulate images.
- This course presents introductions to many of the basic concepts, issues and techniques related to designing, developing and deploying Web sites. During the course, students will learn about Web design, HTML, XML, ASP, VBScript, JavaScript and Cascading Style Sheets (CSS).

Unit – I: Introduction to Web Technology
Introduction to Internet, Basic Internet Terms in Internet Addressing, Internet Tools, Services of Internet, Introduction to Worl Wide Web, Components of Web, Types of Websites, Role of Web Browser and Web server, Types of Web Browsers, Types of Web Server, Flow of Web Inforamtion, Objective of the Website, Basic Interface design, Various Typcs of URLs, Process of Web Publishing.

Unit- II: Creating Static Web Pages and Lists in HTML
Introduction to HTML, Features of HTML, Advantages and Disadvantages of HTML, Features of Static and Dynamic Web Page. Creating a Simple Static Web Page: Creating Web page using HTML, structure of HTML document, HTML Tags, tags attributes, Basic Elements: <html>, <head>, <title>, <p>, <br>, <h1> to <h6>, <pre>, <marquee>, <hr>, <br>,<center>, Text formatting tags, Using colors for the web, Physical and Logical Tags, Special Characters.
Lists in HTML: Ordered List, Unordered List, Definition List and Nested List.

Unit-III: Adding Links, Images, Background and table
Tables: Components of a tabs, basic table tags and their related attributes. Paths: Relative Path and Absolute path, Linking HTML Pages: Link Tag <a href...>, kinds of linking, linking to URLs.
Unit-IV: Adding Images and Image Maps
Adding images to HTL pages: Image formats for Internet and HTML, Image tag and their related attributes, Inline images, Links to (external) images, images as hyperlinks, using images as background. Image Maps: What are image maps, tags used for image mapping, Client-side and server-side Image maps.

Unit-V: Forms, Frames and Embedding Multimedia
Frames: Introduction to Frame, <Frameset> and <frame> tag with its attributes, creating frames, linking frames, <noframes> tag, complex frameset, floating and inline frame. Forms: <Form> tag and its attributes,, <input> tag and its attributes, Form controls: Text controls, password fields, radio buttons, checkboxes, reset and submit buttons, Form control selection, Option processing and Text Area, Hidden fields. Embedding Multimedia: Introduction, Embedding Multimedia, inserting Sound/Audio Formats, Inserting video file formats.

Suggested Readings


Websites
1. www.w3schools.com/
2. www.htmlcodetutorial.com/
   Jmarshall.com/easy
1. To find the sum of any number of integers entered as command line arguments
2. To find the factorial of a given number
3. To learn use of single dimensional array by defining the array dynamically.
4. To learn use of .length in case of a two dimensional array
5. To convert a decimal to binary number
6. To check if a number is prime or not, by taking the number as input from the keyboard
7. To find the sum of any number of integers interactively, i.e., entering every number from the keyboard, whereas the total number of integers is given as a command line argument
8. Write a program that show working of different functions of String and StringBuffer classes like setCharAt, setLength, append, insert, concat and equals.
9. Write a program to create a —distance— class with methods where distance is computed in terms of feet and inches, how to create objects of a class and to see the use of this pointer
10. Modify the —distance— class by creating constructor for assigning values (feet and inches) to the distance object. Create another object and assign second object as reference variable to another object reference variable. Further create a third object which is a clone of the first object.
11. Write a program to show that during function overloading, if no matching argument is found, then java will apply automatic type conversions (from lower to higher data type)
12. Write a program to show the difference between public and private access specifiers. The program should also show that primitive data types are passed by value and objects are passed by reference and to learn use of final keyword
13. Write a program to show the use of static functions and to pass variable length arguments in a function.
14. Write a program to demonstrate the concept of boxing and unboxing.
15. Create a multi-file program where in one file a string message is taken as input from the user and the function to display the message on the screen is given in another file (make use of Scanner package in this program).
16. Write a program to create a multilevel package and also creates a reusable class to generate Fibonacci series, where the function to generate fibonacci series is given in a different file belonging to the same package.
17. Write a program that creates illustrates different levels of protection in classes/subclasses belonging to same package or different packages
18. Write a program —DivideByZero— that takes two numbers a and b as input, computes a/b, and invokes Arithmetic Exception to generate a message when the denominator is zero.
19. Write a program to show the use of nested try statements that emphasizes the sequence of checking for catch handler statements.
20. Write a program to create your own exception types to handle situation specific to your application (Hint: Define a subclass of Exception which itself is a subclass of Throwable).
21. Write a program to demonstrate priorities among multiple threads.
22. Write a program to demonstrate multithread communication by implementing synchronization among threads (Hint: you can implement a simple producer and consumer problem).
23. Write a program to create URL object, create a URLConnection using the openConnection() method and then use it examine the different components of the URL and content.
24. Write a program to implement a simple datagram client and server in which a message that is typed into the server window is sent to the client side where it is displayed.
25. Write a program that creates a Banner and then creates a thread to scrolls the message in the banner from left to right across the applet’s window.
26. Write a program to get the URL/location of code (i.e. java code) and document (i.e. html file).
27. Write a program to demonstrate different mouse handling events like mouseClicked(), mouseEntered(), mouseExited(), mousePressed, mouseReleased() and mouseDragged().
28. Write a program to demonstrate different keyboard handling events.
29. Write a program to generate a window without an applet window using main() function.
30. Write a program to demonstrate the use of push buttons.
1. Write a C Program to find the number of subsets of a set contains n elements.

2. Write a C Program to find transitive closure of a relation.

3. Write a C Program to prove
   \[
   \frac{1}{1\times2} + \frac{1}{2\times3} + \ldots + \frac{1}{n(n+1)} = \frac{n}{n+1}
   \]

4. Write a C Program to perform the sum \(= 1+ (1+2) + (1+2+3) + \ldots + (1+2\ldots+n)\)

5. Write a C program to print Fibonacci series till Nth term using recursion

6. Write a C program to calculate factorial of a number using recursion

7. Write a C Program to find a minimum spanning tree using Prim’s algorithm

8. Write a C program to find the shortest path with the lower cost in a graph using Dijkstra’s Algorithm

9. Write a C Program to construct the truth table for the following formula.
   (i) \(P \land Q \land \overline{R}\)   (ii) \(P \land \overline{Q} \land R\)   (iii) \(P \land Q \land \overline{R}\)

10. Write a C Program to prove De – Morgan’s law.
1. Create a simple HTML document about yourself or a topic of your choice using the basic tags such as Bold, Italic, Heading, images, horizontal rule and images.
2. Create a HTML document which includes an unordered list, ordered list, definition list to your document and create a link to Yahoo.
3. Develop static pages (using only HTML) of an online Book store. The pages should resemble: www.amazon.com. The website should consist of the following pages.
   - Home page
   - Registration and user login
   - User profile page
   - Items catalog
   - Shopping cart
   - Payment by credit card
   - Order confirmation
4. Create a table in HTML document with the following formats
   - Table with no borders
   - Display table headers
   - To handle cells that have no content
   - Table cells that span more than one row or one column
   - Adding background image to a table
5. Design a web page using CSS (Cascading Style Sheets) which includes the following:
   - Use different font, styles: In the style definition you define how each selector should work (font, color etc.). Then, in the body of your pages, you refer to these selectors to activate the styles.
   - Set a background image for both the page and single elements on the page.
   - Control the repetition of the image with the background-repeat property. As background-repeat:repeat Tiles the image until the entire page is filled, just like an ordinary background image in plain HTML.
   - Define styles for links as
     A:link
     A:visited
     A:active
     A:hover
   - Work with layers
   - Add a customized cursor: Selector {cursor:value}
6. Create and save an XML document at the server, which contains 10 users information. Write a program, which takes User Id as an input and returns the user details by taking the user information from the XML document.

7. Creation of a XML document of 20 students of III IT. Add their roll numbers, marks obtained in 5 subjects, total and percentage and save this XML document at the server. Write a program that takes students’ roll number as an input and returns the students marks, total and percentage by taking the students’ information from the XML document.

8. Write an XML file which will display the Book information which includes the following:
   - Title of the book
   - Author Name
   - ISBN number
   - Publisher name
   - Edition
   - Price

   Write a Document Type Definition (DTD) to validate the above XML file.

9. Develop a web page using Java script to perform the following information
   - Input student information
   - Display student results for a given roll number.

10. Develop a web page for online exam using Javascript

11. Add validations to the 3rd program of registration, user login, user profile and payment by creditr card using Java Script

12. Write a Java Script for selection sort

13. Create a web page using two image file which switch between one another as the mouse pointer moves over the image. Use the On Mouse Over and On Mouse event handler

14. Using VBScript to create a calculator

15. Create a HTML form that has number of text boxes when the form runs in the browser fill the textboxes with data. Write a Java script that verifies that all text boxes have been filled, if a text box has been left empty pop up an alert message indicating which box has been left empty. When Alerts OK button is clicked, set focus to that specific text box. If all the text boxes are filled, display thank you.
SCOPE
Data structures and algorithms are the building blocks in computer programming. This course will give students a comprehensive introduction of common data structures, and algorithm design and analysis. This course also intends to teach data structures and algorithms for solving real problems that arise frequently in computer applications, and to teach principles and techniques of computational complexity.

OBJECTIVES
• possess intermediate level problem solving and algorithm development skills on the computer
  • be able to analyze algorithms using big-Oh notation
  • understand the fundamental data structures such as lists, trees, and graphs
  • understand the fundamental algorithms such as searching, and sorting

UNIT-I
Arrays-Single and Multi-dimensional Arrays, Sparse Matrices (Array and Linked Representation); Stacks Implementing single / multiple stack/s in an Array; Prefix, Infix and Postfix expressions, Utility and conversion of these expressions from one to another; Applications of stack; Limitations of Array representation of stack

UNIT-II
Linked Lists Singly, Doubly and Circular Lists (Array and Linked representation); Normal and Circular, representation of Stack in Lists; Self Organizing Lists; Skip Lists Queues, Array and Linked representation of Queue, De-queue, Priority Queues

UNIT-III
Trees - Introduction to Tree as a data structure; Binary Trees (Insertion, Deletion , Recursive and Iterative Traversals on Binary Search Trees); Threaded Binary Trees (Insertion, Deletion, Traversals); Height-Balanced Trees (Various operations on AVL Trees).

UNIT-IV
Searching and Sorting,Linear Search, Binary Search, Comparison of Linear and Binary Search, Selection Sort, Insertion Sort, Insertion Sort, Shell Sort, Comparison of Sorting Techniques

UNIT-V
Hashing - Introduction to Hashing, Deleting from Hash Table, Efficiency of Rehash Methods, Hash Table Reordering, Resolving collusion by Open Addressing, Coalesced Hashing, Separate Chaining, Dynamic and Extendible Hashing, Choosing a Hash Function, Perfect Hashing, Function
Suggested Readings


Web Sites
http://www.cs.sunysb.edu/~skiena/214/lectures/
www.amazon.com/Teach-Yourself-Structures-Algorithms
Semester – III

17ITU302 OPERATING SYSTEMS 4H – 4C

Instruction Hours / week: L: 4 T: 0 P: 0  Marks: Int : 40 Ext : 60  Total: 100

SCOPE
This course recognize the concepts and principles of operating systems, provide students with the basic knowledge and skills of memory, device and Process management and techniques and provide experience on MS Windows and LINUX environment.

OBJECTIVES

- Explain basic Idea about the operating system.
- Concept and techniques involved in memory, device and Process management.
- Work in MS Windows and LINUX environment.

UNIT-I


UNIT-II


UNIT-III

Memory Management: Physical and Virtual address space-Memory Allocation strategies – Fixed and Variable partitions-Paging-Segmentation-Virtual memory.

UNIT-IV

File and I/O Management: Directory structure-File operations-File Allocation methods-Device management.

UNIT-V


Suggested Readings


Web Sites
1. www.cs.columbia.edu/~nieh/teaching/e6118_s00/
2. www.clarkson.edu/~jnm/cs644
3. pages.cs.wisc.edu/~remzi/Classes/736/Fall2002/
SCOPE
It is an entry-level course for creating database systems. This course is designed for students who have an interest in developing applications. It is used to understand the role and nature of relational database management systems (RDBMS) in today’s IT environment.

OBJECTIVES

- To understand the application development environment.
- To gain programming Skills and Database Creation in RDBMS.
- Ability to use SQL for storing and retrieving data from the RDBMS.
- Ability to arrive at a normalized design of tables and other database objects in RDBMS.
- Ability to use PL/SQL

UNIT - I Understanding Database Fundamentals

Unit - II Entities and Entity Relationships

Unit - III Objects

Unit - IV Overview of PL/SQL
Functions: Local and global – procedures vs functions – stored procedures, functions – create procedure syntax - create function syntax – calling procedures, functions. Replacing and dropping procedures, functions.

Unit - V Packages and Normalization

Suggested Readings

Web Sites
2. www.databasedir.com
17ITU304A  ANDROID PROGRAMMING  3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0  Marks: Int : 40 Ext : 60  Total: 100

SCOPE
This course motivates the students to design, create, deploy, and test applications for the Android mobile phone platform.

OBJECTIVES
- Student can build own Android apps.
- Explain the differences between Android and other mobile development environments.
- Understand how Android applications work, their life cycle, manifest, Intents, and using external resources.
- Design and develop useful Android applications with compelling user interfaces by using, extending, and creating your own layouts and Views and using Menus.
- Take advantage of Android's APIs for data storage, retrieval, user preferences, files, databases, and content providers.

UNIT-I

UNIT-II
Overview of object oriented programming using Java: OOPs Concepts: Inheritance, Polymorphism, Interfaces, Abstract class, Threads, Overloading and Overriding, Java Virtual Machine.

UNIT-III
Development Tools: Installing and using Eclipse with ADT plug-in, Installing Virtual machine for Android sandwich/Jelly bean (Emulator), configuring the installed tools, creating a android project– Hello Word, run on emulator, Deploy it on USB-connected Android device.

UNIT-IV
User Interface Architecture: Application context, intents, Activity life cycle, multiple screen sizes.
User Interface Design: Form widgets, Text Fields, Layouts, Button control, toggle buttons, Spinners(Combo boxes), Images, Menu, Dialog.

UNIT-V
Database: Understanding of SQLite database, connecting with the database.
Suggested Readings

REFERENCES
17ITU304B        PROGRAMMING IN MATLAB          3H – 3C

Instruction Hours / week: L: 3 T: 0 P: 0     Marks: Int : 40 Ext : 60     Total: 100

SCOPE
A student who successfully completes this course should be able to learn how to use MATLAB, learn how to program in MATLAB, ability to create a computer program to solve problems in science and engineering.

OBJECTIVE
- To learn fundamental programming concepts using a block-structured language (MATLAB).
- To learn General problem-solving techniques, including the concept of step-wise refinement applied to the development of algorithms.

UNIT-I
Introduction to Programming: Components of a computer, working with numbers, Machine code, Software hierarchy.

UNIT-II

UNIT-III
Graph Plots: Basic plotting, Built in functions, Generating waveforms, Sound replay, load and save. Procedures and Functions: Arguments and return values, M-files, Formatted console input-output, String handling,

UNIT-IV
Control Statements: Conditional statements: If, Else, Else-if, Repetition statements: While, for loop

UNIT-V
Manipulating Text: Writing to a text file, Reading from a text file, Randomising and sorting a list, searching a list. GUI Interface: Attaching buttons to actions, Getting Input, Setting Output

Suggested Readings

1. Write a program to search an element from a list. Give user the option to perform Linear or Binary search. Use Template functions.

2. WAP using templates to sort a list of elements. Give user the option to perform sorting using Insertion sort, Bubble sort or Selection sort.

3. Implement Linked List using templates. Include functions for insertion, deletion and search of a number, reverse the list and concatenate two linked lists (include a function and also overload operator +).

4. Implement Doubly Linked List using templates. Include functions for insertion, deletion and search of a number, reverse the list.

5. Implement Circular Linked List using templates. Include functions for insertion, deletion and search of a number, reverse the list.

6. Perform Stack operations using Linked List implementation.

7. Perform Stack operations using Array implementation. Use Templates.

8. Perform Queues operations using Circular Array implementation. Use Templates.

9. Create and perform different operations on Double-ended Queues using Linked List implementation.

10. WAP to scan a polynomial using linked list and add two polynomial.

11. WAP to calculate factorial and to compute the factors of a given no. (i) using recursion, (ii) using iteration

12. (ii) WAP to display fibonacci series (i) using recursion, (ii) using iteration

13. WAP to calculate GCD of 2 number (i) with recursion (ii) without recursion

14. WAP to create a Binary Search Tree and include following operations in tree: (a) Insertion (Recursive and Iterative Implementation)
(b) Deletion by copying
(c) Deletion by Merging
(d) Search a no. in BST
(e) Display its preorder, postorder and inorder traversals Recursively
(f) Display its preorder, postorder and inorder traversals Iteratively
(g) Display its level-by-level traversals
(h) Count the non-leaf nodes and leaf nodes
(i) Display height of tree
(j) Create a mirror image of tree
(k) Check whether two BSTs are equal or not

15. WAP to convert the Sparse Matrix into non-zero form and vice-versa.
16. WAP to reverse the order of the elements in the stack using additional stack.
17. WAP to reverse the order of the elements in the stack using additional Queue.
18. WAP to implement Diagonal Matrix using one-dimensional array.
19. WAP to implement Lower Triangular Matrix using one-dimensional array.
20. WAP to implement Upper Triangular Matrix using one-dimensional array.
21. WAP to implement Symmetric Matrix using one-dimensional array.
22. WAP to create a Threaded Binary Tree as per inorder traversal, and implement operations like finding the successor / predecessor of an element, insert an element, inorder traversal.
23. WAP to implement various operations on AVL Tree.
1. Write a program (using fork() and/or exec() commands) where parent and child execute:
   a) same program, same code.
   b) same program, different code.
   c) before terminating, the parent waits for the child to finish its task.
2. Write a program to report behaviour of Linux kernel including kernel version, CPU type and model. (CPU information)
3. Write a program to report behaviour of Linux kernel including information on configured memory, amount of free and used memory. (memory information)
4. Write a program to print file details including owner access permissions, file access time, where file name is given as argument.
5. Write a program to copy files using system calls.
6. Write program to implement FCFS scheduling algorithm.
7. Write program to implement Round Robin scheduling algorithm.
8. Write program to implement SJF scheduling algorithm.
9. Write program to implement non-preemptive priority based scheduling algorithm.
10. Write program to implement preemptive priority based scheduling algorithm.
11. Write program to implement SRJF scheduling algorithm.
12. Write program to calculate sum of n numbers using thread library.
13. Write a program to implement first-fit, best-fit and worst-fit allocation strategies.
Instruction Hours / week: L: 0 T: 0 P: 4   Marks: Int : 40 Ext : 60   Total: 100

1. Create a table with following fields:
   Employee table:

<table>
<thead>
<tr>
<th>Field name</th>
<th>Constraint</th>
<th>Type</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Employee_no</td>
<td>Primary key</td>
<td>Character</td>
<td>6</td>
</tr>
<tr>
<td>Employee_name</td>
<td></td>
<td>Character</td>
<td>30</td>
</tr>
<tr>
<td>Address</td>
<td></td>
<td>Character</td>
<td>25</td>
</tr>
<tr>
<td>Designation</td>
<td></td>
<td>Character</td>
<td>15</td>
</tr>
<tr>
<td>Dob</td>
<td></td>
<td>Date</td>
<td></td>
</tr>
<tr>
<td>Gender</td>
<td>Check</td>
<td>Character</td>
<td>1</td>
</tr>
<tr>
<td>Doj</td>
<td></td>
<td>Date</td>
<td></td>
</tr>
<tr>
<td>Salary</td>
<td></td>
<td>Number</td>
<td>10,2</td>
</tr>
</tbody>
</table>

   Queries:
   a) Display name of the employees whose salary is greater than “10,000”.
   b) Display the details of employees in ascending order according to Employee Code
   c) Display the details of employees earning the highest salary
   d) Display the names of employees who earn more than “Ravi”.

2. Create table named Student with following fields and insert the values:

<table>
<thead>
<tr>
<th>Field name</th>
<th>field type</th>
<th>field size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Student Name</td>
<td>Character</td>
<td>15</td>
</tr>
<tr>
<td>Gender</td>
<td>Character</td>
<td>6</td>
</tr>
<tr>
<td>Roll No.</td>
<td>Character</td>
<td>10</td>
</tr>
<tr>
<td>Department Name</td>
<td>Character</td>
<td>15</td>
</tr>
<tr>
<td>Address</td>
<td>Character</td>
<td>25</td>
</tr>
<tr>
<td>Percentage of marks</td>
<td>Number</td>
<td>4,2</td>
</tr>
</tbody>
</table>

   Queries:
   a) Calculate the average mark percentage of the students
   b) Display the names of the students whose percentage marks are greater than 80%
   c) Display the details of the students who got the highest percentage of marks
   d) Display the details of the students whose mark percentage between 50 and 70
   e) Display the details of the students whose mark percentage is greater than the mark percentage of Roll No=12CA01

3. Create a table with following fields:
   Staff table:

<table>
<thead>
<tr>
<th>Field name</th>
<th>Constraint</th>
<th>Type</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Staff_no</td>
<td>Primary key</td>
<td>Character</td>
<td>6</td>
</tr>
<tr>
<td>Staff_name</td>
<td></td>
<td>Character</td>
<td>30</td>
</tr>
<tr>
<td>Dob</td>
<td></td>
<td>Date</td>
<td></td>
</tr>
<tr>
<td>Dept_code</td>
<td>Foreign key</td>
<td>Character</td>
<td>4</td>
</tr>
<tr>
<td>Designation</td>
<td></td>
<td>Character</td>
<td>15</td>
</tr>
</tbody>
</table>
Basic | Number | 7,2

Department table:

<table>
<thead>
<tr>
<th>Field name</th>
<th>constraint</th>
<th>Type</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dept_code</td>
<td>Primary key</td>
<td>Character</td>
<td>4</td>
</tr>
<tr>
<td>Dept_name</td>
<td></td>
<td>Character</td>
<td>30</td>
</tr>
</tbody>
</table>

Execute the following queries:

1. To list the staff who joined 2 years back.
2. To list the staff in computer science dept.
3. To list the staff_name and the dept_name in which he/she works.
4. To list the maximum and minimum salary in each dept.
5. To list the dept along with the total amount spent on salary
6. To list the name of the employees who draw the salary more than the average salary.

4. Create a table with the following fields:

Book table:

<table>
<thead>
<tr>
<th>Field name</th>
<th>Constraint</th>
<th>Type</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Access_no</td>
<td>Primary key</td>
<td>Character</td>
<td>6</td>
</tr>
<tr>
<td>Title</td>
<td></td>
<td>Character</td>
<td>30</td>
</tr>
<tr>
<td>Author</td>
<td></td>
<td>Character</td>
<td>30</td>
</tr>
<tr>
<td>Publisher</td>
<td></td>
<td>Character</td>
<td>30</td>
</tr>
<tr>
<td>Subject</td>
<td></td>
<td>Character</td>
<td>10</td>
</tr>
<tr>
<td>Price</td>
<td></td>
<td>Number</td>
<td>6,2</td>
</tr>
</tbody>
</table>

Execute the following queries:

1. The title of C and C++ books.
2. The books written by a particular author.
3. The books which costs between Rs.300/- and Rs.500/-
4. The number of books available in each subject.
5. The books in the decreasing order of the cost.

5. Create two tables course and batch with following fields

COURSE: coursecodeno number(5), course name varchar(20), syllabus varchar(20)
BATCH: bcode number(5), coursecode number(5), starting_date date, duration number(3), coursefee number(10,2)

Perform the following queries

- Insert the details for course and batch tables with 10 records
- Show the description of the two tables
- Select all the fields from course & batch tables
- Select all the fields from course & batch tables where coursecode=10
- Select all the fields from batch table where starting date=march 10th
- Select batch code from batch table where net income>50000
• Select coursename, batch code & starting date from batch and course tables where course code of batch table and course code of course table are equal
• Select a syllabus from course where coursecode=5

6. Create a table with the following fields:
Account table:

<table>
<thead>
<tr>
<th>Field name</th>
<th>Constraint</th>
<th>Type</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acc_no</td>
<td>Primary key</td>
<td>Number</td>
<td>4</td>
</tr>
<tr>
<td>Cust_name</td>
<td></td>
<td>Varchar2</td>
<td>30</td>
</tr>
<tr>
<td>Branch_name</td>
<td></td>
<td>Varchar2</td>
<td>30</td>
</tr>
<tr>
<td>Cust_city</td>
<td></td>
<td>Varchar2</td>
<td>30</td>
</tr>
</tbody>
</table>

Borrower table:

<table>
<thead>
<tr>
<th>Field name</th>
<th>Constraint</th>
<th>Type</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acc_no</td>
<td>Foreign key</td>
<td>Number</td>
<td>30</td>
</tr>
<tr>
<td>Branch_name</td>
<td></td>
<td>Varchar2</td>
<td>30</td>
</tr>
<tr>
<td>Amount</td>
<td></td>
<td>Number</td>
<td>8,2</td>
</tr>
</tbody>
</table>

Write queries to perform different types of join.

7. Write the PL/SQL program to find the factorial and fibonacci series of given numbers.

8. (i) Write the PL/SQL program to check whether the string is Palindrome.
   (ii) Write the PL/SQL program to reverse a number
   (iii) Write the PL/SQL program to check whether the number is Armstrong

9. Write a PL/SQL block to create and handle user defined exception
clientmaster

<table>
<thead>
<tr>
<th>Field name</th>
<th>Constraint</th>
<th>Type</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Client_id</td>
<td></td>
<td>Number</td>
<td>6</td>
</tr>
<tr>
<td>Client_name</td>
<td></td>
<td>Varchar2</td>
<td>30</td>
</tr>
<tr>
<td>Address</td>
<td></td>
<td>Varchar2</td>
<td>50</td>
</tr>
<tr>
<td>Phone</td>
<td></td>
<td>Number</td>
<td>10</td>
</tr>
<tr>
<td>Balance</td>
<td></td>
<td>Number</td>
<td>10,2</td>
</tr>
</tbody>
</table>

10. Create table with following fields:
Product table:

<table>
<thead>
<tr>
<th>Field name</th>
<th>Constraint</th>
<th>Type</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Product_code</td>
<td>Primary key</td>
<td>Varchar2</td>
<td>7</td>
</tr>
<tr>
<td>Product_name</td>
<td></td>
<td>Varchar2</td>
<td>30</td>
</tr>
<tr>
<td>Price</td>
<td></td>
<td>Number</td>
<td>6,2</td>
</tr>
<tr>
<td>Quantity</td>
<td></td>
<td>Number</td>
<td>4</td>
</tr>
</tbody>
</table>
Vendor table:

<table>
<thead>
<tr>
<th>Field name</th>
<th>Constraint</th>
<th>Type</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vendor_name</td>
<td></td>
<td>Varchar2</td>
<td>30</td>
</tr>
<tr>
<td>Vendor address</td>
<td></td>
<td>Varchar2</td>
<td>30</td>
</tr>
<tr>
<td>Product_code</td>
<td>Foreign Key</td>
<td>Varchar2</td>
<td>7</td>
</tr>
</tbody>
</table>

Create a trigger to fire when the Record is deleted and inserted.

11. Write a PL/SQL trigger to update the records while deleting the one record in another table.

Voters_master:

<table>
<thead>
<tr>
<th>Field name</th>
<th>Constraint</th>
<th>Type</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Voterid</td>
<td>Primary key</td>
<td>Number</td>
<td>5</td>
</tr>
<tr>
<td>Name</td>
<td></td>
<td>Varchar2</td>
<td>30</td>
</tr>
<tr>
<td>Ward_no</td>
<td>Primary Key</td>
<td>Number</td>
<td>4</td>
</tr>
<tr>
<td>Dob</td>
<td></td>
<td>Date</td>
<td></td>
</tr>
<tr>
<td>Address</td>
<td></td>
<td>Varchar2</td>
<td>150</td>
</tr>
</tbody>
</table>

New_list

<table>
<thead>
<tr>
<th>Field name</th>
<th>Constraint</th>
<th>Type</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Voterid</td>
<td></td>
<td>Number</td>
<td>5</td>
</tr>
<tr>
<td>Ward_no</td>
<td></td>
<td>Number</td>
<td>4</td>
</tr>
<tr>
<td>Name</td>
<td></td>
<td>Varchar2</td>
<td>30</td>
</tr>
<tr>
<td>Description</td>
<td></td>
<td>Character</td>
<td>50</td>
</tr>
</tbody>
</table>

12. Create a table to store the salary details of the employees in a company. Declare the cursor id to contain empno, employee name and net salary. Use cursor to update the employee details.

Salary:

<table>
<thead>
<tr>
<th>Field name</th>
<th>Constraint</th>
<th>Type</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Emp_no</td>
<td>Primary key</td>
<td>Number</td>
<td>4</td>
</tr>
<tr>
<td>Emp_name</td>
<td></td>
<td>Varchar2</td>
<td>30</td>
</tr>
<tr>
<td>Designation</td>
<td></td>
<td>Varchar2</td>
<td>25</td>
</tr>
<tr>
<td>Dept</td>
<td></td>
<td>Varchar2</td>
<td>30</td>
</tr>
<tr>
<td>Basic</td>
<td></td>
<td>Number</td>
<td>5</td>
</tr>
</tbody>
</table>

13. Create a table stock contains the itemcode varchar2(10), itemname varchar2(50), current_stock number(5), date_of_last_purchase date. Write a stored procedure to seek for an item using itemcode and delete it, if the date of last purchase is before 1 year from the current date. If not, update the current stock.
14. Create a table to contain phone_number, user_name, address. Write a function to search for address using phone_number.
1. Create ―Hello World‖ application. That will display ―Hello World‖ in the middle of the screen in the emulator. Also display ―Hello World‖ in the middle of the screen in the Android Phone.

2. Create an application with login module. (Check username and password).

3. Create spinner with strings taken from resource folder (res >> value folder) and on changing the spinner value, Image will change.

4. Create a menu with 5 options and and selected option should appear in text box.

5. Create a list of all courses in your college and on selecting a particular course teacher-in-charge of that course should appear at the bottom of the screen.

6. Create an application with three option buttons, on selecting a button colour of the screen will change.

7. Create and Login application as above. On successful login, pop up the message.

8. Create an application to Create, Insert, update, Delete and retrieve operation on the database.
1. Write a program to assign the following expressions to a variable A and then to print out the value of A.
   a. \( \frac{3+4}{5+6} \)
   b. \( 2\pi^2 \)
   c. \( \sqrt{2} \)
   d. \( (0.0000123 + 5.67 \times 10^{-3}) \times 0.4567 \times 10^{-4} \)

2. Celsius temperatures can be converted to Fahrenheit by multiplying by 9, dividing by 5, and adding 32. Assign a variable called C the value 37, and implement this formula to assign a variable F the Fahrenheit equivalent of 37 Celsius.

3. Set up a vector called N with five elements having the values: 1, 2, 3, 4, 5. Using N, create assignment statements for a vector X which will result in X having these values:
   a. 2, 4, 6, 8, 10
   b. 1/2, 1, 3/2, 2, 5/2
   c. 1, 1/2, 1/3, 1/4, 1/5
   d. 1, 1/4, 1/9, 1/16, 1/25

4. A supermarket conveyor belt holds an array of groceries. The price of each product (in pounds) is \([ 0.6, 1.2, 0.5, 1.3 ]\); while the numbers of each product are \([ 3, 2, 1.5 ]\). Use MATLAB to calculate the total bill.

5. The sortrows(x) function will sort a vector or matrix X into increasing row order. Use this function to sort a list of names into alphabetical order.

6. The —identity‖ matrix is a square matrix that has ones on the diagonal and zeros elsewhere. You can generate one with the eye() function in MATLAB. Use MATLAB to find a matrix B, such that when multiplied by matrix \( A = \begin{bmatrix} 1 & 2 \\ -1 & 0 \end{bmatrix} \) the identity matrix \( \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \) is generated. That is \( A \cdot B = I \).

7. Create an array of N numbers. Now find a single MATLAB statement that picks out from that array the 1, 4, 9, 16, \( \ldots \), \( \sqrt{N} \)th entries, i.e. those numbers which have indices that are square numbers.

8. Draw a graph that joins the points (0,1), (4,3), (2,0) and (5,-2).

9. The seeds on a sunflower are distributed according to the formula below. Plot a small circle at each of the first 1000 co-ordinates:
   \[ r_n = \sqrt{n} \]
   \[ \theta_n = \frac{137.51}{180} \pi n \]
10. Calculate 10 approximate points from the function \( y=2x \) by using the formulae:
   i. \( x_n = n \)
   ii. \( y_n = 2n + \text{rand} - 0.5 \)
Fit a line of best fit to these points using the function `polyfit()` with degree=1, and generate co-ordinates from the line of best fit using `polyval()`. Use the on-line help to find out how to use these functions. Plot the raw data and the line of best fit.

11. Calculate and replay 1 second of a sinewave at 500Hz with a sampling rate of 11025Hz. Save the sound to a file called `ex35.wav`. Plot the first 100 samples.

12. Calculate and replay a 2 second chirp. That is, a sinusoid that steadily increases in frequency with time, from say 250Hz at the start to 1000Hz at the end.

13. Build a square wave by adding together 10 odd harmonics: 1f, 3f, 5f, etc. The amplitude of the nth harmonic should be \( 1/n \). Display a graph of one cycle of the result superimposed on the individual harmonics.

14. Write a function called `FtoC` (`ftoc.m`) to convert Fahrenheit temperatures into Celsius. Make sure the program has a title comment and a help page. Test from the command window with:
   i. `FtoC(96)`
   ii. `lookfor Fahrenheit`
   iii. `help FtoC`

15. Write a program to input 2 strings from the user and to print out (i) the concatenation of the two strings with a space between them, (ii) a line of asterisks the same length as the concatenated strings, and (iii) the reversed concatenation. For example:
   i. Enter string 1: Mark
   ii. Enter string 2: Huckvale
   iii. Mark Huckvale
   iv. **************
   v. elavkcuH kraM
SCOPE
This course is to master the fundamentals of data communications networks by gaining a working knowledge of data transmission concepts, understanding the operation of all seven layers of OSI Model and the protocols used in each layer.

OBJECTIVES
- Various transmission media, their comparative study, fiber optics and wireless media
- Categories and topologies of networks (LAN and WAN) Layered architecture (OSI and TCP/IP) and protocol suites.
- Channel error detection and correction, MAC protocols, Ethernet and WLAN.
- Details of IP operations in the INTERNET and associated routing principles

UNIT I
Introduction to Computer Networks: Network definition; network topologies; network classifications; network protocol; layered network architecture; overview of OSI reference model; overview of TCP/IP protocol suite. Data Communication Fundamentals and Techniques: Analog and digital signal; data-ratelimts; digital to digital line encoding schemes; pulse code modulation; parallel and serial transmission;

UNIT – II
Digital to analog modulation--; multiplexing techniques- FDM, TDM; transmission media. Networks Switching Techniques and Access mechanisms: Circuit switching; packetswitching - connectionless datagram switching, connection-oriented virtual circuit switching; dial-up modems; digital subscriber line; cable TV for data transfer.

UNIT – III
Data Link Layer Functions and Protocol: Error detection and error correction techniques; data-link control- framing and flow control; error recovery protocols- stop and wait ARQ, go-back-n ARQ; Point to Point Protocol on Internet.

UNIT – IV
Multiple Access Protocol and Networks: CSMA/CD protocols; Ethernet LANS; connecting LAN and back-bone networks- repeaters, hubs, switches, bridges, router and gateways; Networks Layer Functions and Protocols: Routing; routing algorithms; network layer protocol of Internet- IP protocol, Internet control protocols.

Unit V
Transport Layer Functions and Protocols: Transport services- error and flow control, Connection establishment and release- three way handshake; Overview of Application layer protocol: Overview of DNS protocol; overview of WWW &HTTP protocol.

Suggested Readings

WEB SITES
1. en.wikipedia.org/wiki/Internet_protocol_suite
3. www.yale.edu/pclt/COMM/TCPIP.HTM
4. www.w3schools.com/tcpip/default.asp
SCOPE
The graduates of the software engineering program shall be able to apply proper theoretical, technical, and practical knowledge of software requirements, analysis, design, implementation, verification and validation, and documentation. This course enables the students to resolve conflicting project objectives considering viable tradeoffs within limitations of cost, time, knowledge, existing systems, and organizations.

OBJECTIVES
- Apply their knowledge of mathematics, sciences, and computer science to the modeling, analysis, and measurement of software artifacts.
- Work effectively as leader/member of a development team to deliver quality software artifacts.
- Analyze, specify and document software requirements for a software system.
- Implement a given software design using sound development practices.
- Verify, validate, assess and assure the quality of software artifacts.
- Design, select and apply the most appropriate software engineering process for a given project, plan for a software project, identify its scope and risks, and estimate its cost and time.
- Express and understand the importance of negotiation, effective work habits, leadership, and good communication with stakeholders, in written and oral forms, in a typical software development environment.

UNIT-I

UNIT-II

UNIT-III
UNIT-IV

UNIT-V

Suggested Readings


WEB SITES
SCOPE
Visual Basic .Net is emphasizing OOP more than ever before, scope has become even more important because much of the power of classes and objects is all about restricting scope and hiding implementation details to make things simpler.

OBJECTIVES
• This course introduces computer programming using the VISUAL BASIC programming language with object-oriented programming principles.
• Emphasis is on event-driven programming methods, including creating and manipulating objects, classes, and using object-oriented tools such as the class debugger.
• To identify the differences between the procedural languages and event driven languages.

UNIT I
Beginning Visual Basic: Introduction to Visual Basic: Introduction Graphics User Interface (GUI), Programming Language (Procedural, Object oriented, event driven), The Visual Basic environment IDE, Introduction to VB Controls: Textboxes, Frames, check boxes, options buttons, setting a border and style, the shape control, the line control, working with multiple controls and their properties, designing the user interface, keyboard access, tab controls. Default & controls property, coding for Controls, list box and combo box and their properties, filling the list box using property window/add item method, picture/image box and their properties.

UNIT II

UNIT III
Writing Code: Control flow statements – If – Then – If-then-else – Nested control statements – Select case – Loop statements – Do-loop – For-Next – While Wend – Exit statement. Displaying message in Message box, testing whether input is valid or not. Collections – procedures – Subroutines – Functions – Calling procedures – Object Browser – Creating classes and Objects – I/O statements
UNIT IV

**Working with forms and procedures:** Introducing to forms and types of forms and setting form properties, creating, adding, removing forms in project, hide, show method, load, unload, statement, Me keywords, Referring to objects on a different forms. Creating an application using controls: What is on the toolbar – Textbox control – Picture box – Image box – Label box – Frame – List box – Option button – Combo box – Command Button – check box – The Drive, Directory, File list controls – Teh Line & Shape control – Scroll Box – Data – Timer.

UNIT V

**Multiple Document Interface & Menus:** Why MDI Forms – Features of an MDI forms – Loading MDI forms & child forms – creating a simple MDI forms – Accessing MDI froms – creating MENUS – POP-UP MENUS. Data access controls: JET database Engine – ADODC – DAO Data control – ODBC Data Source Administrator – DATA REPORT.

Suggested Readings

SCOPE
This course relates to the interface between web servers and their clients. The course provides information includes markup languages, programming interfaces and languages, and standards for document identification and display. The use of Web technology makes to enhance active student learning and improves their creativity in making web pages.

OBJECTIVES
- To create new web page
- To understand the fundamental features of web applications
- To understand the objects and components needed for a web designing.

UNIT – I
Java: Use of Objects, Array and ArrayList class JavaScript: Datatypes, operators, functions, control structures, events and event handling.

UNIT – II
JDBC: JDBC fundamentals, Establishing Connectivity and working with connection interface, working with statements, creating and executing SQL statements, working with ResultSet objects.

UNIT – III

UNIT – IV
JSP: Implicit Objects, conditional processing, displaying values, Using an expression to Set an Attribute, declaring variables and methods, error handling and debugging, sharing data between JSP pages, Requests, and Users, Database Access.

UNIT – V
Java beans: Jaba Beans Fundamentals, JAR Files, Introspection, Developing a simple Bean, Connecting to DB.
Suggested Readings
SCOPE
This course relates to the interface between web servers and their clients. The course provides information includes markup languages, programming interfaces and languages, and standards for document identification and display. The use of Web technology makes to enhance active student learning and improves their creativity in making web pages.

OBJECTIVES

- To Create a new webpage
- To understand the fundamental features of web applications.
- To understand the objects and components needed for a web designing.

UNIT-I
Introduction: Understanding Mark-up Languages, Introduction to XML and its Goals.

UNIT-II

UNIT-III
Other XML Concepts: Scripting XML

UNIT-IV
Other XML Concepts: XML as Data, Linking with XML

UNIT-V
XML with Style: XSL –Style Sheet Basics, XSL basics, XSL style sheets.

Suggested Readings

2. Michael, J. Young. Step by Step XML.
1. Simulate Cyclic Redundancy Check (CRC) error detection algorithm for noisy channel.
2. Simulate and implement stop and wait protocol for noisy channel.
3. Simulate and implement go back n sliding window protocol.
4. Simulate and implement selective repeat sliding window protocol.
5. Simulate and implement distance vector routing algorithm
<table>
<thead>
<tr>
<th>S. No</th>
<th>Practical Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>• Problem Statement</td>
</tr>
<tr>
<td></td>
<td>• Process Model</td>
</tr>
<tr>
<td>2.</td>
<td>Requirement Analysis:</td>
</tr>
<tr>
<td></td>
<td>• Creating a Data Flow</td>
</tr>
<tr>
<td></td>
<td>• Data Dictionary, Use Cases</td>
</tr>
<tr>
<td>3.</td>
<td>Project Management:</td>
</tr>
<tr>
<td></td>
<td>• Computing FP</td>
</tr>
<tr>
<td></td>
<td>• Effort</td>
</tr>
<tr>
<td></td>
<td>• Schedule, Risk Table, Timeline chart</td>
</tr>
<tr>
<td>4.</td>
<td>Design Engineering:</td>
</tr>
<tr>
<td></td>
<td>• Architectural Design</td>
</tr>
<tr>
<td></td>
<td>• Data Design, Component Level Design</td>
</tr>
<tr>
<td>5.</td>
<td>Testing:</td>
</tr>
<tr>
<td></td>
<td>• Basis Path Testing</td>
</tr>
</tbody>
</table>

Sample Projects:

1. **Criminal Record Management**: Implement a criminal record management system for jailers, police officers and CBI officers

2. **DTC Route Information**: Online information about the bus routes and their frequency and fares

3. **Car Pooling**: To maintain a web based intranet application that enables the corporate employees within an organization to avail the facility of carpooling effectively.

4. Patient Appointment and Prescription Management System

5. Organized Retail Shopping Management Software

6. Online Hotel Reservation Service System

7. Examination and Result computation system

8. Automatic Internal Assessment System

9. Parking Allocation System

10. Wholesale Management System
Implement the following projects using Visual Basic Language

1. To calculate the Simple interest and compound interest
2. To perform the following string functions
   a. Upper to lower
   b. String concatenation
   c. String length
   d. String compare
   e. Bold, Italic and Underline
   f. Rtrim and Ltrim
   g. Change the background of form
3. To draw different shapes and fill with different colors using MDI form
4. To implement a simple calculator
5. To create an ActiveX Control
6. To animate the picture using timer control
7. To convert text to voice
8. To simulate web browser
9. Develop a project for Railway Reservation System
10. Develop a project for Employee Payroll system
11. Develop a project for Student Information system. Generate data report for Student marklist
12. Develop a project to generate Barcode for Library applications
Create event driven program for the following:

1. Print a table of numbers from 5 to 15 and their squares and cubes using alert.
2. Print the largest of three numbers
3. Find the factorial of a number n
4. Enter a list of positive numbers terminated by Zero. Find the sum and average of these numbers.
5. A person deposits Rs.1000 in a fixed account yielding 5% interest. Compute the amount in the account at the end of each year for n years.
6. Read n numbers. Count the number of negative numbers, positive numbers and zeroes in the list.
1. Information Structure
In this exercise, student will practice identifying the structure of an information object. For the sample document provided below: Label the information structures you see, including containing structures. 1. Draw a tree representation of the structure.

2. Deconstructing an XML Document
In this exercise, student will practice identifying the explicit structure within an XML document. In a sense, this is the reverse of what you did in Exercise #1. For the sample XML markup below, create a document-like representation (or a simple drawing) for the content contained within the XML tags:

```xml
<book>
  <coverInfo>
    <title>The XML Handbook</title>
    <author>Charles F. Goldfarb</author>
    <author>Paul Prescod</author>
    <edition>Second</edition>
    <description>The definitive XML resource: applications, products, and technologies. Revised and expanded—over 600 new pages.</description>
  </coverInfo>
</book>
```

3. Creating XML Markup
In this exercise, create some XML markup based on the tree representation from Exercise #1 above, and the content from the original sample document.

4. Well-Formedness
This exercise checks your understanding of the constraints for well-formedness. Are the following document instances well-formed? Explain any NO answers.

```xml
<list><title>The first list</title><item>An item</item><item>Another item</item></list>
<para>Bathing a cat is a <emph>relatively</emph> easy task as long as the cat is willing.</para>
<bibl><title>How to Bathe a Cat</title><author>Merlin Bauer</author></bibl>
```
5. Well Formedness
This exercise is a bit more challenging than the previous example. Here is a fragment of an XML document instance. Identify all the places where it fails to match the constraints for well-formedness.

```xml
<PROCEDURE><TITLE>How to Bathe a Cat</TITLE>
<OVERVIEW> This procedure tells you how to bathe a cat. </OVERVIEW>
<WARNING> Cats don't like to take baths. You could get hurt doing this. Be sure to obtain all the required protective gear before you start. </WARNING>
<EQUIPMENT>
ITEM>Hockey Mask</ITEM>
ITEM>Padded Full-body Kevlar Armor</ITEM>
ITEM>Tub full of warm water</ITEM>
ITEM>Towels</ITEM>
ITEM>First Aid kit</ITEM>
ITEM>Cat Shampoo</ITEM>
</EQUIPMENT>
<INSTRUCTIONS>
STEP> Locate the cat, who by now is hiding under the bed. </STEP>
STEP> Place the cat in the tub of water. </STEP>
STEP> Using the First Aid kit, repair the damage to your head and arms. </STEP>
STEP> Place the cat back in the tub and hold it down. </STEP>
STEP> Wash it really fast, then make an effort to dry it with the towels. </STEP>
STEP> Decide not to do this again. </STEP>
</INSTRUCTIONS>
```
17ITU501A  COMPUTER GRAPHICS  Semester – V

Instruction Hours / week: L: 4 T: 0 P: 0      Marks: Int : 40 Ext : 60      Total: 100

SCOPE
This course presents an introduction to computer graphics designed to give the student an overview of fundamental principles. The course will include an overview of common graphics hardware, 2D and 3D transformations and viewing, and basic raster graphics concepts such as scan-conversion, and clipping. Methods for modeling objects as polygonal meshes or smooth surfaces, and as rendering such as hidden-surface removal, shading, illumination, and shadows will be investigated.

OBJECTIVES
- Have a knowledge and understanding of the structure of an interactive computer graphics system, and the separation of system components.
- Have a knowledge and understanding of geometrical transformations and 3D viewing.
- Be able to create interactive graphics applications.
- Have a knowledge and understanding of techniques for representing 3D geometrical objects.
- Have a knowledge and understanding of the fundamental principles of local and global illumination models.

UNIT-I

UNIT-II

UNIT-III
Fundamental Techniques in Graphics : Raster scan line, circle and ellipse drawing, thick primitives, Polygon filling, line and polygon clipping algorithms, 2D and 3D Geometric Transformations, 2D and 3D Viewing Transformations (Projections- Parallel and Perspective), Vanishing points.

UNIT-IV
Geometric Modeling : Representing curves & Surfaces.

UNIT V
Suggested Readings


Websites

1. http://www.fileformat.info/mirror/egff/ch02_01.htm
SCOPE
Software Testing is designed to establish that the software is working satisfactorily as per the requirements.

COURSE OBJECTIVES
- Software Testing is a process designed to prove that the program is error free.
- Software The job of testing is to certify that the software does its job correctly and can be used in production.

UNIT – I Testing Fundamentals

UNIT – II Examining the code
Static White-Box testing- Formal reviews – Coding Standards and Guidelines- Generic Code Review Checklist. Testing the software with X-Ray glasses: Dynamic White-Box testing-Dynamic White-Box testing versus Debugging-Testing the Pieces- Data Coverage- Code Coverage.

Flowgraphs and Path Testing
Path-testing Basics – Predicates, Path Predicates and Achievable Paths-Path sensitizing-Path Instrumentation-Implementation and Application of Path Testing

UNIT – III Transaction-Flow Testing and Data-Flow Testing

UNIT – IV Domain Testing
Domains and Paths-Domain Testing-Domains and Interface Testing-Domains and Testability

UNIT – V Logic-Based Testing and State Graphs
Motivational Overview-Decision Tables-Path Expressions Again-KV Charts-Specifications State Graphs-Good State Graphs and Bad-State Testing

Suggested Readings

Web Sites
1. www.testinggeek.com
2. www.softwaretestinghelp.com
3. www.softwaretestinginstitute.com
17ITU502A  . NET PROGRAMMING  4H – 4C

Instruction Hours / week: L: 4 T: 0 P: 0  Marks: Int : 40 Ext : 60  Total: 100

SCOPE
Understanding the platform. Determinism and concurrency; Handling input and output securely; safe error handling and logging; Engineering for security features; software security in operations.

OBJECTIVES
- Grasp the fundamentals of a programming language and know the basic differences between programming languages
- .NET using WPF is a latest technology followed in the IT field.
- Choose the architecture based on the problem to be solved.
- Differentiate between the types of applications supported by .Net
- Build, compile and execute a VB .Net program
- Apply techniques to develop error-free software

UNIT I
Introduction to .NET: .NET framework features & architecture, CLR, common Type system, MSIL, Assemblies and class libraries. Introduction to visual studio, Project basics, types of project in .Net, IDE of VB .Net – Menu bar, Tool bar, Solution Explorer, Toolbox, Properties Window, Form Designer, Output Window, Object browser. The environment: Editor tab, format tab, general tab, docking tab. Visual development & event driven programming – Methods and events.

UNIT II

UNIT III

UNIT IV
UNIT V

Suggested Readings


Websites

1. www.startvbdotnet.com
2. www.functionx.com
3. www.dotnetspider.com
4. www.developerfusion.com
5. http://www.wdftutorial.net/HelloWPF.html
SCOPE
This course is to master the fundamentals of communications networks by gaining a working knowledge of data transmission concepts, understanding the operation of all seven layers of OSI Model and the protocols used in each layer.

OBJECTIVES
- Various transmission media, their comparative study, fiber optics and wireless media
- Categories and topologies of networks (LAN and WAN) Layered architecture (OSI and TCP/IP) and protocol suites.
- TCP, UDP, SCTP protocols Ethernet and LAN administration.
- Details of IP operations in the INTERNET and associated routing principles

UNIT-I

UNIT-II
Socket Programming: Socket Introduction; TCP Sockets; TCP Client/Server Example; signal handling

UNIT-III
I/O multiplexing using sockets; Socket Options; UDP Sockets; UDP client server example; Address lookup using sockets.

UNIT-IV
Network Applications: Remote logging; Email; WWW and HTTP.

UNIT-V
LAN administration: Linux and TCP/IP networking: Network Management and Debugging.

Suggested Readings
17ITU503A  MACHINE LEARNING  4H – 4C

Instruction Hours / week: L: 4 T: 0 P: 0  Marks: Int : 40 Ext : 60  Total: 100

SCOPE
The course aims to provide an introduction to the basic principles, techniques, and applications of Machine Learning.

OBJECTIVES
- Students will have a broad understanding of machine learning algorithms and their use in data-driven knowledge discovery and program synthesis.
- Students will have designed and implemented several machine learning algorithms in Java.
- Students will also be able to identify, formulate and solve machine learning problems that arise in practical applications.
- Students will have a knowledge of the strengths and weaknesses of different machine learning algorithms (relative to the characteristics of the application domain) and be able to adapt or combine some of the key elements of existing machine learning algorithms to design new algorithms as needed.

UNIT-I
Introduction: Concept of Machine Learning, Applications of Machine Learning, Key elements of Machine Learning, Supervised vs. Unsupervised Learning, Statistical Learning: Bayesian Method, The Naive Bayes Classifier

UNIT-II
Softwares for Machine Learning and Linear Algebra Overview : Plotting of Data, Vectorization, Matrices and Vectors: Addition, Multiplication, Transpose and Inverse using available tool such as MATLAB.

UNIT-III
Linear Regression: Prediction using Linear Regression, Gradient Descent, Linear Regression with one variable, Linear Regression with multiple variables, Polynomial Regression, Feature Scaling/Selection.
Logistic Regression: Classification using Logistic Regression, Logistic Regression vs. Linear Regression, Logistic Regression with one variable and with multiple variables.

UNIT-IV
Regularization: Regularization and its utility: The problem of Overfitting, Application of Regularization in Linear and Logistic Regression, Regularization and Bias/Variance.

UNIT-V

Suggested Readings
SCOPE
This course introduce students to the basic concepts and techniques of Data Mining, develop skills of using recent data mining software for solving practical problems, gain experience of doing independent study and research.

OBJECTIVES
- To introduce students to the basic concepts and techniques of Data Mining.
- To develop skills of using recent data mining software for solving practical problems.
- To gain experience of doing independent study and research.
- Possess some knowledge of the concepts and terminology associated with database systems, statistics, and machine learning

UNIT-I
Overview: Predictive and descriptive data mining techniques

UNIT-II
Supervised and unsupervised learning techniques

UNIT-III
Process of knowledge discovery in databases, pre-processing methods

UNIT-IV
Data Mining Techniques: Association Rule Mining, classification and regression techniques, clustering

UNIT-V
Scalability and data management issues in data mining algorithms, measures of interestingness.

Suggested Readings
WEB SITES
1. Thedacs.Com
2. Dwreview.Com
3. Pcai.Com
4. Eruditionhome.Com
SCOPE
The objectives of this course are to make the students learn the fundamental theories and techniques of digital image processing, cover the fundamental concepts of visual perception and image acquisition, basic techniques of image manipulation, segmentation and coding, and a preliminary understanding of Computer Vision.

OBJECTIVES

- To perform image manipulations and analysis in many different fields.
- To provide students with the ability to apply knowledge of computing, mathematics, science and engineering to solve problems in multidisciplinary research.

UNIT-I

UNIT-II
Hotelling Transform, Fourier Transforms and properties, FFT (Decimation in Frequency and Decimation in Time Techniques), Convolution, Correlation, 2-D sampling, Discrete Cosine Transform, Frequency domain filtering.

UNIT-III:

UNIT – IV
FAX compression (ITUITT Group-3 and Group-4), Symbol-based coding, JBIG-2, Bit-plane encoding, Bit-allocation, Zonal Coding, Threshold Coding, JPEG, Lossless predictive coding, Lossy predictive coding, Motion Compensation
UNIT-V
Morphological Image Processing: Basics, SE, Erosion, Dilation, Opening, Closing, Hit-or-Miss Transform, Boundary Detection, Hole filling, Connected components, convex hull, thinning, thickening, skeletons, pruning, Geodesic Dilation, Erosion, Reconstruction by dilation and erosion. Image Segmentation: Boundary detection based techniques, Point, line detection, Edge detection, Edge linking, local processing, regional processing, Hough transform, Thresholding, Iterative thresholding, Otsu's method, Moving averages, Multivariable thresholding, Region-based segmentation, Watershed algorithm, Use of motion in segmentation

Suggested Readings

SCOPE
Acquire basic knowledge on Multimedia devices. Understand current trends in multimedia by experiencing a variety of applications and development packages.

OBJECTIVES
- This course in curriculum is an introduction to the multimedia and its applications.
- This course enables students to understand how the web pages are designed interactively.
- How to critically evaluate website quality, learn how to create and maintain quality web pages learn to create and manipulate images.
- To gain the skills and project-based experience needed for entry into web design and development careers.

UNIT I

UNIT II

UNIT III

UNIT IV

UNIT V

Suggested Readings


Websites

1. en.wikipedia.org/wiki/Multimedia
2. www.arena-multimedia.com/
3. www.nextwavemultimedia.com/
1. Write a program to implement Bresenham’s line drawing algorithm.

2. Write a program to implement mid-point circle drawing algorithm.

3. Write a program to clip a line using Cohen and Sutherland line clipping algorithm.

4. Write a program to clip a polygon using Sutherland Hodgeman algorithm.

5. Write a program to apply various 2D transformations on a 2D object (use homogenous coordinates).

6. Write a program to apply various 3D transformations on a 3D object and then apply parallel and perspective projection on it.

7. Write a program to draw Hermite/Bezier curve.
1. Write a program that take three inputs (a, b and c) that represents the sides of a triangle and the output is one of the below four.
   a) Not a triangle
   b) Scalene triangle
   c) Isoceles triangle
   d) Equilateral triangle
   a) Generate test cases using boundary value analysis, equivalence class partitioning and decision table testing
   b) Generate test cases using basis path testing
   c) Run code coverage tool
2. Write a program that determines the nature of roots of a quadratic equation. Output should be one of the following
   • Not a quadratic equation
   • Complex roots
   • Real roots
   • Single roots
   a) Generate test cases using boundary value analysis, equivalence class partitioning and decision table testing
   b) Generate test cases using basis path testing
   c) Run code coverage tool
3. Write a program that checks whether the number is even or odd. Run code coverage tools and find the amount of code being covered.
4. Write a program that dynamically allocates memory to 10 integers using malloc() or calloc() and do not free memory leading to memory leaks. Verify the same using Valgrind. Now, free memory using free() at the end of the program to avoid memory leaks. Verify the same using Valgrind().
5. Using Selenium IDE, write a test suite containing minimum 4 test cases.
6. Conduct a test suite from any two websites.
7. Write and test a program to login a specific webpage
1. Write a VB .Net program to calculate simple interest and compound interest.
2. Write a VB .Net program to implement Calculator.
3. Write a VB.Net program to implement Notepad
4. Write a VB.Net program to draw several shapes and fill with color.
5. Write a VB .Net program to perform the following in list box
   a) Add an item
   b) Delete an item
   c) List count
   d) Clear the list
6. Write a program to calculate the total marks of the student and print the grades.
7. Write a VB .Net program to implement employee payroll system
8. Write a VB .Net program to create and manipulate a file.
9. Write a program to implement a web browser
10. Write a program to maintain the details of doctors in a hospital with their specializations
11. Write a program to animate the picture using timer control.
12. Write a program to move the object from one location to another. Change the color and size of object at different time interval.
13. Write a program to place 10 pictures in the listbox. Using timer control the take the picture from listbox and change the form background after specific time interval.
14. Write a program to implement speaking program. Get the text input from the user and convert into voice.
15. Write a program to implement chatting
Instruction Hours / week: L: 0 T: 0 P: 4 Marks: Int : 40 Ext : 60 Total: 100

1. Simulate Cyclic Redundancy Check (CRC) error detection algorithm for noisy channel.
2. Simulate and implement stop and wait protocol for noisy channel.
3. Simulate and implement go back n sliding window protocol.
4. Simulate and implement selective repeat sliding window protocol.
5. Simulate and implement distance vector routing algorithm.
1. Perform elementary mathematical operations in Octave/MATLAB like addition, multiplication, division and exponentiation.

2. Perform elementary logical operations in Octave/MATLAB (like OR, AND, Checking for Equality, NOT, XOR).

3. Create, initialize and display simple variables and simple strings and use simple formatting for variable.

4. Create/Define single dimension / multi-dimension arrays, and arrays with specific values like array of all ones, all zeros, array with random values within a range, or a diagonal matrix.

5. Use command to compute the size of a matrix, size/length of a particular row/column, load data from a text file, store matrix data to a text file, finding out variables and their features in the current scope.

6. Perform basic operations on matrices (like addition, subtraction, multiplication) and display specific rows or columns of the matrix.

7. Perform other matrix operations like converting matrix data to absolute values, taking the negative of matrix values, adding/removing rows/columns from a matrix, finding the maximum or minimum values in a matrix or in a row/column, and finding the sum of some/all elements in a matrix.

8. Create various type of plots/charts like histograms, plot based on sine/cosine function based on data from a matrix. Further label different axes in a plot and data in a plot.

9. Generate different subplots from a given plot and color plot data.

10. Use conditional statements and different type of loops based on simple example/s.

11. Perform vectorized implementation of simple matrix operation like finding the transpose of a matrix, adding, subtracting or multiplying two matrices.

12. Implement Linear Regression problem. For example, based on a dataset comprising of existing set of prices and area/size of the houses, predict the estimated price of a given house.

13. Based on multiple features/variables perform Linear Regression. For example, based on a number of additional features like number of bedrooms, servant room, number of balconies, number of houses of years a house has been built – predict the price of a house.
14. Implement a classification/ logistic regression problem. For example based on different features of students data, classify, whether a student is suitable for a particular activity. Based on the available dataset, a student can also implement another classification problem like checking whether an email is spam or not.

15. Use some function for regularization of dataset based on problem 14.

16. Use some function for neural networks, like Stochastic Gradient Descent or back propagation - algorithm to predict the value of a variable based on the dataset of problem
Instruction Hours / week: L: 0 T: 0 P: 4 Marks: Int : 40 Ext : 60 Total: 100

1. Use the following learning schemes, with the default settings to analyze the weather data (in weather.arff). for test options, first choose “Use training set”, then choose “Percentage split” using default 66% percentage split. Report model percent error rate.

2. Use iris dataset preprocess and classify it with j4.8 and Naive Bayes classifier. Examine the tree in the classifier output panel.


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

5. Using weka experimenter perfrom comparison analysis of j4.8, oneR and ID3 for vote dataset.

6. Using weka experimenter perfrom comparison analysis of Naive Bayes with different datasets.

7. Apply ZeroR, OneR and j4.8, to classify the iris data in an experiment using 10 train and test runs, with 66% of the data used for 34% used for testing.

8. Using Weka Knowledge flow set up a flow to load an ARFF file (batch mode) and perform a cross-validation using j4.8 (WEKS’s C4.5 implementation).

9. Draw multiple ROC curves in the same plot window, using j4.8 and RandomForest as classifiers.

10. Use any three clustering algorithm on Vehicle data set and find best among them.
1. Write program to read and display digital image using MATLAB or SCILAB
   a. Become familiar with SCILAB/MATLAB Basic commands
   b. Read and display image in SCILAB/MATLAB
   c. Resize given image
   d. Convert given color image into gray-scale image
   e. Convert given color/gray-scale image into black & white image
   f. Draw image profile
   g. Separate color image in three R G & B planes
   h. Create color image using R, G and B three separate planes
   i. Flow control and LOOP in SCILAB
   j. Write given 2-D data in image file

2. To write and execute image processing programs using point processing method
   a. Obtain Negative image
   b. Obtain Flip image
   c. Thresholding
   d. Contrast stretching

3. To write and execute programs for image arithmetic operations
   a. Addition of two images
   b. Subtract one image from other image
   c. Calculate mean value of image
   d. Different Brightness by changing mean value

4. To write and execute programs for image logical operations
   a. AND operation between two images
   b. OR operation between two images
   c. Calculate intersection of two images
   d. Water Marking using EX-OR operation
   e. NOT operation (Negative image)

5. To write a program for histogram calculation and equalization using
   a. Standard MATLAB function
   b. Program without using standard MATLAB functions
   c. C Program

6. To write and execute program for geometric transformation of image
   a. Translation
   b. Scaling
   c. Rotation
   d. Shrinking
7. To understand various image noise models and to write programs for
   a. image restoration
   b. Remove Salt and Pepper Noise
   c. Minimize Gaussian noise
   d. Median filter and Weiner filter

8. Write and execute programs to remove noise using spatial filters
   a. Understand 1-D and 2-D convolution process
   b. Use 3x3 Mask for low pass filter and high pass filter

9. Write and execute programs for image frequency domain filtering
   a. Apply FFT on given image
   b. Perform low pass and high pass filtering in frequency domain
   c. Apply IFFT to reconstruct image

10. Write a program in C and MATLAB/SCILAB for edge detection using
    different edge detection mask

11. Write and execute program for image morphological operations erosion and dilation.

12. To write and execute program for wavelet transform on given image and
    perform inverse wavelet transform to reconstruct image.
1. Change a Shape to Another Shape. (Shape Animation)
2. Create a Man to walk with the help of Key Frame Animation.
3. Change the Colors of an object with the help of Animation.
4. Draw a Bird with Flash tools and make it fly with key Frame Animation.
5. Create a Shining Stones with the help of Movie Clip.
6. Create an animation to represent a growing moon using frame by frame animation.
7. Create an animation to bounce a ball on steps.
8. Simulate movement of a cloud.
9. Create Morphing between two images.
10. Create an Action script to execute for an event in an application.
12. Animate Plane Flying with the Clouds.
14. Create a Web Page using Photoshop
15. Given a picture of a flower with a background. Extract the flower and organize on a different background.
16. Display the given picture through your name using mask.
SCOPE
This course design focuses on the structure of the website including the information architecture, the layout or the pages and the conceptual design with branding. PHP helps the students for developing dynamic web pages.

OBJECTIVES
- To work with open source applications that deal with database and website development.
- Establish a working environment for PHP web page development
- Very familiar with GUI, coded, modified controls

UNIT-I
Introduction to PHP:
PHP introduction, inventions and versions, important tools and software requirements (like Web Server, Database, Editors etc.) -PHP with other technologies, scope of PHP
Basic Syntax, PHP variables and constants-Types of data in PHP , Expressions, scopes of a variable (local, global)-PHP Operators: Arithmetic, Assignment, Relational, Logical operators, Bitwis , ternary and MOD operator.-PHP operator Precedence and associativity

UNIT-II
Handling HTML form with PHP:
Capturing Form Data -GETand POST form methods- Dealing with multi value fields - Redirecting a form after submission -PHP conditional events and Loops: PHP IF Else conditional statements ( Nested IF and Else) -Switch case, while ,For and Do While Loop - Goto , Break ,Continue and exit

UNIT-III
PHP Functions:
Function, Need of Function , declaration and calling of a function-PHP Function with arguments, Default Arguments in Function-Function argument with call by value, call by reference-Scope of Function Global and Local

UNIT-IV
String Manipulation and Regular Expression: (3L) Creating and accessing String , Searching & Replacing String-Formatting, joining and splitting String , String Related Library functions -Use and advantage of regular expression over inbuilt function-Use of preg_match(), preg_replace(), preg_split() functions in regular expression.

UNIT-V
Array:

Anatomy of an Array, Creating index based and Associative array, Accessing array
- Looping with Index based array, with associative array using each() and foreach()
- Some useful Library function

Suggested Readings

5. David Sklar., & Adam Trachtenberg. PHP Cookbook: Solutions & Examples for PHP.
Semester – VI

17ITU601B UNIX / LINUX PROGRAMMING 4H – 4C

Instruction Hours / week: L: 3 T: 0 P: 0  Marks: Int : 40 Ext : 60  Total: 100

SCOPE
This course teaches the student the concepts and principles that underlie modern operating systems, and a practice component to relate theoretical principles with operating system implementation. Learn about processes and processor management, concurrency and synchronization, memory management schemes, file system and secondary storage management, security and protection, etc.

OBJECTIVES
- Understand fundamental operating system abstractions such as processes, threads, files, semaphores, IPC abstractions, shared memory regions, etc.
- Understand how the operating system abstractions can be used in the development of application programs, or to build higher level abstractions
- Understand how the operating system abstractions can be implemented
- Understand the principles of concurrency and synchronization, and apply them to write correct concurrent programs/software
- Understand basic resource management techniques (scheduling or time management, space management) and principles and how they can be implemented. These also include issues of performance and fairness objectives, avoiding deadlocks, as well as security and protection.

UNIT-I
Introduction What is Linux/Unix Operating systems, Difference between linux/unix and other operating systems , Features and Architecture, Various Distributions available in the market, Installation, Booting and shutdown process.

UNIT-II
System processes (an overview), External and internal commands, Creation of partitions in OS, Processes and its creation phases – Fork, Exec, wait

UNIT-III
User Management and the File System Types of Users, Creating users, Granting rights User management commands, File quota and various file systems available, File System Management and Layout, File permissions, Login process, Managing Disk Quotas, Links (hard links, symbolic links)

UNIT-IV
Shell introduction and Shell Scripting What is shell and various type of shell, Various editors present in Linux Different modes of operation in vi editor, What is shell script, Writing and executing the shell script , Shell variable (user defined and system variables)
UNIT-V

**System calls, Using system calls**  Pipes and Filters, Decision making in Shell Scripts (If else, switch), Loops in shell. Functions, Utility programs (cut, paste, join, tr, uniq utilities), Pattern matching utility (grep)

**Suggested Readings**

Semester – VI

17ITU602A  E-COMMERCE TECHNOLOGIES  4H – 4C

Instruction Hours / week: L: 4 T: 0 P: 0  Marks: Int : 40 Ext : 60  Total: 100

SCOPE
- Students would be able to analyze the concept of electronic market and market place.
- Students would be able to understand the business models and standards

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

UNIT I - An Introduction to Electronic commerce

UNIT II - The Internet and WWW
Evolution of Internet, Domain Names and Internet – Organization (.edu, .com, .mil, .gov, .net etc), Types of Network, Internet Service Provider, World Wide Web, Internet & Extranet, Role of Internet in B2B Application, building own website, cost, time, reach, registering a domain name, web promotion, Target email, Banner, Exchange, Shopping Bots.

UNIT III: Electronic data
Electronic data exchange introduction, concepts of EDI and Limitation, Application of eDI, Disadvantages of eDI, EDI model, Electronic Payment System: Introduction, Types of Electronic Payment system, Payment types, Value exchange system, credit card system electronic fund transfer, Paperless bill, modern payment cash, Electronic cash.

UNIT IV: Planning for Electronic Commerce
Planning Electronic commerce initiatives, linking objectives to business strategies, measuring cost objectives, comparing benefits to Costs, strategies for developing electronic commerce web sites.

UNIT V : Internet marketing
The PROS and CONS of online shopping, the CONS of online shopping, Justify an internet business, Internet marketing techniques, The E-Cycle of Internet marketing, personalization e-commerce.
**Suggested Readings**


**Websites**

SCOPE
The main objective of the course is to portray the recent trends in the field of cloud computing and providing exposures to some open source and commercial clouds.

OBJECTIVES
• Provide a good understanding of the concepts, standards and protocols in Cloud computing

UNIT-I

UNIT-II
Cloud Computing Architecture: Comparison with traditional computing architecture (client/server), Services provided at various levels, Service Models- Infrastructure as a Service(IaaS), Platform as a Service(PaaS), Software as a Service(SaaS), How Cloud Computing Works, Deployment Models- Public cloud, Private cloud, Hybrid cloud, Community cloud, Case study of NIST architecture.

UNIT-III
Case Studies: Case study of Service model using Google App Engine, Microsoft Azure, Amazon EC2 , Eucalyptus.

UNIT-IV
Service Management in Cloud Computing: Service Level Agreements (SLAs), Billing & Accounting, Comparing Scaling Hardware: Traditional vs. Cloud, Economics of scaling.

UNIT-V
Cloud Security: Infrastructure Security- Network level security, Host level security, Application level security, Data security and Storage- Data privacy and security Issues, Jurisdictional issues raised by Data location, Authentication in cloud computing.

Suggested Readings

WEB SITES
1. en.wikipedia.org/wiki/Cloud_computing
**Instruction Hours / week:** L: 3 T: 0 P: 0  **Marks:** Int : 40 Ext : 60  **Total:** 100

**SCOPE**
It exposes the students to study numerical techniques as powerful tools in scientific computing

**OBJECTIVES**
This course a deep knowledge to the learners to understand the basic concepts of numerical methods which utilize computers to solve Engineering problems that are not easily solved or even impossible to solve by analytical means.

**UNIT I**

**UNIT II**
Gauss elimination method (with row pivoting) and Gauss-Jordan method – Gauss thomas method for tridiagonal systems. Iterative methods: Jacobi and Gauss-seidal interative methods.

**UNIT III**
Interpolation: Lagrange’s form and Newton’s form – Finite difference operators – Gregory Newton forward and backward differences inerpolation Piecewise polynomial interpolation: Linear interpolation – Cubic spline interpolation (only method).

**UNIT IV**

**UNIT V**

**Suggested Readings**
Instruction Hours / week: L: 4 T: 0 P: 0      Marks: Int : 40 Ext : 60      Total: 100

SCOPE
This course enables for good understanding of the role of system programming and the scope of duties and tasks of a system programmer. This course enables to learn the concepts and principles of developing system-level software (e.g., compiler, and networking software).

OBJECTIVES
- To introduce students the concepts and principles of system programming
- To provide students the knowledge about both theoretical and practical aspects of system programming, teaching them the methods and techniques for designing and implementing system-level programs.
- To train students in developing skills for writing system software with the aid of sophisticated OS services, programming languages and utility tools.

UNIT-I
Assemblers & Loaders, Linkers: One pass and two pass assembler design of an assembler, Absolute loader, relocation and linking concepts, relocating loader and Dynamic Linking., overview of compilation, Phases of a compiler.

UNIT-II
Lexical Analysis:
Role of a Lexical analyzer, Specification and recognition of tokens, Symbol table, lexical

UNIT-III
Parsing:
Bottom up parsing- LR parser, yalTU. Intermediate representations: Three address code generation, syntax directed translation, translation of types, control Statements.

UNIT-IV
Storage organization: Activation records stack allocation.

UNIT-V
Code Generation: Object code generation
Suggested Readings
1. Create a PHP page using functions for comparing three integers and print the largest number.

2. Write a function to calculate the factorial of a number (non-negative integer). The function accept the number as an argument.

3. WAP to check whether the given number is prime or not.

4. Create a PHP page which accepts string from user. After submission that page displays the reverse of provided string.

5. Write a PHP function that checks if a string is all lower case.

6. Write a PHP script that checks whether a passed string is palindrome or not? (A palindrome is word, phrase, or sequence that reads the same backward as forward, e.g., madam or nurses run)

7. WAP to sort an array.

8. Write a PHP script that removes the whitespaces from a string.
Sample string : 'The quick brown fox' Expected Output : Thequickbrownfox

9. Write a PHP script that finds out the sum of first n odd numbers.

10. Create a login page having user name and password. On clicking submit, a welcome message should be displayed if the user is already registered (i.e.name is present in the database) otherwise error message should be displayed.

11. Write a PHP script that checks if a string contains another string.

12. Create a simple 'birthday countdown' script, the script will count the number of days between current day and birth day.

13. Create a script to construct the following pattern, using nested for loop.

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

14. Write a simple PHP program to check that emails are valid.
15. WAP to print first n even numbers.

16. $color = array('white', 'green', 'red')

Write a PHP script which will display the colors in the following way : Output :
white, green, red,
• green • red
• white

17. Using switch case and dropdown list display a —Hello— message depending on the
language selected in drop down list.

18. Write a PHP program to print Fibonacci series using recursion.

19. Write a PHP script to replace the first 'the' of the following string with 'That'.

Sample : 'the quick brown fox jumps over the lazy dog.'
Expected Result : That quick brown fox jumps over the lazy dog.
1. Write a shell script to check if the number entered at the command line is prime or not.
2. Write a shell script to modify —call command to display calendars of the specified months.
3. Write a shell script to modify —call command to display calendars of the specified range of months.
4. Write a shell script to aITUept a login name. If not a valid login name display message — Entered login name is invalid.
5. Write a shell script to display date in the mm/dd/yy format.
6. Write a shell script to display on the screen sorted output of —who command along with the total number of users.
7. Write a shell script to display the multiplication table any number,
8. Write a shell script to compare two files and if found equal asks the user to delete the duplicate file.
9. Write a shell script to find the sum of digits of a given number.
10. Write a shell script to merge the contents of three files, sort the contents and then display them page by page.
11. Write a shell script to find the LCD (least common divisor) of two numbers.
12. Write a shell script to perform the tasks of basic calculator.
13. Write a shell script to find the power of a given number.
14. Write a shell script to find the binomial coefficient C(n , x).
15. Write a shell script to find the permutation P(n,x).
16. Write a shell script to find the greatest number among the three numbers.
17. Write a shell script to find the factorial of a given number.
18. Write a shell script to check whether the number is Armstrong or not.
19. Write a shell script to check whether the file have all the permissions or not.
17ITU612A          E-COMMERCE TECHNOLOGIES- PRACTICAL          Semester – VI

Instruction Hours / week: L: 0 T: 0 P: 4   Marks: Int : 40 Ext : 60     Total: 100

1. Write a HTML program to implement the use of Image map.
2. Write a CSS to implement selectors in HTML
3. Write a CSS to implement pesudo – classes with in-line styles
4. Write a Javascript program to validate a web form
5. Write a Javascript program to allow visitors to see history of visiting your page
6. Write a Javascript program to change random color each 5 seconds
7. Write a pearl program to read a list of n strings (from STDIN) into an array and print a random string from the list (Use srand;rand(@array))
8. Write a pearl program to read a list of n numeric’s from STDIN and find the max, min, range, median and mode. Input size of the list n interactively.
9. Write a pearl program to read a file of words and replaces all words in the file with their uppercase equivalent (hint: use tr/a-z/A-Z/)
10. Write VBScript program to print Fibonacci series using Do..while loop and For loop.
11. Write VBScript program to generate date and time in different format
12. Write VBScript program to print student marklist
13. Develop an ASP code to retrieve information from forms
14. Develop an ASP code to reading and writing cookies information
15. Develop an ASP code using response object methods
1. Create virtual machines that access different programs on same platform.

2. Create virtual machines that access different programs on different platforms.

3. Working on tools used in cloud computing online-
   a) Storage
   b) Sharing of data
   c) Manage your calendar, to-do lists,
   d) A document editing tool

4. Exploring Google cloud

5. Exploring Microsoft cloud

6. Exploring Amazon cloud
1. Find the roots of the equation by bisection method
2. Find the roots of the equation by secant/Regula-Falsi method.
3. Find the roots of the equation by Newton’s method
4. Find the solution of a system of nonlinear equation using Newton’s method
5. Find the solution of tridiagonal system using Gauss Thomas method
6. Find the solution of system of equations using Jacobi/Gauss-Seidal method
7. Find the cubic spline interpolating function
8. Evaluate the approximate value of finite integrals using Gaussian/Romberg integration
9. Solve the boundary value problem using Finite difference method

Note: Programming is to be done in any one of computer algebra systems: MATLAB/MATHEMATICA/MAPLE
1. To implement an assembler for a hypothetical language.
2. To get familiar with lex: write a program to recognize numbers, identifiers.
3. To get familiar with yacc: write a desk calculator.