Bachelor of Computer Applications (BCA)

New Syllabus

w. e. f. 2020-21

Prepared
By
Krishna University

Chairman of the Committee
Prof. Y. K. SundaraKrishna
## Syllabus of Bachelor of Computer Applications (BCA)

w. e. f 2020-21.

### COURSE STRUCTURE

<table>
<thead>
<tr>
<th>S.No</th>
<th>Semester</th>
<th>Hours/ Week</th>
<th>No of Credits</th>
<th>Max. Marks Internal assessment</th>
<th>Max. Marks University Exam</th>
<th>Total Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>I</td>
<td>30</td>
<td>25</td>
<td>125</td>
<td>625</td>
<td>750</td>
</tr>
<tr>
<td>2</td>
<td>II</td>
<td>32</td>
<td>27</td>
<td>125</td>
<td>675</td>
<td>800</td>
</tr>
<tr>
<td>3</td>
<td>III</td>
<td>32</td>
<td>27</td>
<td>125</td>
<td>675</td>
<td>800</td>
</tr>
<tr>
<td>4</td>
<td>IV</td>
<td>36</td>
<td>30</td>
<td>150</td>
<td>750</td>
<td>900</td>
</tr>
<tr>
<td>5</td>
<td>V</td>
<td>36</td>
<td>30</td>
<td>150</td>
<td>750</td>
<td>900</td>
</tr>
<tr>
<td>#6</td>
<td>VI</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>#Total</td>
<td></td>
<td>166</td>
<td>139</td>
<td>675</td>
<td>3475</td>
<td>4150</td>
</tr>
</tbody>
</table>

*# To be finalized*
<table>
<thead>
<tr>
<th>S.No</th>
<th>Paper Code</th>
<th>Subject</th>
<th>Hours/Week</th>
<th>No of Credits</th>
<th>Max. Marks Internal assessment</th>
<th>Max. Marks University Exam</th>
<th>Total Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td></td>
<td>English – I</td>
<td>4</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>2.</td>
<td></td>
<td>Language(H/T/S) – I</td>
<td>4</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>3.</td>
<td></td>
<td>Life Skill Course – I</td>
<td>2</td>
<td>2</td>
<td>-0-</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>4.</td>
<td></td>
<td>Skill Development Course – I</td>
<td>2</td>
<td>2</td>
<td>-0-</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>5.</td>
<td>C1</td>
<td>Computer Fundamentals &amp; Office tools</td>
<td>4</td>
<td>4</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>C1-P</td>
<td>Computer Fundamentals &amp; Office tools-Lab</td>
<td>2</td>
<td>1</td>
<td>-0-</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>6.</td>
<td>C2</td>
<td>Programming in C</td>
<td>4</td>
<td>4</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>C2-P</td>
<td>Programming in C Lab</td>
<td>2</td>
<td>1</td>
<td>-0-</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>7.</td>
<td>C3</td>
<td>Numerical and Statistical Methods</td>
<td>4</td>
<td>4</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>C3-P</td>
<td>Numerical and Statistical Methods-Lab</td>
<td>2</td>
<td>1</td>
<td>-0-</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Total</td>
<td>30</td>
<td>25</td>
<td>125</td>
<td>625</td>
<td>750</td>
</tr>
</tbody>
</table>
## II Semester

<table>
<thead>
<tr>
<th>S.No</th>
<th>Paper Code</th>
<th>Subject</th>
<th>Hours/Week</th>
<th>No of Credits</th>
<th>Max. Marks Internal assessment</th>
<th>Max. Marks University Exam</th>
<th>Total Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td></td>
<td>English – II</td>
<td>4</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>2.</td>
<td></td>
<td>Language(H/T/S) – II</td>
<td>4</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>3.</td>
<td></td>
<td>Life Skill Course – II</td>
<td>2</td>
<td>2</td>
<td>-0-</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>4.</td>
<td></td>
<td>Skill Development Course – II</td>
<td>2</td>
<td>2</td>
<td>-0-</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>5.</td>
<td></td>
<td>Skill Development Course – III</td>
<td>2</td>
<td>2</td>
<td>-0-</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>6.</td>
<td>C4</td>
<td>Data Structures</td>
<td>4</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>C4-P</td>
<td>Data Structures Lab</td>
<td>2</td>
<td>2</td>
<td>-0-</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>7.</td>
<td>C5</td>
<td>Object Oriented Analysis &amp; Design</td>
<td>4</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>C5-P</td>
<td>Object Oriented Analysis &amp; Design Lab</td>
<td>2</td>
<td>2</td>
<td>-0-</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>8.</td>
<td>C6</td>
<td>Database Management Systems</td>
<td>4</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>C6-P</td>
<td>Database Management Systems Lab</td>
<td>2</td>
<td>2</td>
<td>-0-</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Total</strong></td>
<td><strong>32</strong></td>
<td><strong>27</strong></td>
<td><strong>125</strong></td>
<td><strong>675</strong></td>
<td><strong>800</strong></td>
</tr>
</tbody>
</table>
## III Semester

<table>
<thead>
<tr>
<th>S.No</th>
<th>Paper Code</th>
<th>Subject</th>
<th>Hours/Week</th>
<th>No of Credits</th>
<th>Max. Marks Internal assessment</th>
<th>Max. Marks University Exam</th>
<th>Total Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>English –III</td>
<td>4</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
<td></td>
</tr>
<tr>
<td>2.</td>
<td>Language(H/T/S) – III</td>
<td>4</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
<td></td>
</tr>
<tr>
<td>3.</td>
<td>Life Skill Course – III</td>
<td>2</td>
<td>2</td>
<td>-0-</td>
<td>50</td>
<td>50</td>
<td></td>
</tr>
<tr>
<td>4.</td>
<td>Life Skill Course – IV</td>
<td>2</td>
<td>2</td>
<td>-0-</td>
<td>50</td>
<td>50</td>
<td></td>
</tr>
<tr>
<td>5.</td>
<td>Skill Development Course – IV</td>
<td>2</td>
<td>2</td>
<td>-0-</td>
<td>50</td>
<td>50</td>
<td></td>
</tr>
<tr>
<td>6.</td>
<td>C7</td>
<td>Accounting and Financial Management</td>
<td>4</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>C7-P</td>
<td>Accounting and Financial Management Lab</td>
<td>2</td>
<td>2</td>
<td>-0-</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>7.</td>
<td>C8</td>
<td>Object Oriented Programming Through Java</td>
<td>4</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>C8-P</td>
<td>Object Oriented Programming Through Java Lab</td>
<td>2</td>
<td>2</td>
<td>-0-</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>8.</td>
<td>C9</td>
<td>Operating Systems</td>
<td>4</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>C9-P</td>
<td>Operating Systems Lab</td>
<td>2</td>
<td>2</td>
<td>-0-</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td></td>
<td>32</td>
<td>27</td>
<td>125</td>
<td>675</td>
<td>800</td>
</tr>
</tbody>
</table>
### IV Semester

<table>
<thead>
<tr>
<th>S. No</th>
<th>Paper Code</th>
<th>Subject</th>
<th>Hours/Week</th>
<th>No of Credits</th>
<th>Max. Marks Internal assessment</th>
<th>Max. Marks University Exam</th>
<th>Total Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>C10</td>
<td>Cyber Laws</td>
<td>4</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>C10-P</td>
<td>Cyber Laws Lab</td>
<td>2</td>
<td>2</td>
<td>-0-</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>2.</td>
<td>C11</td>
<td>Data Mining and Data Warehousing</td>
<td>4</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>C11-P</td>
<td>Data Mining and Data Warehousing Lab</td>
<td>2</td>
<td>2</td>
<td>-0-</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>3.</td>
<td>C12</td>
<td>Web Programming</td>
<td>4</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>C12-P</td>
<td>Web Programming Lab</td>
<td>2</td>
<td>2</td>
<td>-0-</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>4.</td>
<td>C13</td>
<td>Design of Object Oriented Applications</td>
<td>4</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>C13-P</td>
<td>Design of Object Oriented Applications Lab</td>
<td>2</td>
<td>2</td>
<td>-0-</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>5.</td>
<td>C14</td>
<td>Data Analytics using R</td>
<td>4</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>C14-P</td>
<td>Data Analytics using R Lab</td>
<td>2</td>
<td>2</td>
<td>-0-</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>6.</td>
<td>C15</td>
<td>Object Oriented Software Engineering</td>
<td>4</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
</tr>
<tr>
<td>Subject</td>
<td>Hours/Week</td>
<td>No of Credits</td>
<td>Max. Marks Internal assessment</td>
<td>Max. Marks University Exam</td>
<td>Total Marks</td>
<td></td>
<td></td>
</tr>
<tr>
<td>---------</td>
<td>------------</td>
<td>---------------</td>
<td>-------------------------------</td>
<td>----------------------------</td>
<td>-------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SEC1</td>
<td>4</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SEC1-P</td>
<td>2</td>
<td>2</td>
<td>-0-</td>
<td>50</td>
<td>50</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SEC2</td>
<td>4</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SEC2-P</td>
<td>2</td>
<td>2</td>
<td>-0-</td>
<td>50</td>
<td>50</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SEC3</td>
<td>4</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SEC3-P</td>
<td>2</td>
<td>2</td>
<td>-0-</td>
<td>50</td>
<td>50</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SEC4</td>
<td>4</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SEC4-P</td>
<td>2</td>
<td>2</td>
<td>-0-</td>
<td>50</td>
<td>50</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SEC5</td>
<td>4</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SEC5-P</td>
<td>2</td>
<td>2</td>
<td>-0-</td>
<td>50</td>
<td>50</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SEC6</td>
<td>4</td>
<td>3</td>
<td>25</td>
<td>75</td>
<td>100</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SEC6-P</td>
<td>2</td>
<td>2</td>
<td>-0-</td>
<td>50</td>
<td>50</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>36</td>
<td>30</td>
<td>150</td>
<td>750</td>
<td>900</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Semester-I

Computer Fundamentals and Office Tools

<table>
<thead>
<tr>
<th>Semester</th>
<th>Course Code</th>
<th>Course Title</th>
<th>Hours/Week</th>
<th>Hours</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>C1</td>
<td>Computer Fundamentals and Office Tools</td>
<td>4</td>
<td>60</td>
<td>4</td>
</tr>
</tbody>
</table>

Course Objectives:

- To introduce the concepts of computer fundamentals and their applications for the efficient use of office technology in a business environment.
- To introduce the fundamentals of computing devices and reinforce computer vocabulary, particularly with respect to personal use of computer hardware and software.
- To provide hands-on use of Word, Excel and PowerPoint.

Course Outcomes:

- Describe the usage of computers and why computers are essential components in business and society.
- Identify categories of programs, system software and applications. Organize and work with files and folders.
- Compose, format and edit a word document and working with macros.
- Create work sheets and using various functions.
- Make presentations and inserting multimedia in them.

Syllabus

UNIT – I


Basic Computer Organization: Input Unit, Output Unit, Storage Unit, Arithmetic Logic Unit, Control Unit, Central Processing Unit.

UNIT – II

**Computer Software:** Types of Software, Logical systems Architecture, Acquiring Software, Software developmental Steps, Software Engineering.

**Computer Languages:** Machine Language, Assembly Language, High Level Language, Some High Level Languages, Characteristics of good Programming Language.


UNIT –III

**MS-Word:** Features of MS-Word, MS-Word Window components, working with formatted text, Shortcut keys, Formatting documents: Selecting text, Copying & moving data, Formatting characters, changing cases, Paragraph formatting, Indents, Drop Caps, Using format painter, Page formatting, Header & footer, Bullets & numbering, Tabs, Forming tables. Finding & replacing text, go to (F5) command, proofing text (Spell check, Auto correct), Reversing actions, Macros, Inserting pictures, Hyperlinks, Mail merging, Printing documents.

UNIT IV

**MS-Excel:** Excel Features, Spreadsheets, workbooks, creating, saving & editing a workbook, Renaming sheet, cell entries (numbers, labels, and formulas), spell check, find and replace, Adding and deleting rows and columns Filling series, fill with drag, data sort, Formatting worksheet, Functions and its parts, Some useful Functions in excel (SUM, AVERAGE, COUNT, MAX, MIN, IF), Cell referencing (Relative, Absolute, Mixed), What-if analysis Introduction to charts: types of charts, creation of charts, printing a chart, printing worksheet.

UNIT V

**MS-PowerPoint:** Features of PowerPoint, Uses, components of slide, templates and wizards, using template, choosing an auto layout, using outlines, adding subheadings, editing text, formatting text, using master slide, adding slides, changing color scheme, changing background and shading, adding header and footer, adding clip arts and auto shapes. Various presentation, Working in slide sorter view (deleting, duplicating, rearranging slides), adding transition and animations to slide show, inserting music or sound on a slide, viewing slide show, Printing slides.

**Text Books:**
- Fundamentals of Computers -ReemaThareja, Oxford University Press India

**References:**
- Fundamentals of Computers – V. Rajaraman, Prentice Hall of India
- Introduction to Computers – Peter Norton Mcgraw Hill.
List of Experiments

1) Design a visiting card for managing director of a company as per the following specification.
   - Size of visiting card is $3 \frac{1}{2} \times 2$
   - Name of the company with big font
   - Phone number, Fax number and E-mail address with appropriate symbols.
   - Office and Residence address separated by a line

2) Create a table with following columns and display the result in separate cells for the following
   - Emp Name, Basic pay, DA, HRA, Total salary.
   - Sort all the employees in ascending order with the name as the key
   - Calculate the total salary of the employee
   - Calculate the Grand total salary of the employee
   - Find highest salary and
   - Find lowest salary

3) Prepare an advertisement to company requiring software professional with the following
   - Attractive page border
   - Design the name of the company using Word Art
   - Use at least one clipart.
   - Give details of the company (use bullets etc)
   - Give details of the Vacancies in each category of employee’s (Business manager, Software engineers, System administrators, Programmers, Data entry operators) qualification required.

4) Create a letterhead of a company with the following specifications
   - Name of the company on the top of the page 2 with big font and good style
   - Phone no, Fax no and E-mail address with symbols.
   - Main products manufactured by the company
   - Slogans if any should be specify in bold at the bottom

5) Create two pages of curriculum vitae of a graduate with the following specifications
   - Table to show qualifications with proper headings
   - Appropriate left and right margins
   - Format $\frac{1}{2}$ page using two-column approach about your self
   - Name on each page at the top right side
   - Page no. in the footer on the right side.

6) Write a macro format document as below
   - Line spacing “2” (double)
   - Paragraph indent of 0.1
7) Create a letter as the main document and create 10 records for the 10 persons use mail merge to create letter for selected persons among 10.

8) Create an electronic spread sheet in which you enter the following decimal numbers and convert them into octal, Hexadecimal and binary numbers and vice-versa.

   **Decimal Numbers:** 35, 68, 95, 78, 165, 225, 355, 375, 465
   **Binary Numbers:** 101, 1101, 11111, 10001, 11101111

9) Calculate the net pay of the employees following the conditions below.

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
</tr>
</thead>
</table>
   1 |   | Employee | Employee | Basic | DA | HRA | GPF | Gross | Income | Net |
       | Number | name     | pay     |      |    |     | pay | pay | tax   | pay |
   2 |   |   |   |   |   |   |   |   |      |    |

   **DA:** - 56% of the basic pay if Basic pay is greater than 20000 or else 44%.
   **HRA:** - 15% of the Basic pay subject to maximum of Rs.4000.
   **GPF:** - 10% of the basic pay.
   **INCOME TAX:** - 10% of basic if Basic pay is greater than 20000.

Find who is getting highest salary & who is get lowest salary?

10) The ABC Company shows the sales of different product For 5 years. Create BAR Graph, 3D and Pie chart for the following.

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>S.No.</td>
<td>Year</td>
<td>Pro 1</td>
<td>Pro 2</td>
<td>Pro 3</td>
<td>Pro 4</td>
</tr>
<tr>
<td>---</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
</tr>
</tbody>
</table>
   1 | 1     | 1989  | 1000  | 800   | 900   | 1000  |
   2 | 2     | 1990  | 800   | 80    | 500   | 900   |
   3 | 3     | 1991  | 1200  | 190   | 400   | 800   |
   4 | 4     | 1992  | 400   | 200   | 300   | 1000  |
   5 | 5     | 1993  | 1800  | 400   | 400   | 1200  |

11) Create a suitable examination database and find the sum of the marks (total) of each student and respective, class secured by the student.

   **Pass:** if marks in each subject >=35
   **Distinction:** if average >=75
   **First class:** if average >=60 but <75
   **Second class:** if average >=50 but less than 60
   **Third class:** if average>=35 but less than 50
   **Fail:** if marks in any subject <35
12) Enter the following data into the sheet.

<table>
<thead>
<tr>
<th>Name</th>
<th>Department</th>
<th>Salary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anusha</td>
<td>Accounts</td>
<td>12000</td>
</tr>
<tr>
<td>Rani</td>
<td>Engineering</td>
<td>24000</td>
</tr>
<tr>
<td>Lakshmi</td>
<td>Accounts</td>
<td>9000</td>
</tr>
<tr>
<td>Purnima</td>
<td>Marketing</td>
<td>20000</td>
</tr>
<tr>
<td>Bindu</td>
<td>Accounts</td>
<td>4500</td>
</tr>
<tr>
<td>Tejaswi</td>
<td>Accounts</td>
<td>11000</td>
</tr>
<tr>
<td>Swetha</td>
<td>Engineering</td>
<td>15000</td>
</tr>
<tr>
<td>Saroja</td>
<td>Marketing</td>
<td>45000</td>
</tr>
<tr>
<td>Sunitha</td>
<td>Accounts</td>
<td>5600</td>
</tr>
<tr>
<td>Sandhya</td>
<td>Engineering</td>
<td>24000</td>
</tr>
<tr>
<td>Harika</td>
<td>Marketing</td>
<td>8000</td>
</tr>
</tbody>
</table>

- Extract records for department in Accounts and Salary >10000
- Sort the data by salary with the department using “sort commands”.
- Calculate total salary for each department using Subtotals

13) Enter the following data into the sheet.

<table>
<thead>
<tr>
<th></th>
<th>Raju</th>
<th>Rani</th>
<th>Mark</th>
<th>Rosy</th>
<th>Ismail</th>
<th>Reshma</th>
</tr>
</thead>
<tbody>
<tr>
<td>English</td>
<td>76</td>
<td>89</td>
<td>43</td>
<td>51</td>
<td>76</td>
<td>87</td>
</tr>
<tr>
<td>2nd Lang</td>
<td>55</td>
<td>85</td>
<td>78</td>
<td>61</td>
<td>47</td>
<td>33</td>
</tr>
<tr>
<td>Maths</td>
<td>65</td>
<td>82</td>
<td>34</td>
<td>58</td>
<td>52</td>
<td>65</td>
</tr>
<tr>
<td>Computers</td>
<td>45</td>
<td>91</td>
<td>56</td>
<td>72</td>
<td>49</td>
<td>56</td>
</tr>
<tr>
<td>Human Values</td>
<td>51</td>
<td>84</td>
<td>54</td>
<td>64</td>
<td>32</td>
<td>64</td>
</tr>
</tbody>
</table>

Apply the conditional formatting for marks
- 35 below Red
- 35 to 50 Blue
- 51 to 70 Green
- 71 to 100 Yellow

14) Create a presentation using templates.
15) Create a Custom layout or Slide Master for professional presentation.
16) Create a presentation with slide transitions and animation effects.
17) Create a table in PPT and apply graphical representation on it.
PROGRAMMING IN C

<table>
<thead>
<tr>
<th>Semester</th>
<th>Course Code</th>
<th>Course Title</th>
<th>Hours/Week</th>
<th>Hours</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>C2</td>
<td>PROGRAMMING IN C</td>
<td>4</td>
<td>60</td>
<td>4</td>
</tr>
</tbody>
</table>

Course Objectives:

- Provides knowledge on Algorithms, Flow chart and different programming languages.
- To train the students with basic concepts of programming using C.
- Provides complete knowledge of C language.
- Helps to develop logics which will help them to create program and applications in C.
- Learning the basic programming constructs, they can easily switch over to any other language in future.

Course Outcomes:

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

- Understand the basic terminology used in computer programming.
- Write, compile and debug programs in C language.
- Use different data types in a computer program.
- Design programs involving decision structures, loops and functions.
- Understand the dynamics of memory by the use of pointers and Structures.
- Apply different operations in File handling.

Syllabus

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

UNIT - II:
UNIT - III:
Arrays and Strings: Arrays: Introduction, Declaration of Arrays, Accessing elements of the Array – Storing Values in Array, Calculating the length of the Array, Operations that can be performed on Array, One dimensional array, Accessing one dimensional array, Passing one dimensional array to function, Two dimensional Arrays, Accessing two dimensional arrays, Passing two dimensional arrays to functions. Strings: Introduction, String Operations using String functions.

UNIT - IV:

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

PRESCRIBED TEXT BOOKS:
Computer Fundamentals and Programming in C by REEMA THAREJA from OXFORD UNIVERSITY PRESS

REFERENCE BOOKS:
4. Teach your C Skills-Kanithker
List of Experiments

1. Write a C program to calculate the expression: \(((a\times b)/c) + (a+b-c)\) .
2. Write a C program to calculate \((a+b+c)^3\) .
3. Write a C program to check whether the given number is Prime or Not.
4. Write a C program to find the sum of individual digits of a given number.
5. Program to convert Hours into seconds.
6. Write a C program to generate all the prime numbers between 1 and n, where n is a value supplied by the user.
7. Write a program to check whether given number is Palindrome or Not.
8. Write a C program to check whether a given 3-digit number is Armstrong number or not.
9. Write a C program to print the numbers in triangular form.
   
   1
   1 2
   1 2 3
   1 2 3 4
10. Program to display number of days in given month using Switch – Case.
11. Write a C program to perform the following:
    i. Addition of two matrices.
    ii. Multiplication of two matrices.
12. Write a C program to determine if the given string is a palindrome or not.
13. Write C program to find the factorial of a given integer using recursive function.
14. Write a C program to concatenate two strings using pointers.
15. Write a C program to find the length of string using pointers.
16. Program to display Student Details using Structures.
17. Write a C program to
    i. Write data into a File.
    ii. Read data from a File.
NUMERICAL AND STATISTICAL METHODS

<table>
<thead>
<tr>
<th>Semester</th>
<th>Course Code</th>
<th>Course Title</th>
<th>Hours/Week</th>
<th>Hours</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>C3</td>
<td>NUMERICAL AND STATISTICAL METHODS</td>
<td>4</td>
<td>60</td>
<td>4</td>
</tr>
</tbody>
</table>

Course Objectives:

- To learn how to perform error analysis for arithmetic operations.
- To demonstrate working of various numerical methods.
- To provide a basic understanding of the derivation and use of methods of interpolation and numerical integration.
- To impart knowledge of various statistical techniques.
- To develop students understanding through laboratory activities to solve problems related to above stated concepts.

Course Outcomes:

- Skill to choose and apply appropriate numerical methods to obtain appropriate solutions to difficult mathematical problems.
- Ability to apply various statistical techniques such as Measures of Central Tendency and Dispersion.
- Understanding of relationship between variables using the method of Correlation and Fit Analysis.
- Skill to execute programs of various Numerical Methods and Statistical techniques for solving mathematical problems.

Syllabus

UNIT 1: Solution of equations (polynomial and transcendental equations) interval having methods, secant, Regula – Falsi, Newton – Raphson methods, Fixed point Iteration method.

UNIT 3: Interpolation: Forward and backward differences, Newton’s forward and backward formula, Lagrange’s interpolation and Lagrange’s inverse interpolation formula.

Numerical differentiation, integration: Numerical differentiation forward and backward formula, Trapezoidal and Simpsons formulas.

Statistical Methods:

UNIT 4: Basic concepts and definition of statistics: Mean, Median, Mode, standard deviation, coefficient of variation, skewness and kurtosis, Karl Pearson Correlation coefficient, Rank Correlation and illustrated examples.

UNIT 5: Probability: Basic concepts and definition of probability, Probability axioms, Conditional probability, Addition and Multiplication theorem of probability (Based on set theory concepts), Bayes theorem, problems and applications.

TEXT BOOKS:

1. Sunil S. Patil Numerical and Statistical Methods EBPB.

REFERENCE BOOKS:

List of Experiments

Unit – 1

1) Find the root of the Equation \( x^2 = 1.2 \) by using Regula - Falsi method.
2) Solve the Equation \( \sin x = 5x - 2 \) by Iteration method.
3) Apply Newton - Raphson method, to find and approximate root, correct to three decimal places, of the Equation \( x^3 - 3x - 5 = 0 \), which lies near \( x = 2 \).
4) Find the root of the Equation \( x \sin x + \cos x = 0 \) by using Newton – Raphson method.
5) Find the root of the Equation \( x^3 + x - 1 = 0 \) by Iteration method, given that a root lies near 1.

Unit – 2

1) Solve the system of Equations \( 3x+y-z = 3, \ 2x-8y+z=-5, \ x-2y+9z=8 \) using Gauss – Elimination method.
2) Using Gauss –Jordan method solve the system, \( 2x+y+z=10, \ 3x+2y+3z = 18, \ x+4y+9z=16. \)
3) Solve the Equations \( 2x+3y+z=9, \ x+2y+3z=6, \ 3x+y+2z=8 \) by LU – Decomposition method.
4) Solve the system of Equations \( , 8x-3y+2z=20, \ 4x+11y-z=33, \ 6x+3y+12z=35 \) by using Gauss- Seidel method.

Unit – 3

1) Using Newton’s forward interpolation formula, the given table of values,

<table>
<thead>
<tr>
<th>X</th>
<th>1.1</th>
<th>1.3</th>
<th>1.5</th>
<th>1.7</th>
<th>1.9</th>
</tr>
</thead>
<tbody>
<tr>
<td>f(x)</td>
<td>0.21</td>
<td>0.69</td>
<td>1.25</td>
<td>1.89</td>
<td>2.61</td>
</tr>
</tbody>
</table>

Obtain the value of \( f(x) \) when \( x = 1.4 \)
2) Using Lagrange’s Interpolation formula, find the value of $y$, corresponding to $x = 10$

3) from the following table

<table>
<thead>
<tr>
<th>X</th>
<th>5</th>
<th>6</th>
<th>9</th>
<th>11</th>
</tr>
</thead>
<tbody>
<tr>
<td>Y</td>
<td>12</td>
<td>13</td>
<td>14</td>
<td>16</td>
</tr>
</tbody>
</table>

4) From the following table of values of $x$ & $y$, obtain $\frac{dy}{dx}$ and $\frac{d^2y}{dx^2}$ for $x = 1.5$

<table>
<thead>
<tr>
<th>X</th>
<th>1.5</th>
<th>2.0</th>
<th>2.5</th>
<th>3.0</th>
<th>4.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Y</td>
<td>3.375</td>
<td>7.0</td>
<td>13.625</td>
<td>24.0</td>
<td>38.875</td>
</tr>
</tbody>
</table>

5) Evaluate $\int_0^1 x^3 \, dx$ with five sub-intervals by Trapezoidal rule.

6) Evaluate $\int_0^1 \frac{1}{1+x} \, dx$ using Simpson’s 3/8 rule taking $h = 1/6$.

---

**Unit – 4**

1) Find the Karl Pearson’s coefficient of skewness for the following data:

<table>
<thead>
<tr>
<th>Marks</th>
<th>0-10</th>
<th>10-20</th>
<th>20-30</th>
<th>30-40</th>
<th>40-50</th>
<th>50-60</th>
<th>60-70</th>
</tr>
</thead>
<tbody>
<tr>
<td>No.of students</td>
<td>10</td>
<td>12</td>
<td>18</td>
<td>25</td>
<td>16</td>
<td>14</td>
<td>8</td>
</tr>
</tbody>
</table>

2) Find Bowley’s coefficient of skewness for the following data:

<table>
<thead>
<tr>
<th>Salary</th>
<th>500-600</th>
<th>600-700</th>
<th>700-800</th>
<th>800-900</th>
<th>900-1000</th>
<th>1000-1100</th>
<th>1100-1200</th>
<th>1200-1300</th>
</tr>
</thead>
<tbody>
<tr>
<td>No.of persons</td>
<td>10</td>
<td>28</td>
<td>40</td>
<td>64</td>
<td>25</td>
<td>18</td>
<td>9</td>
<td>6</td>
</tr>
</tbody>
</table>

3) Find the standard deviation from Assumed mean method for the following data:

<table>
<thead>
<tr>
<th>Class interval</th>
<th>0-10</th>
<th>10-20</th>
<th>20-30</th>
<th>30-40</th>
<th>40-50</th>
<th>50-60</th>
<th>60-70</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frequency</td>
<td>1</td>
<td>4</td>
<td>17</td>
<td>45</td>
<td>26</td>
<td>5</td>
<td>2</td>
</tr>
</tbody>
</table>

4) Find the coefficient of skewness for the following data:

<table>
<thead>
<tr>
<th>Variable</th>
<th>0-5</th>
<th>5-10</th>
<th>10-15</th>
<th>15-20</th>
<th>20-25</th>
<th>25-30</th>
<th>30-35</th>
<th>35-40</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frequency</td>
<td>2</td>
<td>5</td>
<td>7</td>
<td>13</td>
<td>21</td>
<td>16</td>
<td>8</td>
<td>3</td>
</tr>
</tbody>
</table>

5) Find the rank correlation coefficient for the following data:

<table>
<thead>
<tr>
<th>X</th>
<th>65</th>
<th>45</th>
<th>67</th>
<th>38</th>
<th>48</th>
<th>50</th>
<th>26</th>
<th>47</th>
<th>70</th>
<th>62</th>
</tr>
</thead>
<tbody>
<tr>
<td>Y</td>
<td>64</td>
<td>40</td>
<td>58</td>
<td>46</td>
<td>52</td>
<td>49</td>
<td>38</td>
<td>47</td>
<td>59</td>
<td>60</td>
</tr>
</tbody>
</table>
1) Three dice are tossed together. Find the probability that exactly two of the three numbers that show on them are equal.

2) What is the probability that a card drawn at random from the pack of playing cards may be either a Queen or a Jack?

3) If two cards are drawn from a well shuffled pack, find the probability that at least one of the two is Hearts.

4) A bag contains 4 Red, 6 Blue balls and a second bag contains 4 Blue & 6 Green balls. A ball is taken out from each bag. Find the probability that one ball is red and the other ball is Green. The probability that an event A happens in one trial of an experiment is 0.4. Three independent trails of the experiment are performed. Find the probability that the event A happens at least once.
Course Objectives:

- The objective of the course is to make a student to implement data structures and organize data based on data structures for efficient access.

Course Outcomes:

- Identify data structures suitable to solve problems.
- Developing algorithms.
- Identifying the use of Time and Space Complexity.
- Implementing different sorting & searching techniques.

Syllabus

Unit I


**Preliminaries**- Mathematical Notation and Functions, Algorithmic Notation, Control Structures, Complexity of Algorithms. Other Asymptotic Notations, Sub algorithms, Variables, Data Types.

Unit II

**String Processing** – Storing Strings, Character Data Type, String Operations, Word Processing, Pattern Matching Algorithms.

**Arrays, Records and Pointers** – Linear Arrays, Representation and Traversing Linear Arrays, Inserting and Deleting, Bubble Sort, Linear Search, Binary Search, Multidimensional Arrays, Pointer Arrays, Record Structures, Representation of records in memory, Parallel Arrays, Matrices, Sparse Matrices.

Unit III

**Linked Lists** – Representation, Traversing, Searching, Memory Allocation: Garbage Collection, Insertion, Deletion, Header Linked Lists, Two-Way Lists

**Stacks, Queues, Recursion**- Stacks, Array representation, Linked List representation, Arithmetic Expressions; Polish notation, Quick sort, Recursion, Towers of Hanoi,
Implementation of recursive procedures by stacks, Queues, Linked representation of Queues, Dqeques, Priority Queues.

Unit IV

Trees- Binary trees, Representing and traversing binary trees, Traversal algorithms using stacks, Header nodes, Binary Search Trees, Searching, Insertion and Deletion in Binary Search Trees, AVL Search Trees, Insertion and Deletion in AVL trees, m-way search trees, searching, insertion and deletion in m-way search tree, B Trees, searching, insertion and deletion in a B-tree, Heap: Heap Sort, Huffman’s Algorithms, General Trees

Unit V

Graphs- Terminology, Sequential representation of Graphs, Warshall’s Algorithm, Linked representation of Graphs, Operations on Graphs, Traversing a Graph, Topological Sorting

Sorting and Searching- Insertion Sort, Selection sort, Merging, Merge sort, Radix sort, Searching and Data modification, Hashing

Text books:

REFERENCE BOOKS:
2. Theory and Problems of Data Structures by Seymour Lipschutz, McGraw Hill (Schaum’s Outlines)
# Data Structures Lab

<table>
<thead>
<tr>
<th>Semester</th>
<th>Course Code</th>
<th>Course Title</th>
<th>Hours/Week</th>
<th>Hours</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>II</td>
<td>C4-P</td>
<td>Data Structures Lab</td>
<td>2</td>
<td>30</td>
<td>1</td>
</tr>
</tbody>
</table>

## List of Lab Experiments

1. Write a C program to implement matrix multiplication.
2. Write a C program to implement stack using arrays.
3. Write a C program to implement queue using arrays.
4. Write a C program to implement circular queue using arrays.
5. Write a C program to implement dequeue using arrays.
6. Write a C program to implement single linked list using the methods `create()`, `insert()`, `search()`, `delete()` and `display()`.
7. Write a C program to implement double linked list.
8. Write a C program to implement stack using linked list.
9. Write a C program to implement queue using linked list.
10. Give a solution to towers of Hanoi using C program.
11. Write a C program to implement bubble sort.
12. Write a C program to implement selection sort.
13. Write a C program to implement insertion sort.
14. Write a C program to implement merge sort.
15. Write a C program to implement quick sort.
OBJECT ORIENTED ANALYSIS AND DESIGN

<table>
<thead>
<tr>
<th>Semester</th>
<th>Course Code</th>
<th>Course Title</th>
<th>Hours/Week</th>
<th>Hours</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>II</td>
<td>C5</td>
<td>OBJECT ORIENTED ANALYSIS AND DESIGN</td>
<td>4</td>
<td>60</td>
<td>4</td>
</tr>
</tbody>
</table>

Course Objectives:

- To provide a sound understanding of the fundamental concepts and historical evolution of the model.
- To facilitate a mastery of the notation and process of object-oriented analysis and design.

Course Outcomes:

By the end of the course, student will be able to:
- Have Knowledge in evolution and foundations of OO Model and its elements.
- Identify relationship between classes and objects.
- Know importance of classification and can identify classes and objects.
- Have basic knowledge of UML.
- Knowledge in syntax and semantics of UML.

Syllabus

UNIT I:

UNIT II:


UNIT III:

**Classification** - The *Importance of Proper Classification*: The Difficulty of Classification, The Incremental and Iterative Nature of Classification. *Identifying classes and Objects*: Classical and Modern Approaches. *Object Oriented Analysis*: Classical Approaches, Behavior Analysis, Domain Analysis, Use Case Analysis, CRC Cards, Informal English Description, Structured Analysis. *Key Abstractions and Mechanisms*: Identifying Key Abstractions: Refining Key Abstractions, Naming Key Abstractions. Identifying Mechanisms.

UNIT IV:


UNIT V:


**TEXT BOOK:**

**REFERENCE BOOKS:**

List of Lab Experiments

1. Demonstrate Package Diagram for Hydroponics Gardening system.
2. Demonstrate Component Diagram for the Environmental control system.
3. Demonstrate Deployment Diagram for Environmental control system.
4. Identify Use Cases and develop the Use Case Diagram for Hydroponics Gardening system.
5. Demonstrate Activity Diagram for Hydroponics Gardening system.
6. Demonstrate Class Diagram for the Environmental control system.
7. Demonstrate sequence diagram Environmental controller system.
8. Demonstrate sequence diagram for returning and removing books for the library system.
9. Demonstrate use case for returning book with fine for library system.
10. Draw the State Machine Diagram for the Duration Timer.
11. Draw the Interaction Diagram for Library system.
12. Draw the Composite structure diagram for the Hydroponics Gardening system’s water tank.
13. Draw the Timing Diagram for a valve object that is controlled to fill the Water storage tank object in Hydroponics Gardening system.
14. Demonstrate Object Diagram for the library system.
15. Draw the Communication Diagram for the Hydroponics Gardening system.
DATA BASE MANAGEMENT SYSTEM

<table>
<thead>
<tr>
<th>Semester</th>
<th>Course Code</th>
<th>Course Title</th>
<th>Hours/Week</th>
<th>Hours</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>II</td>
<td>C6</td>
<td>DATA BASE MANAGEMENT SYSTEM</td>
<td>4</td>
<td>60</td>
<td>4</td>
</tr>
</tbody>
</table>

Course Objectives:

- Graduates will have the expertise in analyzing real time problems and providing appropriate solutions related to Computer Science & Engineering.
- Graduates will have the knowledge of fundamental principles and innovative technologies to succeed in higher studies and research.
- Graduates will continue to learn and to adapt technology developments combined with deep awareness of ethical responsibilities in profession.

Course Outcomes:

- An ability to apply Knowledge of computing and mathematics in Computer Science & Engineering.
- An ability to analyze a problem, identify and define the computing requirements appropriate to its solution.
- An ability to design, implement and evaluate a computer-based system to meet desired needs with appropriate societal considerations.
- An ability to conduct investigations, interpret data and provide conclusions in investigating complex problems related to Computer Science & Engineering.
- An ability to engage in continuing professional development and life-long learning.

Syllabus

UNIT-I

Overview of Database Systems: Managing Data, File System versus DBMS, Advantages of DBMS, Describing and storing Data in a DBMS, Queries in DBMS, Transaction Management, Structure of DBMS, People who work with DBMS

Introduction to Database design: ER diagrams, Beyond ER Design, Entities, Attributes and Entity sets, Relationships and Relationship sets, Additional features of ER Model, Conceptual Design with the ER Model.

UNIT-II

Relational Model: Introduction to the Relational Model - Integrity Constraints over Relations, Enforcing Integrity constraints, Querying relational data, Logical data base Design, Introduction to Views destroying/altering Tables and Views.
**Relational Algebra and Calculus:** Relational Algebra - Selection and Projection, Set operations, Renaming, Joins, Division, Examples of Algebra Queries, Relational calculus - Tuple relational Calculus - Domain relational calculus - Expressive Power of Algebra and calculus.

**UNIT-III**

**Working with MySQL - Introduction:** What is MySQL?, Database Terms, Install MySQL on windows, Install MySQL on Ubuntu, Access MySQL, Start the Command Line Interface.  
**MySQL Command Syntax:** SQL Commands Syntax, Data Types Create a Database, Create a User, Create a Table.  
**Important SQL Commands:** Insert New Records, Modify a Table, Query a Database, Advanced Select Statements, Remove a Row, Limit Clause, Update the contents of a Field, Sort Results.  
**Logical Operators**  
**Advanced Usage:** Primary Keys, MySQL Functions, MySQL string functions, MySQL date functions, MySQL aggregate functions  

**Basics of Functional Dependencies and Normalization for Relational Databases:** Informal Design Guidelines for Relational Schemas, Functional Dependencies, Normal Forms Based on Primary Keys, General Definitions of Second and Third Normal Forms, Boyce-Codd Normal Form, Multivalued Dependency and Fourth Normal Form, Join Dependencies and Fifth Normal Form  

**UNIT IV**

**Introduction to Transaction Processing Concepts and Theory:** Introduction to Transaction Processing – Transaction and System Concepts – Desirable Properties of Transactions – Characterizing schedules based on Serializability – Characterizing schedules based on Recoverability  

**Concurrency Control Techniques:** Two Phase Locking Techniques for Concurrency Control – Concurrency Control Based on Timestamp ordering – Multiversion Concurrency Control Techniques  

**UNIT-V**


**Indexing Structures for Files:** Types of Single-Level Ordered Indexes, Multilevel Indexes, Dynamic Multilevel Indexes Using B-Trees and B+ Trees, Indexes on Multiple Keys, Other Types of Indexes.  

**TEXT BOOKS:**

REFERENCES BOOKS:

List of Experiments

**Cycle-I**: Aim: Marketing company wishes to computerize their operations by using following tables.

**Table Name: Client- Master**
**Description:** Used to store client information

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Data Type</th>
<th>Size</th>
<th>Attribute</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLIENT_ NO</td>
<td>Varchar2</td>
<td>6</td>
<td>Primary key and first letter must start with ‘C’</td>
</tr>
<tr>
<td>NAME</td>
<td>Varchar2</td>
<td>20</td>
<td>Not null</td>
</tr>
<tr>
<td>ADDRESS 1</td>
<td>Varchar2</td>
<td>30</td>
<td></td>
</tr>
<tr>
<td>ADDRESS S</td>
<td>Varchar2</td>
<td>30</td>
<td></td>
</tr>
<tr>
<td>CITY</td>
<td>Varchar2</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>PINCODE</td>
<td>Varchar2</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>STATE</td>
<td>Varchar2</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>BAL_DUE</td>
<td>Number</td>
<td>10,2</td>
<td></td>
</tr>
</tbody>
</table>

**Table Name: Product_ Master**
**Description:** Used to store product information

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Data Type</th>
<th>Size</th>
<th>Attribute</th>
</tr>
</thead>
<tbody>
<tr>
<td>PRODUCT _NO</td>
<td>Varchar2</td>
<td>6</td>
<td>Primary key and first letter must start with ‘P’</td>
</tr>
<tr>
<td>DESCRIPTION</td>
<td>Varchar2</td>
<td>15</td>
<td>Not null</td>
</tr>
<tr>
<td>PROFIT _PERCENT</td>
<td>Number</td>
<td>4,2</td>
<td>Not null</td>
</tr>
<tr>
<td>UNIT _MEASUE</td>
<td>Varchar2</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>QTY_ ON_ HAND</td>
<td>Number</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>REORDER_ LVL</td>
<td>Number</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>SELL _PRICE</td>
<td>Number</td>
<td>8, 2</td>
<td>Not null, cannot be 0</td>
</tr>
</tbody>
</table>
Table Name: Salesman_master
Description: Used to store salesman information working for the company.

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Data Type</th>
<th>Size</th>
<th>Attribute</th>
</tr>
</thead>
<tbody>
<tr>
<td>SALESMAN_NO</td>
<td>Varchar2</td>
<td>6</td>
<td>Primary key and first letter must start with 'S'</td>
</tr>
<tr>
<td>SALESMAN_NAME</td>
<td>Varchar2</td>
<td>20</td>
<td>Not null</td>
</tr>
<tr>
<td>ADDRESS1</td>
<td>Varchar2</td>
<td>30</td>
<td></td>
</tr>
<tr>
<td>ADDRESS2</td>
<td>Varchar2</td>
<td>30</td>
<td></td>
</tr>
<tr>
<td>CITY</td>
<td>Varchar2</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>PINCODE</td>
<td>Number</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>STATE</td>
<td>Varchar2</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>SAL_AMT</td>
<td>Number</td>
<td>8,2</td>
<td>Not null, cannot be 0</td>
</tr>
<tr>
<td>TGT_TO_GET</td>
<td>Number</td>
<td>6,2</td>
<td>Not null, cannot be 0</td>
</tr>
<tr>
<td>YTD_SALES</td>
<td>Number</td>
<td>6,2</td>
<td>Not null</td>
</tr>
<tr>
<td>REMARKS</td>
<td>Varchar2</td>
<td>20</td>
<td></td>
</tr>
</tbody>
</table>

Table Name: SALES-ORDER
Description: Used to store client’s orders

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Data Type</th>
<th>Size</th>
<th>Attribute</th>
</tr>
</thead>
<tbody>
<tr>
<td>ORDER_NO</td>
<td>Varchar2</td>
<td>6</td>
<td>Primary key and first letter must start with 'S'</td>
</tr>
<tr>
<td>CLIENT_NO</td>
<td>Varchar2</td>
<td>6</td>
<td>Foreign Key</td>
</tr>
<tr>
<td>ORDER_DATE</td>
<td>Date</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DELY_ADDRESS</td>
<td>Varchar2</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>SALESMAN_NO</td>
<td>Varchar2</td>
<td>6</td>
<td>Foreign Key</td>
</tr>
<tr>
<td>DELY_TYPE</td>
<td>Char</td>
<td>1</td>
<td>Delivery: part(p)/ full(f) and default 'F'</td>
</tr>
<tr>
<td>BILL_YN</td>
<td>Char</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>DELY_DATE</td>
<td>Date</td>
<td></td>
<td>Can’t be less than order date</td>
</tr>
</tbody>
</table>

Table Name: SALES_ORDER_DETAILS
Description: Used to store client’s order with details of each product ordered.
Solve the following queries by using above tables.

1. Retrieve the list of names, city and the state of all the clients.
2. List all the clients who are located in ‘Mumbai’ or ‘Bangalore’.
3. List the various products available from the product_master table.
4. Find the names of sales man who have a salary equal to Rs.3000.
5. List the names of all clients having ‘a’ as the second letter in their names.
6. List all clients whose Bal due is greater than value 1000.
7. List the clients who stay in a city whose first letter is ‘M’.
8. List all information from sales-order table for orders placed in the month of July.
9. List the products whose selling price is greater than 1000 and less than or equal to 3000.
10. Find the products whose selling price is greater than 1000 and also find the new selling price as original selling price 0.50.
11. Find the products in the sorted order of their description.
12. Find the products with description as ‘540HDD’ and ‘Pen drive’.
13. Count the total number of orders.
14. Print the description and total qty sold for each product.
15. Calculate the average qty sold for each client that has a maximum order value of 15,000.
16. Find all the products whose quantity on hand is less than reorder level.
17. List the order number and day on which clients placed their order.
18. Find out the products and their quantities that will have to deliver in the current month.
19. Find the names of clients who have placed orders worth of 10000 or more.
20. Find the client names who have placed orders before the month of June, 2008.

Cycle-II

Aim: A manufacturing company deals with various parts and various suppliers supply these parts. It consists of three tables to record its entire information. Those are as follows.

Supplier (Supplier_No, Sname, City, status)
Part(Part_no, pname, color, weight, city, cost)
Shipment (supplier_No, Part_no, city)
JX(project_no, project_name, city)
SPJX (Supplier_no, part_no, project_no, city)

2. Get project names for projects supplied by supplier S.
3. Get colors of parts supplied by supplier S1.
4. Get part numbers for parts supplied to any project in Mumbai.
5. Find the id’s of suppliers who supply a red or pink parts.
6. Find the pnames of parts supplied by London supplier and by no one else.
7. Get the names of the parts supplied by the supplier ‘Mart’ and ‘Miller’.
8. Get supplier names for suppliers who do not supply part P2.
9. Get all pairs of supplier numbers such that the suppliers concerned are “colocated”.

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Data Type</th>
<th>Size</th>
<th>Attribute</th>
</tr>
</thead>
<tbody>
<tr>
<td>ORDER_NO</td>
<td>Varchar2</td>
<td>6</td>
<td>Primary key references SALES_ORDER table</td>
</tr>
<tr>
<td>PRODUCT_NO</td>
<td>Varchar2</td>
<td>6</td>
<td>Foreign Key references SALES_ORDER_table</td>
</tr>
<tr>
<td>QTY_ORDERED</td>
<td>Number</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>QTY_DISP</td>
<td>Number</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>PRODUCT_RATE</td>
<td>Number</td>
<td>10,2</td>
<td>Foreign Key</td>
</tr>
</tbody>
</table>
10. Get suppliers names for the suppliers who supply at least one red part.

**Cycle –III Employee Database**

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

Emp(Empno, Ename, Job, Mgr, Hiredate, Sal, Comm, Deptno)

Dept(Deptno, Dname, Loc)

1. List the details of employees who have joined before the end of September’ 81.
2. List the name of the employee and designation of the employee, who does not report to anybody.
3. List the name, salary and PF amount of all the employees (PF is calculated as 10% of salary)
4. List the names of employees who are more than 2 years old in the organization.
5. Determine the number of employees, who are taking commission.
6. Update the employee salary by 20% , whose experience is greater than 12 years.
7. Determine the department does not contain any employees.
8. Create a view, which contains employee name and their manager names working in sales department.
9. Determine the employees, whose total salary is like the minimum salary of any department.
10. List the department numbers and number of employees in each department.
11. Determine the employees, whose total salary is like the minimum salary of any department.
12. List average salary for all departments employing more than five people.
13. Determine the names of employees, who take highest salary in their departments.
14. Determine the names of employees, who earn more than their managers.
15. Display ename, dname, even if no employee belongs to that department (use outer join)

**Cycle- IV**

An Airline system would like to keep track their information by using the following relations.

FLIGHTS(fl_no: integer, from: string, to: string, distance: integer, price: integer)

AIRCRAFT(aid: integer, aname: string, cruising_range: integer)

CERTIFIED(eid: integer, aid: integer)

Employees(eid: integer, ename: string, salary: real)

Note that the employees relation describes pilots and other kinds of employees as well; every pilot is certified for aircraft and only pilots are certified to fly. Resolve the following queries.

1. Find the names of pilots whose salary is less than the price of the cheapest route from Newyork to Chicago.
2. For each pilot who is certified for more than 2 aircraft, find the eid’s and the maximum cruising range of the aircraft that he or she certified for.
3. For all aircraft with cruising range over 1,500 miles, find the name of the aircraft and the average salary of all pilots certified for this aircraft.
4. Find the aid’s of all aircraft than can be used from chicaga to LosAngels.
5. Find the name of the pilots certified from some Boeing aircraft.
6. Print the names of pilots who can operate planes with cruising range greater than 3,500 miles, but are not certified by Boeing aircraft.
7. Find the eid’s of employees who are certified for exactly 2 aircrafts.
8. Find the total amount paid to employees as salaries.
9. Find the aid’s of all than can be used on non-stop flights from Chennai to Dubai.
10. Find the eid’s of employee who make second highest salary.

PL/SQL PROGRAMS

1. Write a PL/SQL program to check the given number is strong or not.
2. Write a PL/SQL program to check the given string is palindrome or not.
3. Write a PL/SQL program to swap two numbers without using third variable.
4. Write a PL/SQL program to generate multiplication tables for 2, 4, 6.
5. Write a PL/SQL program to check the given number is Armstrong or not.
6. Write a PL/SQL code to find the factorial of any number.
7. Write a PL/SQL program to display sum of even numbers and sum of odd numbers in the given range.
8. Write a PL/SQL program to check the given number is palindrome or not.
9. The HRD manager has decide to raise the employee salary by 15% write a PL/SQL block to accept the employee number and update the salary of that employee. Display appropriate message based on the existence of the record in Emp table.
10. Write a PL/SQL program to display to 10 rows in Emp table based on their job and salary.
11. Write a PL/SQL program to raise the employee salary by 10% for department number 30 people and also maintain the raised details in the raise table.
12. Write a procedure to update the salary of Employee, who are not getting commission by 10%.
13. Write a PL/SQL procedure to prepare an electricity bill by using following table.
   Table used: Elect

<table>
<thead>
<tr>
<th>Name</th>
<th>Null?</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>MNNO</td>
<td>NOT NULL</td>
<td>NUMBER(3)</td>
</tr>
<tr>
<td>CNAME</td>
<td></td>
<td>VARCHAR2(20)</td>
</tr>
<tr>
<td>CUR_READ</td>
<td></td>
<td>NUMBER(5)</td>
</tr>
<tr>
<td>PREV_READ</td>
<td></td>
<td>NUMBER(5)</td>
</tr>
<tr>
<td>NO_UNITS</td>
<td></td>
<td>NUMBER(5)</td>
</tr>
<tr>
<td>AMOUNT</td>
<td></td>
<td>NUMBER(8,2)</td>
</tr>
<tr>
<td>SER_ TAX</td>
<td></td>
<td>NUMBER(8,2)</td>
</tr>
<tr>
<td>NET_AMT</td>
<td></td>
<td>NUMBER(9,2)</td>
</tr>
</tbody>
</table>

14. Write a PL/SQL program to prepare an telephone bill by using following table and print the monthly bills for each customer.

Table used: Phone

<table>
<thead>
<tr>
<th>Name</th>
<th>Null?</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>TEL_NO</td>
<td>NOT NULL</td>
<td>NUMBER(6)</td>
</tr>
<tr>
<td>CNAME</td>
<td></td>
<td>VARCHAR2(20)</td>
</tr>
<tr>
<td>CITY</td>
<td></td>
<td>VARCHAR2(10)</td>
</tr>
<tr>
<td>PR_READ</td>
<td></td>
<td>NUMBER(5)</td>
</tr>
<tr>
<td>CUR_READ</td>
<td></td>
<td>NUMBER(5)</td>
</tr>
<tr>
<td>NET_AMT</td>
<td></td>
<td>NUMBER(5)</td>
</tr>
<tr>
<td>TOT_AMT</td>
<td></td>
<td>NUMBER(8,2)</td>
</tr>
</tbody>
</table>
15. Write a PL/SQL program to raise the employee salary by 10 %, who are completed their
25 years of service and store the details at appropriate tables
(Define the Retair_Emp_Table)

16. Write a PL/SQL program to evaluate the grade of a student with following conditions:
   For pass: all marks > 40
   For I class: Total % > 59
   For II Class: Total % between >40 and < 60
   For III class: total % = 40
   And also maintain the details in abstract table.

1. Table STD

<table>
<thead>
<tr>
<th>Name</th>
<th>Null?</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>NO</td>
<td>NOT NULL</td>
<td>NUMBER</td>
</tr>
<tr>
<td>NAME</td>
<td></td>
<td>VARCHAR2(10)</td>
</tr>
<tr>
<td>INTNO</td>
<td>NOT NULL</td>
<td>NUMBER</td>
</tr>
<tr>
<td>CLASS</td>
<td>NOT NULL</td>
<td>VARCHAR2(10)</td>
</tr>
<tr>
<td>M1</td>
<td></td>
<td>NUMBER</td>
</tr>
<tr>
<td>M2</td>
<td></td>
<td>NUMBER</td>
</tr>
<tr>
<td>M3</td>
<td></td>
<td>NUMBER</td>
</tr>
<tr>
<td>M4</td>
<td></td>
<td>NUMBER</td>
</tr>
<tr>
<td>M5</td>
<td></td>
<td>NUMBER</td>
</tr>
</tbody>
</table>

2. Table Abstract

<table>
<thead>
<tr>
<th>Name</th>
<th>Null?</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>STDNO</td>
<td></td>
<td>NUMBER</td>
</tr>
<tr>
<td>STDNAME</td>
<td></td>
<td>VARCHAR2(10)</td>
</tr>
<tr>
<td>CLASS</td>
<td></td>
<td>VARCHAR2(10)</td>
</tr>
<tr>
<td>MONTH</td>
<td></td>
<td>VARCHAR2(10)</td>
</tr>
<tr>
<td>INTNO (INTEGER NUMBER)</td>
<td></td>
<td>NUMBER</td>
</tr>
<tr>
<td>TOT</td>
<td></td>
<td>NUMBER</td>
</tr>
<tr>
<td>GRADE</td>
<td></td>
<td>VARCHAR2(10)</td>
</tr>
<tr>
<td>PERCENT</td>
<td></td>
<td>NUMBER</td>
</tr>
<tr>
<td>DAT_ENTER</td>
<td></td>
<td>DATE</td>
</tr>
</tbody>
</table>
### Course Objectives:

- This paper is designed to impart knowledge regarding concepts of Financial Accounting. This course is useful for Students to get placements in different offices as well as companies in Accounts departments.

### Course Outcomes:

- Company Setup & Configurations.
- Recording Financial Transactions.
- Financial Reports Analysis.

### Syllabus

#### Unit I

**Fundamentals of Accounting**

Meaning of Accounting, its scope; Objects and limitations; Meaning and application of double entry system, Books of Accounts, Ledgers -Debtors ledger, Creditors ledgers and General ledger; Cash Book and Bank Reconciliation Statement.

#### Unit II

**Financial Statements**


#### Unit III

**Accounting Ratio and Cash Flow Statement**

Ratio Analysis, Objectives of Ratio Analysis, Classification of Accounting Ratios, Advantages of Ratio Analysis, Analysis of Financial Statement through Ratios, Cash Flow Statement,

Unit IV

Cost Concepts and Cost Sheet

Meaning of Cost, Classification of Cost, Various Cost Concepts, Cost Centre, Types of Cost Centres, Cost Unit, Elements of Costs, Cost Sheet

Unit V

Budgetary Control and Marginal Costing

Meaning of Budget, Purpose of Budget, Budgetary Control: Meaning and Essentials, Merits OF budgetary Control system, Steps in preparation of budgets, Classification of budgets, Standard cost and standard costing, Variance analysis, Marginal cost and marginal costing, Advantages of marginal costing, Managerial Application of marginal costing, Break Even Analysis

Capital and Working Capital

Meaning of capital, cost of capital, shares, debentures, capitalisation and capital structure; Meaning of working capital, its components and estimation.

Suggested Readings:

Accounting and Financial Management Lab

<table>
<thead>
<tr>
<th>Semester</th>
<th>Course Code</th>
<th>Course Title</th>
<th>Hours/Week</th>
<th>Hours</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>III</td>
<td>C7-P</td>
<td>Accounting and Financial Management Lab</td>
<td>2</td>
<td>30</td>
<td>1</td>
</tr>
</tbody>
</table>

List of Lab Experiments

1) Create Company using Accounts only.
2) Alter a Company, Shut a Company and Delete a Company in Tally?
3) Splitting Company Data.
4) Creation of Group Companies.
5) Single Ledger Creation with Interest parameters setting.
6) Multi Ledger creation any Ten Accounts.
7) Bank Reconciliation Statement in Tally.
8) Creation of Tally Vault & Change Tally vault Password.
9) Creating Contra voucher.
10) Creating Payment voucher.
11) Creating Receipt voucher.
12) Creating Journal voucher.
13) Creating Sales voucher.
14) Creating Credit Note voucher.
15) Creating Purchases voucher.
16) Creating Debit Note voucher.
17) Displaying Day Book.
18) Displaying Trial Balance.
19) Displaying Profit and Loss Account.
20) Displaying Balance Sheet.
Object Oriented Programming through Java

<table>
<thead>
<tr>
<th>Semester</th>
<th>Course Code</th>
<th>Course Title</th>
<th>Hours/Week</th>
<th>Hours</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>III</td>
<td>C8</td>
<td>Object Oriented Programming through Java</td>
<td>4</td>
<td>60</td>
<td>4</td>
</tr>
</tbody>
</table>

Course Objectives:

To make the students understand fundamentals of Java programming.
- Expose the students to Window based applications using AWT
- Make the students design appropriate Exception Handling in Java
- Make the students understand the concepts of Threads, Files and I/O Streams, Applets, Networking in Java.

Course Outcomes:

The student would become competent enough to write, debug, and document well-structured Java applications
- Demonstrate good object-oriented programming skills in Java
- Able to describe, recognize, apply and implement selected design patterns in Java
- Understand the capabilities and limitations of Java
- Be familiar with common errors in Java and its associated libraries
- Develop excellent debugging skills

Syllabus

Unit I


Introduction to Java Programming: Introduction, Features of Java, Comparing Java and other languages, Applications and Applets, Java Development Kit, More Complex Programs, Java Source file structure, Prerequisites for Compiling and Running Java Programs.

Unit II
Java Language Fundamentals: The building Blocks of Java, Data types, variable declarations, wrapper classes, Operators and Assignment, Control structures, Arrays, Strings, The String Buffer Class.

Java as an OOP Language: Defining classes, Modifiers, Packages, Interfaces.

Unit III

Exception Handling: Introduction, Basics of Exception Handling in Java, Exception Hierarchy, Constructors and Methods in Throwable class, Unchecked and Checked Exceptions, Handling Exceptions in Java, Exception and Inheritance, Throwing User-defined Exceptions, Redirecting and Rethrowing Exceptions, Advantages of Exception – Handling Mechanism.

Multithreading: An Overview of Threads, Creating Threads, Thread Life–cycle, Thread Priorities and Thread Scheduling, Thread Synchronization, Daemon Threads, Thread groups, Communication of Threads.

Unit IV


Database Handling Using JDBC: An Overview of DBMS, JDBC Architecture, Working with JDBC

Unit V


REFERENCE BOOKS:

<table>
<thead>
<tr>
<th>Author</th>
<th>Title</th>
<th>Publisher</th>
</tr>
</thead>
<tbody>
<tr>
<td>Venkateswarlu, EV Prasad, S. Chand</td>
<td>Learn Object Oriented Programming using Java</td>
<td>S CHAND</td>
</tr>
<tr>
<td>Dr. K. Soma Sundaram</td>
<td>Programming in Java 2</td>
<td>JAICO Publishinghouse</td>
</tr>
<tr>
<td>R Nageswararao</td>
<td>JAVA8</td>
<td>CoreJava BlackBook. An Integrated approach</td>
</tr>
</tbody>
</table>
List of Lab Experiments

1. WAP to find the average and sum of the N numbers using Commandline argument.
2. WAP to demonstrate Type Casting.
3. WAP to find the number of arguments provided at runtime.
4. WAP to test the Prime number.
5. WAP to calculate the Simple Interest and Input by the user.
6. WAP to create a simple class to find out the area and perimeter of rectangle and boxes using super and this keyword.
7. WAP to find G.C.D of the number.
8. WAP to design a class account using inheritance and static that show all function of bank (withrawn, deposite).
9. WAP to find the factorial of a given number using Recursion.
10. WAP to design a class using abstract Methods and Classes.
11. WAP to design a String class that perform String Method (Equal, Reverse the string, change case).
12. WAP to handle the Exception using try and multiple catch block.
13. WAP that Implement the Nested try Statements.
14. WAP to create a package that accesses the member of external classes as well as same package.
15. WAP that import the user define package and access the member variable of classes that contained by Package.
16. WAP that show the partial implementation of Interface.
17. WAP to handle the user defined Exception using throw keyword.
18. WAP to create a thread that Implement the Runnable interface.
19. WAP to Implement Interthread communication.
20. WAP to draw the line, Rectangle, oval, text using the graphics method.
21. WAP to create Menu using the frame.
22. WAP to create Dialog box.
23. WAP to Implement the flow layout and Border Layout.
24. WAP to Implement the Grid layout, Card layout.
25. WAP to demonstrate System clock.
26. WAP to create Frame that display the student information.
Course Objectives:

1. To know the basic Structure, Components and Organization of Operating System.
2. To learn the notation of a Process-a Program in Execution, Management, Scheduling and Classic Problems of Synchronization.
3. To gain knowledge in various Memory Management Techniques.
4. To understand Unix Operating System and Various File operations.

Course Outcomes:

The students will be able to:

1. Understand the main components and Structure of Operating System and their functions.
3. Evaluate various device and resources like Memory, Time and CPU Management techniques in distributed systems.
4. Apply different methods for Preventing Deadlocks in a Computer System.
5. Create and build an Application/Service over the UNIX operating system.

Syllabus

Unit I

Unit II

Unit III

Unit IV
**Memory Management:** Background, Basic hardware, Address Binding, Swapping, Contiguous memory allocation, **Paging:** Basic Method, Hardware Support, Protection, Memory Management in UNIX.

**Unit V**
Files and Directories in UNIX, File Structure, File System Implementation of Operating System Functions, File permission, Basic Operation on Files, Changing Permission Modes, Standard files, Processes Inspecting Files, Operating On Files

**TEXT BOOKS**
2. Unix and shell Programming by B.M Harwani, OXFORD University Press

**REFERENCE BOOKS:**
3. Unix Concept and application- Sumitabhadas
OPERATING SYSTEMS Lab

<table>
<thead>
<tr>
<th>Semester</th>
<th>Course Code</th>
<th>Course Title</th>
<th>Hours/Week</th>
<th>Hours</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>III</td>
<td>C9-P</td>
<td>OPERATING SYSTEMS Lab</td>
<td>2</td>
<td>30</td>
<td>1</td>
</tr>
</tbody>
</table>

List of Experiments

1. Introduction to UNIX Operating System, Compare with Windows OS. Writing and executing simple Hello World C Program in UNIX Environment.
2. Working with vi editor: Creating and editing a text file using the standard commands.
3. Getting hands-on on basic UNIX Commands.
4. Execute of various file/directory handling commands.
5. Write a Simple shell script for basic arithmetic and logical calculations.
6. Write Shell scripts to check various attributes of files and directories.
7. Write Shell scripts to perform various operations on given strings.
8. Write Shell scripts to explore system variables such as PATH, HOME etc.
9. Use seed instruction to process /etc/password file.
10. Write a shell script to display list of users currently logged in.
11. Write a shell script to delete all the temporary files.
12. Write a shell script to search an element from an array using binary searching.
13. Write C programs to implement the following Scheduling Algorithms:
    1. First Come First Serve.
    2. Shortest Job First.
    3. Round Robin.

Reference Text Books:

Semester-IV

Cyber Laws

<table>
<thead>
<tr>
<th>Semester</th>
<th>Course Code</th>
<th>Course Title</th>
<th>Hours/Week</th>
<th>Hours</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>IV</td>
<td>C10</td>
<td>Cyber Laws</td>
<td>4</td>
<td>60</td>
<td>4</td>
</tr>
</tbody>
</table>

Course Objectives:

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

Course Outcomes:

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

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

Syllabus

Unit I

Unit II

Unit III
Unit IV

Unit V

Reference Books
Cyber Laws Lab

<table>
<thead>
<tr>
<th>Semester</th>
<th>Course Code</th>
<th>Course Title</th>
<th>Hours/Week</th>
<th>Hours</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>IV</td>
<td>C10-P</td>
<td>Cyber Laws Lab</td>
<td>2</td>
<td>30</td>
<td>1</td>
</tr>
</tbody>
</table>

Lab Experiments

1. Write a program for recovering deleted files from a hard disk.
2. Write a program for gathering evidence.
3. Write a program for viewing files of various formats.
4. Write a program for locating files needed for a forensics investigation.
5. Write a program for performing image and file conversions.
6. Write a program for handling evidence data.
7. Write a program for creating a disk image file of a hard disk partition.
8. Give at least ten cyber crime scenarios to students and make them analyse the scenario and submit report citing cyber laws which are violated.
Course Objectives:

- Be familiar with mathematical foundations of data mining tools.
- Understand and implement classical models and algorithms in data warehouses and data mining.
- Characterize the kinds of patterns that can be discovered by association rule mining, classification and clustering.
- Master data mining techniques in various applications like social, scientific and environmental context. Develop skill in selecting the appropriate data mining algorithm for solving practical problems.

Course Outcomes:

(Five units with each unit having 12 hours of class work)
At the end of the course, the student will demonstrate the following.
The students will be able to:
- Examine the types of the data to be mined and present a general classification of tasks and primitives to integrate a data mining system.
- Apply preprocessing statistical methods for any given raw data
- Discover interesting patterns from large amounts of data to analyze and extract patterns to solve problems, make predictions of outcomes
- Comprehend the roles that data mining plays in various fields and manipulate different data mining techniques
- Select and apply proper data mining algorithms to build analytical applications.
- Evaluate and implement a wide range of emerging and newly-adopted methodologies and technologies to facilitate the knowledge discovery.

Syllabus

Unit I
Central Tendency, Measuring the Dispersion of Data, Data Cleaning, Data Integration and Transformation, Data Reduction.

Unit II

Unit III
Mining Frequent Patterns, Associations, and Correlations: Basic Concepts and a Road Map, Efficient and Scalable Frequent Itemset Mining Methods: The Apriori Algorithm: Finding Frequent Itemsets Using Candidate Generation, Generating Association Rules from Frequent Itemsets, Improving the Efficiency of Apriori, Mining Frequent Itemsets without Candidate Generation. Mining various kinds of Association Rules: Mining Multilevel Association Rules, Mining Multidimensional Association Rules from Relational Databases and Data Warehouses.

Unit IV

Unit V

Prescribed Text Book:
1. Data Mining: Concepts and Techniques Second Edition Jiawei Han University of Illinois at Urbana-Champaign MichelineKamber

References:
1. Data Mining by VikramPudi, P. Radha Krishna, Oxford Universal Press
2. Data Warehousing by ReemaThareja, Oxford University Press
4. Introduction to data mining –G. K. Gupta, PHI
5. Data mining, Data warehouse &Olap-Berson, Tata McGraw Hill

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

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

<table>
<thead>
<tr>
<th>Semester</th>
<th>Course Code</th>
<th>Course Title</th>
<th>Hours/Week</th>
<th>Hours</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>IV</td>
<td>C12</td>
<td>Web Programming</td>
<td>4</td>
<td>60</td>
<td>4</td>
</tr>
</tbody>
</table>

Course Objectives:

- To learn about the Building Blocks of PHP, Arrays.
- To learn about PHP functions and file handling.
- To learn about working with Forms, Sessions, Cookies.
- To learn about Java Script basics.
- To learn about Angular JS basics.

Course Outcomes:

- Able to use Building Blocks of PHP, Access array elements.
- Able to use various functions and handle data using files.
- Able to use working with Forms, Sessions, Cookies.
- Able to implement Java Script.
- Able to implement basic concepts of Angular JS for Web development.

Syllabus

Unit-I: PHP Basics  
Introduction, Identifiers, Variables, Constants, Data Types, Operators, Conditional Statements, PHP Loops.

Working with Arrays: Arrays, Creating Arrays, some Array-Related Functions. Working with Objects: Creating Objects, Object Instance. Working with Strings, Dates and Time: Formatting Strings with PHP, Investigating Strings with PHP, Manipulating Strings with PHP, Using Date and Time Functions in PHP.

Unit-II: PHP Advanced  
Advantages of Using functions, Types of functions, creating and invoking functions, returning values, recursive functions, Object Oriented Concepts, File handling and Data Storage: creating, open/close a file, file operations: read, write, append. File truncate, file uploading, EOF in PHP.

Unit-III: Working with Forms in PHP  
Creating Forms, Accessing Form - Input with User defined Arrays, Combining HTML and PHP code on a single Page, Using Hidden Fields to save state, Redirecting the user, Sending Mail on Form Submission, and Working with File Uploads. Working with Cookies and User Sessions:
Introducing Cookies, Setting a Cookie with PHP, Session Function Overview, Starting a Session, Working with session variables, passing session IDs in the Query String, Destroying Sessions and Unsettling Variables, Using Sessions in an Environment with Registered Users.

Unit-IV: Introduction to Java Script 9 Hrs.

Unit-V: Angular JS 9 Hrs.
Getting started with Angular JS, Modules, Components, Built in Directives, Custom directives. Use of built-in directive, Data Binding, filters, custom filters, constants.

References:
1. FUNDAMENTALS OF OPEN SOURCE SOFTWARE, MN RAO, PHI, 2015.
Web Programming Lab

<table>
<thead>
<tr>
<th>Semester</th>
<th>Course Code</th>
<th>Course Title</th>
<th>Hours/Week</th>
<th>Hours</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>IV</td>
<td>C12-P</td>
<td>Web Programming Lab</td>
<td>2</td>
<td>30</td>
<td>1</td>
</tr>
</tbody>
</table>

Lab experiments

1. Demonstrating PHP operators.
2. Create a PHP program to find odd or even number from given number.
3. Write a PHP program to find maximum of three numbers.
4. Write a PHP program to swap two numbers.
5. Demonstrating while loop in PHP.
6. Demonstrating for loop in PHP.
7. Demonstrating do – while loop in PHP.
8. Demonstrating for each loop in PHP.
9. Write a PHP program for accessing array elements.
10. Write a PHP program to demonstrate various string functions.
11. Write a PHP program to demonstrate Date and Time functions.
12. Write a PHP program to demonstrate invoking functions.
13. Demonstrate recursive function in PHP.
14. Write a PHP program to perform read and write operations on a file.
15. Creating user login form in PHP.
16. Demonstrating Sending Mail on Form Submission in PHP.
17. Demonstrating File Uploads.
18. Demonstrating Working with Cookies.
21. Demonstrating basic commands in java script.
22. Demonstrating conditional structures in Java Script.
23. Demonstrating looping structures in Java Script.
24. Demonstrate accessing array elements in java script.
25. Demonstrating forms and form elements in java script.
26. Demonstrating multiple windows in java script.
27. Demonstrating built-in directives in Angular JS.
28. Demonstrating custom directives in Angular JS.
29. Demonstrating filters in Angular JS.
30. Demonstrating Data binding in Angular JS.
Course Objectives:

- To teach the realistic application of object-oriented analysis and design within a variety of problem domain.

Course Outcomes:

By the end of the course, student will be able to:

- Have Knowledge in micro and macro process.
- Have Knowledge in management planning, quality assurance and metrics along with documentation of object oriented development.
- Have Knowledge in system architecture.
- Basic knowledge in AI and Data Acquisition.
- Knowledge in applications of Object Oriented Design.

Syllabus

UNIT I:

UNIT II:


UNIT III:

**System Architecture** - **Satellite-Based Navigation**: Inception, Elaboration, Construction, Post-Transition. **Control System**: **Traffic Management**: Inception, Elaboration, Construction, Post-Transition.

UNIT IV:

**Artificial Intelligence** - **Cryptanalysis**: Inception, Elaboration, Construction, Post-Transition. **Data Acquisition** - **Weather Monitoring station**: Inception, Elaboration, Construction, Post-Transition.

UNIT V:

**Web Application**: **Vacation Tracking System**: Inception, Elaboration, Construction, Transition and Post-Transition.

**Object-Oriented Programming Languages**: Language Evolution, Smalltalk, C++, Java.

**TEXT BOOK**:

**REFERENCE BOOKS**:


3. Tom Pender, UML Bible, John Wiley and sons.
## Lab Experiments

1. Develop a mini project for Satellite-Based Navigation.
2. Develop a mini project for Traffic Management.
3. Develop a mini project for Cryptanalysis.
4. Develop a mini project for Weather Monitoring Station.
5. Develop a mini project for Vacation Tracking System.
**DATA ANALYTICS USING R**

<table>
<thead>
<tr>
<th>Semester</th>
<th>Course Code</th>
<th>Course Title</th>
<th>Hours/Week</th>
<th>Hours</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>IV</td>
<td>C14</td>
<td>DATA ANALYTICS USING R</td>
<td>4</td>
<td>60</td>
<td>4</td>
</tr>
</tbody>
</table>

**Course Objectives:**

This course will cover all the fundamental algorithms and techniques used in Data Analytics and provide exposure to theory as well as practical knowledge through R used in data analytics. After completing the course, student will learn,

- Fundamental basics of statistics used in analysing the data
- How to find the pattern in the given dataset
- How to interpret the data graphically
- How to apply different types of algorithms for the given dataset

**Course Outcomes:**

- Data-Visualization tools and techniques offer executives and other knowledge workers new approaches to dramatically improve their ability to grasp information hiding in their data.
- Data visualization is a general term that describes any effort to help people understand the significance of data by placing it in a visual context.
- Patterns, trends and correlations that might go undetected in text-based data can be exposed and recognized easier with data visualization software.
- It isn't just the attraction of the huge range of statistical analyses afforded by R that attracts data people to R. The language has also developed a rich ecosystem of charts, plots and visualizations over the years.
- ggplot2 is a data visualization package for the statistical programming language R.
Syllabus

Unit I

Introduction:


Matrices, Arrays and Lists:

Creating matrices –Matrix operations –Applying Functions to Matrix Rows and Columns –Adding and deleting rows and columns –Vector/Matrix Distinction –Avoiding Dimension Reduction –Higher Dimensional arrays –lists –Creating lists –General list operations –Accessing list components and values –applying functions to lists –recursive lists

Data Frames:

Creating Data Frames –Matrix-like operations in frames –Merging Data Frames –Applying functions to Data frames –Factors and Tables –factors and levels –Common functions used with factors –Working with tables -Other factors and table related functions -Control statements –Arithmetic and Boolean operators and values –Default values for arguments -Returning Boolean values

Unit II

Packages in R

Tidy, ggplot2, ggraph, dplyr, tidyquant, dygraphs

Introduction to Data analytics: Overview of Bigdata, Need of Data Analytics, Applications of Data Analytics, Datasets, tools for data analytics

Basic Statistics: Mean, Median, Standard Deviation, Variance, Correlation, Covariance

Unit III

Basic Analysis Techniques:

Chi-Square Test, t-Test, Analysis of Variance, Correlation Analysis

Unit IV

Data Analysis Techniques

Linear Regression, Logistic Regression, Classification Techniques, Clustering Techniques, Ensemble model

Unit V

Data Visualization Using R
Data Visualization, Libraries used for Data Visualization in R, Bar chart, Histogram, Heatmap, Scatter plot, Box Plot, Correlogram, Area Chart

Text Book:

2. R for Data Science: Import, Tidy, Transform, Visualize, and Model Data by Hadley Wickham, O'Reilly
DATA ANALYTICS USING R LAB

<table>
<thead>
<tr>
<th>Semester</th>
<th>Course Code</th>
<th>Course Title</th>
<th>Hours/Week</th>
<th>Hours</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>IV</td>
<td>C14-P</td>
<td>DATA ANALYTICS USING R LAB</td>
<td>2</td>
<td>30</td>
<td>1</td>
</tr>
</tbody>
</table>

List of Experiments

1. R DataFrame: Create, Append, Select, Subset, Merging of dataframes
2. Exporting data to Excel, CSV, SAS, TextFile
3. Handling datasets
4. Mean, Median, Standard Deviation, Variance, Correlation, Covariance in R
5. Correlation in R: Pearson & Spearman with Matrix Example
6. T Test in R
7. Chi-Square Test in R
8. Prediction using linear regression and visualizing the regression graphically
9. Prediction using logistic regression and visualizing the regression graphically
10. Classification algorithms in R
11. Clustering algorithms in R
12. Scatter Plot in R using ggplot2
13. Boxplot in R
14. Barchart & Histogram in R
Object Oriented Software Engineering

<table>
<thead>
<tr>
<th>Semester</th>
<th>Course Code</th>
<th>Course Title</th>
<th>Hours/Week</th>
<th>Hours</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>IV</td>
<td>C15</td>
<td>Object Oriented Software Engineering</td>
<td>4</td>
<td>60</td>
<td>4</td>
</tr>
</tbody>
</table>

Course Objectives:
- Illustrate basic taxonomy and terminology of the software engineering.
- Plan and monitor the control aspects of project.

Course Outcomes:
- Explore the basic concepts of software engineering.
- Choose appropriate life cycle model for a project.
- Implement the phases of the traditional software development process.
- Design various test cases for a software product.
- Analyze different architectural views.

Syllabus

Unit I  The Scope of Object Oriented Software Engineering: Historical Aspects, Economic Aspects, Maintenance Aspects, Requirements, analysis and design aspects, the object oriented Paradigm, Terminology, Ethical Issues.


Unit II  The Software Process: The Unified Process, Iteration and Incrementation, The Requirements Workflow, The Analysis workflow, The Design Workflow, The Implementation workflow, the test workflow, Post Delivery Maintenance, Retirement, the phases of the unified process, one-versus two-dimensional life
cycle models, improving the software process, capability maturity models, costs and benefits of software process improvement.

**Teams**: Team Organization, Democratic Team Approach, Chief Programmer Team Approach, Synchronize and Stabilize Teams, Teams for Agile processes, open source programming teams, people capability maturity model choosing an appropriate team.

**Unit-III**

**Models to Objects**: what is a module? Cohesion, Coupling, Data Encapsulation, Abstract Data Types, Information Hiding, Objects, Inheritance, Polymorphism and Dynamic Binding, The Object-Oriented Paradigm.

**Reusability and Portability**: Objects and Reuse, Reuse during design and implementation reuse and post delivery maintenance, portability, techniques for achieving portability.

**Planning and Estimating**: planning and the software process, Estimating duration and cost, components of a software project management plan, software project management plan framework, planning testing, training requirements, documentations standards.

**Unit-IV**

**The Requirements workflow**: Determining what client needs, overview of the requirements, understanding the domain, the business model, initial requirements, what are object-oriented requirements, rapid prototyping, human factors, reusing the rapid prototype, metrics for the requirement workflow, challenges of the requirements workflow.

**The Analysis Workflow**: The specification document, informal specifications, the analysis workflow, extracting the entity classes, challenges of the analysis workflow.

**The Design Workflow**: Object-Oriented Design, the design workflow, formal techniques for detailed design, real time design techniques, CASE tools for design, metrics for design, challenges of the design workflow.

**Unit-V**

**The implementation workflow**: choice of programming languages, good programming practice, coding standards, code reuse, integration, the implementation workflow.

**Testing**: Quality Issues, Non-Execution based testing, execution based testing, what should be tested?, testing versus correctness proofs, who should perform execution based testing?, when testing stops. Test case selection, Black
Box Unit Testing techniques, Glass-Box Unit Testing Techniques, code walkthrough and inspections, comparison of unit testing techniques, clean room, testing issues integration testing, product testing, and acceptance testing metrics for the implementation workflow challenge of the implementation workflow.

**Post delivery Maintenance:** Why post delivery maintenance is necessary, what is required of post delivery maintenance programmers? Management of post delivery maintenance, maintenance issues, reverse engineering testing during post delivery maintenance metrics for post delivery maintenance, challenges for the post delivery maintenance.

### TEXT BOOKS

<table>
<thead>
<tr>
<th>Author</th>
<th>Title</th>
<th>Publisher</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stephen R. Schach</td>
<td>Object Oriented Software</td>
<td>McGraw Hill Higher Education</td>
</tr>
</tbody>
</table>

### REFERENCE BOOKS

<table>
<thead>
<tr>
<th>Author</th>
<th>Title</th>
<th>Publisher</th>
</tr>
</thead>
<tbody>
<tr>
<td>Timothy C. Lethbridge, Robert Language</td>
<td>Object Oriented Software Engineering</td>
<td>McGraw Hill 2nd Education</td>
</tr>
</tbody>
</table>
## Object Oriented Software Engineering Lab

<table>
<thead>
<tr>
<th>Semester</th>
<th>Course Code</th>
<th>Course Title</th>
<th>Hours/Week</th>
<th>Hours</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>IV</td>
<td>C15-P</td>
<td>Object Oriented Software Engineering Lab</td>
<td>2</td>
<td>30</td>
<td>1</td>
</tr>
</tbody>
</table>

### Practical Syllabus

#### List of Experiments

**Case Studies:**

Design Following Systems in Object Oriented Approach using UML with open source tools (Eclipse UML2 or any other Open source tools):

1. Online Examination System.
2. Online Railway Reservation.
5. Biometric Attendance System.

Note: Student is expected to analyze the system in object oriented manner and design the system in object oriented approach using UML with open source tools.