PONDICHERY UNIVERSITY
(A CENTRAL UNIVERSITY)

B.Sc. Computer Science
(Choice Based Credit System)

Regulations & Syllabus

2017-18 onwards
Pondicherry University
B.Sc.(Computer Science)
REGULATIONS
(Effective from the academic year 2017-2018)

1. **Aim of the Course**

The B.Sc. (Computer Science) course aims to impart the students with fundamental and hands on knowledge of computers, science of computing and modern computer science technologies.

2. **Eligibility of Admission**

Candidates for admission to B.Sc.(CS), shall be required to have passed 10 + 2 system of Examination or equivalent with Mathematics / Business Mathematics / Computer Science/ Computer Applications as one of the subjects of study.

3. **Lateral Entry Admission**

Candidates who have passed Diploma in Computer Science / Information Technology/ Computer Technology / Computer Application in I Class (10+3 years of study) are eligible to apply for the lateral entry to the 2nd year of the course subject to availability of seats, but limited to 10% of the sanctioned intake.

4. **Duration of the course**

The course shall be of three years’ duration spread over six consecutive semesters. The maximum duration to acquire prescribed number of credits in order to complete the Programme of Study shall be twelve consecutive semesters (six years).
5. Medium

The medium of instruction shall be English.

6. Course Structure

<table>
<thead>
<tr>
<th>Category</th>
<th>Course Name</th>
<th>Number Of Papers</th>
<th>Credits Per Paper</th>
<th>Total Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>MIL</td>
<td>Modern Indian Languages</td>
<td>2</td>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>ENG</td>
<td>English</td>
<td>2</td>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>AECC</td>
<td>Ability Enhancement Compulsory Course</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>SEC</td>
<td>Skill Enhancement Course</td>
<td>4</td>
<td>2</td>
<td>8</td>
</tr>
<tr>
<td>GE</td>
<td>Generic Elective Course</td>
<td>2</td>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>DSC</td>
<td>Discipline Specific Core Course</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Theory- 12 Practical – 9 Project -1</td>
<td></td>
<td></td>
<td>12x3=36</td>
</tr>
<tr>
<td></td>
<td>Theory- 3 Practical – 2</td>
<td></td>
<td></td>
<td>9 x 2 =18</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1 x 6 = 6</td>
</tr>
<tr>
<td></td>
<td>Total = 60</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DSE</td>
<td>Discipline Specific Elective Course</td>
<td>6</td>
<td>4</td>
<td>6 x 4 =24</td>
</tr>
<tr>
<td>OE</td>
<td>Open Elective Course</td>
<td>2</td>
<td>3</td>
<td>2 x 3 =6</td>
</tr>
<tr>
<td></td>
<td>Total 120</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**MIL, ENG, AECC**

The crediting of MIL, ENG and AECC courses is as per Pondicherry University UG CBCS regulations.

**DSC and DSE**

At least 60% (72 credits) of the total minimum credit requirement must be earned by the student from DSC and DSE courses as follows in order to obtain the degree - 60 credits from Discipline Specific Core and 12 credits from Discipline Specific Elective courses.
SEC
Out of the 4 Skill Enhancement Courses, one course viz. - Online Course / In-Plant Training (2 weeks) / One month Internship / mini project is mandatory. The Online Course to be studied, the organization to be chosen for In-Plant Training or One month internship is to be validated or approved by a panel of members comprising of the Department Faculty, before a student pursues the same.

For the remaining 3 SEC courses, any of the 2 credit Skill Enhancement Courses specified in the curriculum (B.Sc. CS) could be credited or substituted with Skill Enhancement Courses in the curriculum of other UG computer science courses or Skill Enhancement Courses of other UG Non-Computer Science Disciplines of study that constitute to skill development or an assortment of these without any overlap of courses.

GE
Any 2 of the 3 credit Generic Elective Courses specified in the curriculum (B.Sc. CS) could be credited to constitute the 6 credits or substituted with Generic Elective courses in the curriculum of other UG Computer Science Disciplines of study or UG Courses of Non-Computer Science Disciplines of study that add proficiency to the students - with the advice of the Faculty Advisor, or an assortment of these without any overlap of courses.

DSE
The six 4 credit papers to be credited under DSE can be credited from Discipline Specific Elective specialization stream courses as follows:

I. Three of the 4 credit courses should be credited from one specialization stream courses or across the different specialization stream courses specified in the curriculum.

II. The remaining three of the 4 credit courses may be credited from
   a. Another specialization stream courses of the curriculum or across the different specialization stream courses specified in the curriculum without any overlap of courses credited in I above.
b. Another specialization stream courses or across the different specialization stream courses in the curriculum of other UG Computer Science Disciplines of study without any overlap of courses credited in I above.

or

c. An assortment of the above options in II a and IIb.

**OE**

Any 2 of the 3 credit Open Elective Courses specified in the curriculum (B.Sc. CS) could be credited to constitute the 6 credits or substituted with Open elective courses in the curriculum of other UG Computer Science disciplines of study or substituted with UG Courses of Non-Computer Science Disciplines of study that add proficiency to the students - with the advice of the Faculty Advisor or an assortment of these without any overlap of courses.

**7. Faculty to Students Ratio**

The Faculty to Student Ratio in all the practical / laboratory classes shall be maintained at 1:25.

**8. Pattern of Examination**

I. The End-Semester examination and internal assessments for MIL, ENG, AECC, DSC, GE and OE courses are as per Pondicherry University UG CBCS regulations.

II. All SEC courses (except Online Course / In-Plant Training (2 weeks) / One month Internship) to be treated as a practical / laboratory course and the End-Semester examination to be conducted as per Pondicherry University UG CBCS regulations.

III. The internal assessments for all practical / laboratory courses (for DSC, SEC courses) shall be as follows – 15 marks from two internal practical / laboratory assessment tests and 5 marks based on practical / laboratory course based mini application development.

IV. The internal assessment for DSE courses shall be conducted as follows - 12 marks from two internal assessment tests and 8 marks based only on two internal practical / laboratory assessment tests.
V. The marks for attendance (5 marks) applies to all courses and the awarding of attendance marks is as per Pondicherry University UG CBCS regulations.

VI. The Project work is to be evaluated as follows:

i. The internal assessment (25 marks) is awarded as follows:
   a. 10 marks is awarded based on two internal project reviews conducted in periodic intervals by a panel comprising of members of the Department during the tenure of the project.
   b. The student’s project guide awards 10 marks for the project work and 5 marks for attendance (attendance marks as specified in the Pondicherry University UG CBCS regulations).

ii. The End Semester Examination assessment (75 marks) is evaluated under two aspects viz – i) Project Work – (50 marks) ii) Project Report and Viva-Voce (25 marks)

**Passing Minimum**

Passing Eligibility and classification for the award of the Degree is as per Pondicherry University UG CBCS regulations.

**Lateral Entry**

The Lateral Entry students have to complete 102 credits from the DSC, DSE, GE, SE, OE courses as per curriculum (III$^{rd}$ to VI$^{th}$ semesters). In addition, they should complete the two AECC courses (4 credits) for the award of the degree. One MIL (3 credits) and one ENG (3 credit) courses also need to be completed, if it is not studied in the last three years of the course eligible for lateral entry admission.

Other aspects of CBCS not covered in this document by default conforms to the Pondicherry University UG CBCS regulations.
# PONDICHERRY UNIVERSITY
Bachelor of Computer Science

**PROPOSED STRUCTURE OF THE COURSE UNDER CBCS 2017-2018**

## FIRST SEMESTER

<table>
<thead>
<tr>
<th>COURSE</th>
<th>SUBJECT CODE</th>
<th>Paper</th>
<th>CREDITS</th>
<th>HOURS</th>
</tr>
</thead>
<tbody>
<tr>
<td>MIL</td>
<td>LTAM/LHIN/LTEL/LMAL111</td>
<td>Language-I</td>
<td>3</td>
<td>-</td>
</tr>
<tr>
<td>ENG</td>
<td>ENGL112</td>
<td>English-I</td>
<td>3</td>
<td>-</td>
</tr>
<tr>
<td>DSC-1</td>
<td>CSCS113</td>
<td>Introduction to Problem Solving using C</td>
<td>3</td>
<td>-</td>
</tr>
<tr>
<td>DSC-2</td>
<td>CSCS114</td>
<td>Digital Electronics</td>
<td>3</td>
<td>-</td>
</tr>
<tr>
<td>AECC-1</td>
<td>PADM115</td>
<td>Public Administration</td>
<td>2</td>
<td>-</td>
</tr>
<tr>
<td>DSC-1(lab)</td>
<td>CSCS116</td>
<td>C lab</td>
<td>-</td>
<td>2</td>
</tr>
<tr>
<td>DSC-2 (lab)</td>
<td>CSCS117</td>
<td>Digital lab</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td></td>
<td></td>
<td><strong>18</strong></td>
<td><strong>30</strong></td>
</tr>
</tbody>
</table>

## SECOND SEMESTER

<table>
<thead>
<tr>
<th>COURSE</th>
<th>SUBJECT CODE</th>
<th>Paper</th>
<th>CREDITS</th>
<th>HOURS</th>
</tr>
</thead>
<tbody>
<tr>
<td>MI4L</td>
<td>LTAM/LHIN/LTEL/LMAL121</td>
<td>Language-II</td>
<td>3</td>
<td>-</td>
</tr>
<tr>
<td>ENG</td>
<td>ENGL122</td>
<td>English-II</td>
<td>3</td>
<td>-</td>
</tr>
<tr>
<td>DSC - 3</td>
<td>CSCS123</td>
<td>PYTHON Programming</td>
<td>3</td>
<td>-</td>
</tr>
<tr>
<td>DSC - 4</td>
<td>CSCS124</td>
<td>Data Structures and Algorithms</td>
<td>3</td>
<td>-</td>
</tr>
<tr>
<td>GE – 1 (1 out of 2)</td>
<td>CSCS125</td>
<td>Mathematics for Computer Science</td>
<td>3</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>CSCS126</td>
<td>Numerical Methods</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AECC-2</td>
<td>ENVS127</td>
<td>EVS</td>
<td>2</td>
<td>-</td>
</tr>
<tr>
<td>DSC-3 (lab)</td>
<td>CSCS128</td>
<td>PYTHON lab</td>
<td>-</td>
<td>2</td>
</tr>
<tr>
<td>DSC-4(lab)</td>
<td>CSCS129</td>
<td>Data Structure &amp; Algorithm lab</td>
<td>-</td>
<td>2</td>
</tr>
<tr>
<td><strong>TOTAL</strong></td>
<td></td>
<td></td>
<td><strong>21</strong></td>
<td><strong>30</strong></td>
</tr>
</tbody>
</table>
## THIRD SEMESTER

<table>
<thead>
<tr>
<th>COURSE</th>
<th>SUBJECT CODE</th>
<th>Paper</th>
<th>CREDITS</th>
<th>HOURS</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Theory</td>
<td>Prac.</td>
<td>L</td>
</tr>
<tr>
<td>DSC – 5</td>
<td>CSCS231</td>
<td>Database Management System</td>
<td>3</td>
<td>-</td>
<td>3</td>
</tr>
<tr>
<td>DSC – 6</td>
<td>CSCS232</td>
<td>Visual Programming using C#</td>
<td>3</td>
<td>-</td>
<td>3</td>
</tr>
<tr>
<td>DSC – 7</td>
<td>CSCS233</td>
<td>Computer Networks</td>
<td>3</td>
<td>-</td>
<td>3</td>
</tr>
<tr>
<td>DSC – 8</td>
<td>CSCS234</td>
<td>Software Engineering</td>
<td>3</td>
<td>-</td>
<td>3</td>
</tr>
<tr>
<td>GE-2 (1 out of 2)</td>
<td>CSCS235</td>
<td>Probability and Statistics</td>
<td>3</td>
<td>-</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>CSCS236</td>
<td>Automata Theory &amp; Computations</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DSC-6(lab)</td>
<td>CSCS237</td>
<td>Visual Programming &amp; RDBMS lab</td>
<td>-</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>DSC-7(lab)</td>
<td>CSCS238</td>
<td>Network lab</td>
<td>-</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>1 out of 2</td>
<td>CSCS201</td>
<td>SEC-1 Soft Skills</td>
<td>-</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>CSCS202</td>
<td>SEC-2 Office Automation</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>TOTAL</td>
<td></td>
<td>21</td>
</tr>
</tbody>
</table>

## FOURTH SEMESTER

<table>
<thead>
<tr>
<th>COURSE</th>
<th>SUBJECT CODE</th>
<th>Paper</th>
<th>CREDITS</th>
<th>HOURS</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Theory</td>
<td>Prac.</td>
<td>L</td>
</tr>
<tr>
<td>DSC – 9</td>
<td>CSCS241</td>
<td>Operating Systems</td>
<td>3</td>
<td>-</td>
<td>3</td>
</tr>
<tr>
<td>DSC – 10</td>
<td>CSCS242</td>
<td>Object Oriented Programming using Java</td>
<td>3</td>
<td>-</td>
<td>3</td>
</tr>
<tr>
<td>DSE – 1 DSE – 2 (2 out of 5 stream s)</td>
<td>CSCS243</td>
<td>Client/Server Computing</td>
<td>3</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>CSCS244</td>
<td>Data Warehousing</td>
<td>-</td>
<td>-</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>CSCS245</td>
<td>Object Oriented System Design</td>
<td>-</td>
<td>-</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>CSCS246</td>
<td>Principles of Information Security</td>
<td>3</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>CSCS247</td>
<td>Principles of Programming Languages</td>
<td>-</td>
<td>-</td>
<td>0</td>
</tr>
<tr>
<td>OE-2 (1 out of 2)</td>
<td>CSCS248</td>
<td>Distributed System</td>
<td>3</td>
<td>-</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>CSCS249</td>
<td>Computer Graphics</td>
<td>-</td>
<td>-</td>
<td>0</td>
</tr>
<tr>
<td>DSC-10 (lab)</td>
<td>CSCS250</td>
<td>Object Oriented Programming lab</td>
<td>-</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>1 out of 2</td>
<td>CSCS301</td>
<td>SEC-3 Programming with C++</td>
<td>1</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>CSCS302</td>
<td>SEC-4 Programming with PHP</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>TOTAL</td>
<td></td>
<td>21</td>
</tr>
</tbody>
</table>
### FIFTH SEMESTER

<table>
<thead>
<tr>
<th>COURSE</th>
<th>SUBJECT CODE</th>
<th>Paper</th>
<th>CREDITS</th>
<th>HOURS</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Theory</td>
<td>Prac.</td>
<td>L</td>
<td>T</td>
<td>P</td>
</tr>
<tr>
<td>DSC – 11</td>
<td>CSCS351</td>
<td>Web Technology</td>
<td>3</td>
<td>-</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>DSE – 3</td>
<td>CSCS352</td>
<td>Services Computing</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DSE – 4</td>
<td>CSCS353</td>
<td>Data Mining</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>(2 out of 5)</td>
<td>CSCS354</td>
<td>Software Architecture</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CSCS355</td>
<td>Cryptography and Network Security</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CSCS356</td>
<td>System Software</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OE-2 (1 out of 2)</td>
<td>CSCS357</td>
<td>Artificial Intelligence</td>
<td>3</td>
<td>-</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>CSCS358</td>
<td>Introduction to E-Commerce</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DSC-11 (lab)</td>
<td>CSCS259</td>
<td>Web Technology lab</td>
<td>-</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>1 out of 3</td>
<td>CSCS401</td>
<td>SEC-5 Android Programming</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CSCS402</td>
<td>SEC-6 PROLOG Programming</td>
<td>-</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>CSCS403</td>
<td>SEC-7 software Testing</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Compulsory</td>
<td>CSCS404</td>
<td>SEC-8 Online Course / In-Plant Training (2 weeks) / One month Internship / mini project</td>
<td>-</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>TOTAL</strong></td>
<td><strong>20</strong></td>
<td></td>
<td><strong>30</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Bachelor of Science (COMPUTER SCIENCE)

under CHOICE-BASED CREDIT SYSTEM(CBCS)

(Effective from the academic year 2017- 2018)
Prerequisite: - Basic knowledge of Mathematics and Computers

Objectives:
- To learn the concepts of “C” Programming
- To learn how to use develop software programs for day-to-day applications.

MODULE – I

MODULE- II

MODULE – III
C Functions - Program Modules in C - Math Library Functions – Functions- Function Definitions -Function Prototypes: A Deeper Look - Function Call Stack and Stack Frames- Passing Arguments By Value and By Reference - Recursion vs. Iteration - C Arrays - Defining Arrays - Passing Arrays to Functions- Sorting Arrays- Searching Arrays - Multidimensional Arrays

MODULE – IV
Structure & Union - C Pointers- Pointer Variable Definitions and Initialization- Pointer Operators- Passing Arguments to Functions by Reference - sizeof Operator - Pointer Expressions and Pointer Arithmetic- Relationship between Pointers and Arrays - Pointers to Functions - C Characters and Strings – Character - Handling Library- String-Conversion Functions - Standard Input/Output Library Functions- String-Manipulation Functions -C Formatted Input/Output

MODULE – V
C File Processing - Files and Streams- Creating a Sequential-Access File- Reading Data from a Sequential-Access File - Random-Access Files - Creating a Random-Access File- Writing Data Randomly to a Random-Access File- Reading Data from a Random-Access File- C Preprocessor

Text Books:
LIST OF EXERCISES

1. Simple C programs
2. Program to illustrate control statements
3. Program to illustrate FOR loop
4. Program to illustrate SWITCH & WHILE statements
5. Program to illustrate functions
6. Program to illustrate user-defined functions
7. Program to illustrate arrays
8. Program to illustrate usage of pointers
9. Program to illustrate character handling libraries.
10. Program to illustrate string manipulation
11. Program to illustrate creation of files & streams.
12. Program to illustrate creation, reading & accessing sequential & random files
Prerequisite: Basic knowledge about computers

Objectives:
- To learn the fundamentals of digital system design.
- To learn combinational and sequential logic.
- To learn hardware fundamentals of computer design.

MODULE – I

MODULE – II

MODULE – III

MODULE – IV

MODULE – V
Memory System - Basic concepts, Semi-conductor RAM memories, Read-only memories, Speed, Size and Cost, Cache memories, Performance considerations, Virtual Memories, memory management requirements, Secondary Storage.

Text Books:
LIST OF EXERCISES

1. Study of Logic Gates
2. Design of Adder and Subtractor
3. Design and Implementation of Code Convertors
4. Design of 4-Bit Adder And Subtractor
5. Design and Implementation of Magnitude Comparator
6. 16 Bit Odd/Even Parity Checker and Generator
7. Design and Implementation of Multiplexer and Demultiplexer
8. Design and Implementation of Encoder And Decoder
9. Design and Implementation of 3 Bit Synchronous Up/Down Counter
10. Design and Implementation of Shift Register
11. Simulation of Logic Gates
12. Simulation of Adder and Subtractor
13. Design of 4-Bit Adder and Subtractor
Paper Code: CSCS123

PYTHON PROGRAMMING

Prerequisite: Knowledge of any programming language

Objectives:
- To learn about the fundamentals of computers
- To learn how to install Python, start the Python shell
- To learn to perform basic calculations, print text on the screen and create lists, and perform simple control flow operations using if statements and for loops
- To learn how to reuse code with functions

MODULE – I
Computer Systems - Python Programming Language Computational Thinking - Python Data Types - Expressions, Variables, and Assignments – Strings – Lists – Objects & Classes – Python standard library

MODULE – II
Imperative programming – Python modules – print() function – functional eval() - Execution Control Structures – user-defined functions python variables & assignments parameter passing

MODULE – III
Text Data, Files & Exceptions – Strings revisited – formatted output – files – errors & exceptions - Execution Control Structures – decision control & the IF statement

MODULE – IV
Container and Randomness – Dictionaries – other built-in container types – character encodings & strings – module random

MODULE – V
FOR loop & Iteration Patterns – two-dimensional lists- while loop – more loop patterns – additional iteration control statements- namespaces – encapsulation in functions – global vs local namespaces exceptional flow control – modules as namespaces

Text Books:
LIST OF EXERCISES

1. Program to convert the given temperature from Fahrenheit to Celsius and vice versa depending upon user’s choice.
2. Program to calculate total marks, percentage and grade of a student. Marks obtained in each of the three subjects are to be input by the user. Assign grades according to the following criteria:
   - Grade A: Percentage >=80
   - Grade B: Percentage>=70 and <80
   - Grade C: Percentage>=60 and <70
   - Grade D: Percentage>=40 and <60
   - Grade E: Percentage<40
3. Program, using user-defined functions to find the area of rectangle, square, circle and triangle by accepting suitable input parameters from user.
4. Program to display the first n terms of Fibonacci series.
5. Program to find factorial of the given number.
6. Program to find sum of the following series for n terms: 1 – 2/2! + 3/3! - - - - n/n!
7. Program to calculate the sum and product of two compatible matrices.
8. Program to calculate the mass m in a chemical reaction. The mass m (in gms) disintegrates according to the formula m=60/(t+2), where t is the time in hours. Sketch a graph for t vs. m, where t>0.
9. A population of 1000 bacteria is introduced into a nutrient medium. The population p grows as follows:
   \[ P(t) = \frac{15000(1+t)}{(15+e)} \]
   where the time t is measured in hours. WAP to determine the size of the population at given time t and plot a graph for P vs t for the specified time interval.
10. Input initial velocity and acceleration, and plot the following graphs depicting equations of motion:
    I. velocity wrt time \( v=u+at \)
    II. distance wrt time \( s=u*t+0.5*a*t^2 \)
    III. distance wrt velocity \( s=(v^2-u^2)/2*a \)
Prerequisite: Knowledge of any programming language

Objectives:
- To acquaint students with data structures used when programming for the storage and manipulation of data.
- The concept of data abstraction and the problem of building implementations of abstract data types are emphasized.
- Data Structure Algorithms for stack, queues, linked list, trees, graphs, sorting and searching.

MODULE-I
Definition of a Data structure - primitive and composite Data Types, Arrays, Operations on Arrays, Ordered lists - Stacks - Operations - Applications of Stack - Infix to Postfix Conversion.

MODULE-II

MODULE-III
Trees: Binary Trees - Operations - Graph - Definition, Types of Graphs, Graph Traversal - DFS and BFS.

MODULE-IV

MODULE - V
Role of algorithms in computing - Sorting and Searching Techniques - Elementary sorting techniques –Bubble Sort, Insertion Sort, Merge Sort, Quick Sort

Text Books
LIST OF LAB EXERCISES

1. Implementation of stack
2. Implementation of Queue
3. Implementation of Singly Linked List
4. Implementation of Doubly linked list
5. Implementation of Binary tree and traversals (BFS & DFS)
6. Implementation of Insertion sort
7. Implementation of Selection Sort
8. Implementation of Quick sort
9. Implementation of Merge sort
10. Implementation of Infix to Postfix & Infix to Prefix notations.
DATABASE MANAGEMENT SYSTEM

Prerequisite: Knowledge of data structures and file-handling

Objectives:

- To learn about the basics of database management systems (DBMS), with an emphasis on how to organize, maintain and retrieve efficiently, and effectively the information from a DBMS.
- To learn the fundamental concepts of the relational model, including relations, attributes, domains, keys, foreign keys, entity integrity and referential integrity.
- To learn how to normalize the data using 1st, 2nd & 3rd normal forms
- To define and manipulate the relational databases in SQL.

MODULE - I

MODULE - II
Entity-Relationship Model - Introduction, the building blocks of an entity relationship diagram, classification of entity sets, attribute classification, relationship degree, relationship classification, reducing ER diagram to tables, enhanced entity-relationship model (EER model), generalization and specialization, ISA relationship and attribute inheritance, multiple inheritance, constraints on specialization and generalization, aggregation and composition - advantages of ER modeling.

MODULE - III

MODULE - IV
Structured Query Language - Introduction, History of SQL Standard, Commands in SQL, Data Types in SQL, Data Definition Language, Data Manipulation Language, Data Control Language - Table Modification Commands – primary & foreign keys

MODULE - V
PL/SQL: Introduction, Shortcoming in SQL, Structure of PL/SQL, PL/SQL Language Elements, Data Types, Operators Precedence, Control Structure, steps to Create a PL/SQL, steps to create a Cursors, Procedure, Function, Packages, Exceptions Handling, Database Triggers, Types of Triggers.

Text Books
VISUAL PROGRAMMING USING C#

Prerequisite: Knowledge of C language and DBMS

Objectives:

- To understand the various types of applications
- To get expertise in visual programming
- To understand the functionalities of middleware platform

MODULE – I

MODULE – II
Introduction to Classes and Objects – Introduction - Classes, Objects, Methods, Properties and Instance Variables - Declaring a Class with a Method and Instantiating an Object of a Class - Declaring a Method with a Parameter - UML Class Diagram with a Property - Software Engineering with Properties and set and get Accessors - Initializing Objects with Constructors - Floating-Point Numbers and Type decimal - Control Statements

MODULE – III
Classes and Objects: A Deeper Look – Introduction - Controlling Access to Members - Referring to the Current Object’s Members with the this Reference – Indexers - Default and Parameterless Constructors – Composition - Garbage Collection and Destructors- static Class Members - Data Abstraction and Encapsulation - Object Initializers – Delegates Object-Oriented Programming: Inheritance - Polymorphism, Interfaces and Operator Overloading-Exception Handling

MODULE – IV

MODULE – V
Databases and LINQ - Introduction - relational Databases - LINQ to SQL - Querying a Database with LINQ - Dynamically Binding Query Results - Retrieving Data from Multiple Tables with LINQ - Creating a Master/Detail View Application - Tools and Web Resources Case Study

Text Book:
LIST OF EXERCISES

DBMS

For any TWO online application such as library information system, students; information system, employee information systems, payroll system, ticket reservation system etc., do the followings:

1. Create database and establish relationships between tables
2. Draw ER diagrams
3. Create view to extract details from two or more tables
4. Create stored procedures
5. Create functions
6. Create cursors & database triggers.
7. Create PL/SQLs.

C#

1. Implement Classes and Objects, Inheritance & Polymorphism
2. Implement Interfaces, Operator Overloading, Delegates and Events
3. Implement Exception Handling & Multi-Threading
4. Create Console application & Window Applications.
5. Create programs using SDI & MDI
6. Create program using Database Controls
7. Develop any TWO case studies listed below:
   I. Inventory Control
   II. Retail Shop Management
   III. Employee Information System
   IV. Personal Assistant Program
   V. Students’ Information System
COMPUTER NETWORKS

Prerequisite: Basic knowledge of computers

Objectives:
- Given an environment, after analyzing the channel characteristics, appropriate channel access mechanism and data link protocols are chosen to design a network.
- Given an environment, analyzing the network structure and limitations, appropriate routing protocol is chosen to obtain better throughput.
- Given various load characteristics and network traffic conditions, decide the transport protocols and timers to be used.

MODULE - I
Introduction to Networks – Topology - Network Architecture - Reference Models - Example Networks – Transmission Medias

MODULE - II

MODULE - III

MODULE - IV
Application layer - Domain Naming System - DNS Namespace, Resource Records, Name Servers - Electronic mail - Architecture and Services, The User Agent, Messages Formats, Message Transfer

MODULE - V

Text Books:
LIST OF EXERCISES

1. Implementation of Error Detection / Error Correction Techniques
2. Implementation of Stop and Wait Protocol and sliding window
3. Implementation and study of Go back-N and selective repeat protocols
4. Implementation of High Level Data Link Control
5. Study of Socket Programming and Client – Server model
6. Write a socket Program for Echo/Ping/Talk commands.
7. To create scenario and study the performance of network with CSMA / CA Protocol and compare with CSMA/CD protocols.
8. Network Topology - Star, Bus, Ring
9. Implementation of distance vector routing algorithm
10. Implementation of Link state routing algorithm
11. Encryption and decryption.
SOFTWARE ENGINEERING

Prerequisite: Basic knowledge of programming

Objectives:
- Identify, formulate, and solve software engineering problems, including the specification, design, implementation, and testing of software systems that meet specification, performance, maintenance and quality requirements
- Elicit, analyze and specify software requirements through a productive working relationship with various stakeholders of a software project.
- Need to function effectively as a team member
- Understanding professional, ethical and social responsibility of a software engineer
- Participate in design, development, deployment and maintenance of a medium scale software development project.

MODULE – I

MODULE – II

MODULE – III
Software Project Planning – size estimation – cost estimation – models – Constructive cost model – software risk management – software design – what is design – modularity – strategy of design – function oriented design - object oriented design

MODULE – IV

MODULE – V
Software testing – strategic approach to software testing – terminologies – functional testing – structural testing – levels of testing – validation testing – the art of debugging – testing tools

Text Book:
OPERATING SYSTEMS

Prerequisite: Knowledge of computers & computer organization

Objectives:
- To learn Structure and functions of OS
- To learn Processes and Threads, Scheduling algorithms
- To learn Principles of concurrency and Memory management
- To learn I/O management and File systems

MODULE - I

MODULE - II

MODULE - III

MODULE - IV

MODULE - V

Text Books:
Prerequisite: Basic knowledge of programming

Objectives:
- On successful completion of the course the students should have understood the object oriented programming in java
- Should have idea about GUI bases programming
- Should have idea about database programming

MODULE – I
Introduction – Introduction to java applications – Introduction to classes, objects, methods & Strings - Control statements - Arrays

MODULE – II
Class & Objects – constructor – function overloading & overriding - Inheritance - Polymorphism – Interface – package - exception handling - Introduction to Multithreading

MODULE – III

MODULE – IV
Files, Streams & Object Serialization – Introduction – Files & Streams – Sequential Access Text Files – Object Sterilization

MODULE – V

Text Books:
LIST OF EXERCISES

1. Program to illustrate various date types in Java.
2. Program to illustrate class and objects.
3. Program to illustrate control structures (if-then, while, switch).
4. Program to illustrate the concept of arrays (creation, initialization and processing).
5. Program to illustrate Multidimensional arrays.
6. Program to illustrate Constructor and its overloading.
7. Program to illustrate Inheritance and Packages.
8. Program to illustrate Interface and static methods.
9. Program to illustrate modifiers protected, this, final and super.
10. Program to illustrate Exception Handling Technique.
11. Program to illustrate to input/output streams.
12. Program to illustrate File handling technique.
13. Program to illustrate threading.
14. Program to illustrate simple Java applets.
15. Program to illustrate database programming.
WEB TECHNOLOGY

Prerequisite: Knowledge of operating system, computer network, DBMS, and java language.

Objectives:
- To inculcate knowledge of web technological concepts and functioning of internet
- To learn and program features of web programming languages.
- To understand the major components of internet and associated protocols.
- To design an innovative application for web.

MODULE – I

MODULE - II

MODULE – III

MODULE – IV
Server-Side Programming: Java Servlets - Model-View-Controller Paradigm - Servlet Architecture Overview - Servlets Generating Dynamic Content - Servlet Life Cycle - Parameter Data

MODULE – V
Sessions - Cookies - URL Rewriting - Servlets and Concurrency – database programming using Servlet.

Text Book:
LIST OF EXPERIMENTS

1. Creation of HTML Files
2. Working with Client Side Scripting
   2.1 JavaScript
3. Configuration of web servers
   3.1 Apache Web Server
   3.2 Internet Information Server (IIS)
4. Experiments in Servlet
   5.1 Implementing MVC Architecture using Servlets
   5.2 Data Access Programming (using ADO)
   5.3 Session and Application objects
   5.4 File System Management
5. Write programs in Java to create three-tier applications using servlets
   • for conducting on-line examination.
   • for displaying student mark list. Assume that student information is available in a database which has been stored in a database server.
MICROPROCESSORS AND MICROCONTROLLERS

Prerequisite: Knowledge of computer organization

Objectives:
- To understand the architectures and the instruction set of 8085 microprocessor
- To understand the architectures and the instruction set of 8086 microprocessor
- To understand the architectures and the instruction set of 8051 microcontroller
- To learn the assembly language program using 8085, 8086 and 8051 instructions
- To learn interfacing of microprocessors and microcontrollers with various devices

MODULE – I

MODULE – II
Intel 8085 Interrupts and DMA: 8085 Interrupts – Software and Hardware Interrupts – 8259 Programmable Interrupt Controller - Data Transfer Techniques – Synchronous, Asynchronous and Direct Memory Access (DMA) and 8237 DMA Controller- 8253 Programmable Interval Timer.

MODULE – III

MODULE – IV

MODULE – V

Text Books:
LIST OF EXERCISES

1. Basic Arithmetic and Logical Operations 16 Bit Addition
2. Basic Arithmetic and Logical Operations 16 Bit Subtraction
3. Basic Arithmetic and Logical Operations 16 Bit Multiplication
4. Basic Arithmetic and Logical Operations 16 Bit Division
5. Move a Data Block Without Overlap
7. Code Conversions – Decimal to Hexadecimal
8. Code Conversion – Hexadecimal to Decimal
9. Floating Point Operations – String Manipulations, Sorting and Searching,
   Copying a String
10. Ascending & Descending
PROJECT

The objective of the project is to motivate them to work in emerging/latest technologies, help the students to develop ability, to apply theoretical and practical tools/techniques to solve real life problems related to industry, academic institutions and research laboratories.

The project is of 2 hours/week for one (semester VI) semester duration and a student is expected to do planning, analyzing, designing, coding, and implementing the project. The initiation of project should be with the project proposal. The synopsis approval will be given by the project guides.

The project proposal should include the following:

- Title
- Objectives
- Input and output
- Details of modules and process logic
- Limitations of the project
- Tools/platforms, Languages to be used
- Scope of future application

The project work should be either an individual one or a group of not more than three members and submit a project report at the end of the semester. The students shall defend their dissertation in front of experts during viva-voce examinations.
DISCIPLINE SPECIFIC ELECTIVE (DSE) PAPERS

Specialization Stream – I [ADVANCED COMPUTING STREAM]

Paper Code: CSCS243

CLIENT/SERVER COMPUTING

Prerequisite: Knowledge of computer networks & DBMS

Objectives:
- To learn about objective evaluations and details of Client/Server development tools, used operating system and database management system and its mechanism in respect to client/server computing and network components used in order to build effective client/server applications.

MODULE – I
Introduction – defining client/server computing – Classification of client/server systems – clients/server – advantages & disadvantages – driving forces behind client/server computing

MODULE – II

MODULE – III

MODULE – IV

MODULE - V
System development – hardware & software requirements – communication interface technology – client/server technology & web services – what are web services – web services & client/server/browser – server technology – client/server technology & web applications

Text Book:
SERVICES COMPUTING

Prerequisite: Knowledge of computer networks, client/server computing & distributed system

Objectives:
- To understand the advantages of using XML technology family
- To analyze the problems associated with tightly coupled distributed software architecture
- To use Web services as building block in distributed application development
- To design e-business solutions using SOA and XML based web services

MODULE – I
Web services basics – Introduction - concept of software as a service - more complete definition of Web services - Characteristics of Web services - Service interface and implementation - service-oriented architecture - Operations in the SOA - Web services technology stack - Quality of service(QoS) - Web services interoperability

MODULE – II
Enabling infrastructure - Distributed computing infrastructure- Distributed computing and Internet protocols - client–server model - Characteristics of inter-process communication - Synchronous forms of middleware - Asynchronous forms of middleware - Request/reply messaging - Message-oriented middleware

MODULE – III
Brief overview of XML - XML document structure - URIs and XML namespaces - XML schemas reuse - Document navigation and transformation –

MODULE – IV
Core functionality and standards - SOAP: Simple Object Access Protocol - Inter-application communication and wire protocols - SOAP as a messaging protocol - Structure of a SOAP message - SOAP communication model - Error handling in SOAP - SOAP over HTTP - Advantages and disadvantages of SOAP

MODULE – V
Describing Web services - service description needed - WSDL: Web Services Description Language - Using WSDL to generate client stubs - Non-functional descriptions in WSDL - Registering and discovering Web services - Registering and discovering Web services Service registries - Service discovery - UDDI: Universal Description, Discovery, and Integration

Text Book:
Paper Code: CSCS363

CLOUD COMPUTING

Prerequisite: Knowledge of operating system, distributed system & services computing.

Objectives:
- To impart the principles and paradigm of Cloud Computing
- To understand the Service Model with reference to Cloud Computing
- To comprehend the Cloud Computing architecture and implementation
- To realize the role of Virtualization Technologies

MODULE – I

MODULE – II
Cloud Deployment Models – Introduction - Private Cloud - Public Cloud- Hybrid Cloud- Cloud Service Models- Infrastructure as a Service- Platform as a Service

MODULE – III

MODULE – IV
Virtualization - Approaches to Virtualization- Hypervisors - From Virtualization to Cloud Computing- Programming Models for Cloud Computing

MODULE – V
Software Development in Cloud Introduction - Different Perspectives on SaaS Development - New Challenges - Cloud-Aware Software Development Using PaaS Technology

Text Book:
DISCIPLINE SPECIFIC ELECTIVE (DSE) PAPERS

Specialization Stream – II [BUSINESS INTELLIGENCE STREAM]

Paper Code: CSCS244

DATA WAREHOUSING

Prerequisite: Knowledge of database management system

Objectives:

- To learn the fundamentals of designing large-scale data warehouses using relational technology.
- To study the design aspects, planning and development.

MODULE - I

MODULE - II

MODULE - III
Hardware and operational design – server hardware, network hardware – parallel technology – Security input on design of Hardware – backup and recovery – Service level Agreement – Operating the data warehouse.

MODULE IV

MODULE - V

Text Books:
1. Sam Anahory & Dennis Murray, “Data Warehousing in the real world”, Pearson Education.
DATA MINING

**Prerequisite:** Knowledge of database management system

**Objectives:**
- To understand the concepts of Data Mining.
- To learn about Classification, prediction and cluster analysis techniques.
- To learn about applications of Data and knowledge mining.

**MODULE - I**
An Introduction to Data Mining - Introduction - The Data Mining Process - The Basic Data Types - The Major Building Blocks - Association Pattern Mining - Data Clustering - Outlier Detection - Data Classification - Impact of Complex Data Types on Problem Definitions - Scalability Issues and the Streaming Scenario - Some Application Scenarios

**MODULE – II**
Data Preparation – Introduction - Feature Extraction and Portability - Data Cleaning - Data Reduction and Transformation

**MODULE – III**
Similarity and Distances - Introduction - Multidimensional Data - Text Similarity Measures - Temporal Similarity Measures - Graph Similarity Measures - Supervised Similarity Functions

**MODULE – IV**

**MODULE – V**
Cluster Analysis – Introduction - Feature Selection for Clustering - Representative-Based Algorithms - Hierarchical Clustering Algorithms - Cluster Validation Clustering Categorical Data - Outlier Analysis – Introduction - Extreme Value Analysis - Clustering for Outlier Detection - Distance-Based Outlier Detection

**Text Book:**
FOUNDATIONS OF DATA ANALYTICS

Prerequisite: Knowledge of DBMS and data mining

Objectives:

- To learn relevant parts of statistics, computer science, and machine learning that are crucial to data science.
- To learn science from a pragmatic, practice-oriented viewpoint.
- To learn about useful statistical and machine learning concepts, include concrete code examples, and explore partnering with and presenting to non-specialists.

MODULE - I
Introduction to Data Science - Data science process – roles, stages in data science project – working with data from files – working with relational databases – exploring data – managing data – cleaning and sampling for modeling and validation – introduction to NoSQL.

MODULE - II

MODULE - III

MODULE - IV
Map Reduce: Introduction – distributed file system – algorithms using map reduce, Matrix-Vector Multiplication by Map Reduce – Hadoop - Understanding the Map Reduce architecture - Writing Hadoop Map Reduce Programs - Loading data into HDFS – Executing the Map phase

MODULE - V
Delivering Results - Documentation and deployment – producing effective presentations– Introduction to graphical analysis – plot () function – displaying multivariate data – matrix plots – multiple plots in one window - exporting graph - using graphics parameters - Case studies.

Text Books:
DISCIPLINE SPECIFIC ELECTIVE (DSE) PAPERS

Specialization Stream – III [SOFTWARE ENGINEERING STREAM]

Paper Code: CSCS245

OBJECT ORIENTED SYSTEM DESIGN

Prerequisite: Knowledge of object oriented programming

Objectives:
- Understand software modeling and Architectural Concepts
- Understand and apply UML notations in designing software
- Gain knowledge about Static and Dynamic modeling

MODULE – I
Introduction - overview - Object basics - Object state and properties, Behavior, Methods, Messages- Object Oriented system development life cycle - Benefits of OO Methodology. - Overview of Prominent OO Methodologies - Rumbaugh OMT-. The Booch methodology- Jacobson's OOSE methodologies- Unified Process

MODULE – II

MODULE – III
Class Modeling and Design Approaches - Three approaches for identifying classes - using Noun phrases, Abstraction, Use Case Diagram - Comparison of approaches- Using combination of approaches. - Flexibility guidelines for class diagram: Cohesion, Coupling, Forms of coupling - class Generalization, class specialization versus aggregation -Behavioral - State diagram - State Diagram states - Interaction diagrams- Sequence diagram - Sequence diagram notations - Activations in sequence diagram- Collaboration diagram - Collaboration diagram notations

MODULE – IV
Approaches for developing dynamic systems- Top-down approach for dynamic systems- Bottom-up approach for dynamic systems- Flexibility Guidelines for Behavioral Design - Architectural view- Logical architecture - Hardware architecture - deployment diagram notations, nodes, object migration between node - Process architecture - process and threads notations in UML, object synchronization, invocation schemes for threads - Implementation architecture - component diagram notations and examples.

MODULE – V
Reuse - Libraries, Frame works components and Patterns- Reuse of classes- Reuse of components- Reuse of frameworks, black box framework, white box frame- Reuse of patterns - Architectural pattern and Design pattern.

Text books:
SOFTWARE ARCHITECTURE

Prerequisite: Knowledge of software engineering

Objectives:
- To learn about Architectural styles and Quality Attributes.
- To learn about common tools and terminology related to software architecture.
- To learn the role of the Software Architect with a development project.
- To learn how to use methods for constructing and evaluating architectures.
- To learn Advance Concepts in Architecture.

MODULE - I

MODULE - II

MODULE - III

MODULE - IV

MODULE - V

Text Books:
SOFTWARE QUALITY MANAGEMENT

Prerequisite: Knowledge of software engineering

Objectives

- To learn quality assurance plans
- To learn how to apply quality assurance tools & techniques
- To learn about standards and certifications
- To learn how to describe procedures and work instructions in software organizations

MODULE - I

MODULE - II

MODULE - III
Software Quality Infrastructure Components - Procedures and Work Instructions – Supporting Quality Devices - Staff Training- Instructing and Certification - Preventive and Corrective Actions – Configuration Management - Documentation and Quality Records Controls.

MODULE - IV
Software Quality Management Components - Project Progress Control - components of project progress control- Progress control of internal projects and external participants- Implementation of project progress control

MODULE - V

Text Books:
DISCIPLINE SPECIFIC ELECTIVE (DSE) PAPERS

Specialization Stream – IV [INFORMATION SECURITY STREAM]

Paper Code: 246

PRINCIPLES OF INFORMATION SECURITY

Prerequisite: Basic knowledge of computers

Objectives:

- To provide an understanding of principal concepts, major issues, technologies and basic approaches in information security.
- Develop an understanding of information assurance as practiced in computer operating systems, distributed systems, networks and representative applications.
- Gain familiarity with prevalent network and distributed system attacks, defenses against them and forensics to investigate the aftermath.

MODULE – I

MODULE – II

MODULE – III

MODULE – IV

MODULE – V

Text Book:
CRYPTOGRAPHY AND NETWORK SECURITY

Prerequisite: Knowledge of mathematics, information security & computer networks.

Objectives:
- To learn about network security
- To learn Computer Network Vulnerabilities
- To learn how to deal with Network Security Challenges
- Develop a basic understanding of cryptography, how it has evolved and some key encryption techniques used today.
- Develop an understanding of security policies (such as authentication, integrity and confidentiality)
- To learn about network security threats and countermeasures

MODULE – I

MODULE – II

MODULE – III
Computer Network Vulnerabilities - Sources of Vulnerabilities- Vulnerability Assessment - Cyber Crimes and Hackers - Cyber Crimes – Hacker - Dealing with the Rising Tide of Cyber Crimes

MODULE – IV

MODULE – V

Text Books:
ETHICAL HACKING

Prerequisite: Knowledge of cryptography & information security

Objectives:
- To understand how intruders, escalate privileges in a system.
- To understand Intrusion Detection, Policy Creation, Social Engineering, DDoS Attacks, Buffer Overflows and Types of Attacks and Protections.
- To learn Classification and Mechanism of Ethical Hacking.
- To learn the basic principles, instrumentation and applications of Ethical Hacking.

MODULE I

MODULE II

MODULE III

MODULE IV
Routers, Firewall & Honeypots, IDS & IPS, Web Filtering, Vulnerability, Penetration Testing, Session Hijacking, Web Server, SQL Injection, Buffer Overflow, Reverse Engineering, Email Hacking, Incident Handling & Response, Bluetooth Hacking, Mobile Phone Hacking

MODULE V
Social Engineering, Host Reconnaissance, Session Hijacking, Hacking - Web Server, Database, Password Cracking, Network and Wireless, Trojan, Backdoor, UNIX, LINUX, Microsoft, Buffer Overflow, Denial of Service Attack.

Text Books:
DISCIPLINE SPECIFIC ELECTIVE (DSE) PAPERS

Specialization Stream – V [SCIENCE OF PROGRAMMING LANGUAGE STREAM]

Paper Code: 247

<table>
<thead>
<tr>
<th>L</th>
<th>T</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

PRINCIPLES OF PROGRAMMING LANGUAGES

Prerequisite: Knowledge of programming

Objectives:
- To understand the constructs of programming language
- To know the different programming paradigms
- To associate the specific paradigm and language to solve a problem
- To learn new programming techniques

MODULE - I
The challenge of programming language design - Criteria for language design-some possible solutions - Defining syntax: General problem of describing syntax, formal methods of describing syntax, BNF, Syntax Graphs - Syntax and program Reliability.

MODULE - II
Variables, Expressions and statements - Variables and assignment statement, Binding Time and Storage Allocation, Constants and initialization, Expressions, Statements-Condition, Iteration - GOTO and Labels – Types - Data types and Typing. Enumerated and elementary, pointer, structured Data types, Type coercion & Equivalence - Scope and Extent

MODULE - III
Procedures - General features, Parameter evaluation & passing, Call-By-Name, Specification of objects in a procedure, aliasing, Overloading, Generic functions, Co-routines - Abstract data types - concept of abstraction, Encapsulation, Introduction to data abstraction, design issues, parameterized abstract data types.

MODULE - IV
Exception Handling - Introduction, Exception Handling in PL/I, Exception Handling in Ada, Exception Handling in C++ - Concurrency - Basic concepts, subprogram-level concurrency, statement-level concurrency, semaphore, Monitors, Message passing.

MODULE - V

Text Books
Paper Code: 356

SYSTEM SOFTWARE

Prerequisite: Knowledge of computer organization and programming

Objectives:
  • To understand the working of assemblers, loaders, linkers, macroprocessors and compilers.

Module – I
Introduction: System software and machine architecture, traditional (CISC) machines, RISC machines.

Module - II
Assemblers: Basic assembler functions, machine dependent and machine independent assembler features, one-pass assemblers, multi pass assemblers, MASM assembler, SPARC assembler.

Module - III
Loaders and Linkers: Basic loader functions, machine dependent and machine independent loader features, linkage editors, dynamic linking, bootstrap loaders.

Module - IV
Macro Processors: Basic macro processor functions, machine dependent and machine independent macro processor features, macro processor design options.

Module - V
Compilers: Basic compiler functions, machine-dependent compiler features, machine-independent compiler features, compiler design options the YACC compiler-compiler.

Text Books:
PRINCIPLES OF COMPILER DESIGN

Prerequisite: Knowledge of system software and principles of programming

Objectives:
- To gain basic features of system software (assemblers / loaders / linkers / compilers)
- To gain knowledge on data structures required for implementation of system software like assemblers/loaders/compliers
- To understand the design of assemblers.
- To understand the role of loaders and linkers in Loading, relocation and linking.
- To understand the various phases of designing a compiler
- To use grammars for parsing.
- To understand the various types of code optimization and code generation techniques.

MODULE - I

MODULE - II

MODULE - III

MODULE - IV

MODULE - V
Code generation – issues in the design of a code generator – runtime storage management – basic blocks and flow graph – register allocation and assingment – DAG representation of basic blocks, generating code from DAGs- introduction to code generation – introduction – principle sources of optimization – peephole optimization – potimization of basic blocks

Text Book:
OPEN ELECTIVE – I

Paper Code: 248

DISTRIBUTED SYSTEM

Prerequisite: Knowledge of operating systems, DBMS and Computer Networks
Objective:
- To make the students to understand the collaborative operations of collections of computer systems.

MODULE I

MODULE II

MODULE III
Remote Invocation – Introduction - Request-reply protocols - Remote procedure call - Remote method invocation - Group communication - Publish-subscribe systems - Message queues - Shared memory approaches -Distributed objects - Case study: CORBA -from objects to components

MODULE IV
Peer-to-peer Systems – Introduction - Napster and its legacy - Peer-to-peer – Middleware - Routing overlays - Overlay case studies: Pastry, Tapestry

MODULE V
- Distributed File Systems –Introduction - File service architecture - Distributed mutual exclusion – Elections

Text Book:
OPEN ELECTIVE – II

Paper Code: 249

COMPUTER GRAPHICS

Prerequisite: Knowledge of computers and programming

Objectives:
- Gain knowledge about graphics hardware devices and software used.
- Understand the two dimensional graphics and their transformations.
- Understand the three dimensional graphics and their transformations.
- Be familiar with understand clipping technique

MODULE - I

MODULE - II

MODULE - III

MODULE - IV

MODULE - V

Text Book:
OPEN ELECTIVE – III

Paper Code: CSCS357

ARTIFICIAL INTELLIGENCE

Prerequisite: Knowledge of predicate calculus and programming

Objectives:
• To study the concepts of Artificial Intelligence and Methods of solving problems using Artificial Intelligence
• To understand the basic techniques of knowledge representation and their use and components of an intelligent agent
• To be able to implement basic decision making algorithms, including search based and problem solving techniques, and first-order logic.
• To know the basic issues in machine learning

MODULE - I
Introduction to AI & Production Systems - Introduction - AI problems, foundation of AI and history of AI intelligent agents - Agents and Environments - the concept of rationality, the nature of environments, structure of agents, problem solving agents, problem formulation.

MODULE - II

MODULE - III
Representation of Knowledge - Knowledge Representation & Reasons logical Agents, Knowledge – based Agents, the Wumpus world, logic, propositional logic, Resolution patterns in propositional logic, Resolution, Forward & Backward Chaining

MODULE - IV
First order logic - Inference in first order logic, propositional vs. first order inference, unification & lifts forward chaining, Backward chaining, Resolution - Learning - Learning from observations – forms of learning

MODULE - V
An Overview of Prolog - An example program: defining family relations - Extending the example program by rules -A recursive rule definition - How Prolog answers questions - Declarative and procedural meaning of programs - Syntax and Meaning of Prolog Programs - Lists, Operators, Arithmetic - Using Structures: Example Programs

Text Books:
OPEN ELECTIVE – IV

CSCS358

INTRODUCTION TO E-COMMERCE

Prerequisite: Knowledge of computer networks

Objectives:
- To learn both the technical and business-related implications of electronically mediated commerce.
- To learn the development of electronic business from its origins in electronic data interchange to its current growing importance.
- To learn the potential of electronic business for future development and the development of the 'Information Society' and ethical issues facing business organizations in their daily use of the Internet

MODULE – I
Introduction to e-commerce – benefits of e-commerce – impact of e-commerce – classification of e-commerce – Web 2.0 based social networking platform for social media e-commerce – application of e-commerce technologies

MODULE – II

MODULE – III

MODULE - IV

MODULE – V

Text Book:
SKILL ENHANCEMENT COURSES (SEC)

Paper Code: CSCS201

SOFT SKILLS

Prerequisite: Basic knowledge of English language

Objectives:
- To enable learners to develop their communicative competence.
- To facilitate learners to improve their soft skills.
- To equip learners with employability skills to enhance their prospect of placements.

MODULE - I
Nature of technical communication: Stages of communication – Channels of communication – Nature of technical communication – Importance and need for technical communication – Technical communication skills - The Listening process: Types of listening – Listening with a purpose – Barriers to listening – The speech process – Conversion and oral skills – Body language.

MODULE - II

MODULE - III
Presentation Skills: Planning the presentation – Preparing the presentation – Organizing your presentation – Rehearsing the presentation – Improving delivery

Text Book:

SOFT SKILLS LAB – EXERCISES

1. ORAL PRESENTATION
- TV violence.
- Is the Fast-Food Industry Accountable Legally for poor health?
- Intelligence depends more on the environment than genetic factors.
- Environment vs. technology Impact of technology on learning
- Learning does not eradicate ignorance
- How WiFi improved your life?

2. GROUP DISCUSSION
- NGOs - Do they serve peoples’ interests or are they pressure groups?
- Role of women in development.
- Kids today are not what they used to be.
- Repeated elections - Should taxpayers pay for it?
- In India, the whole is less than the parts - Do we lack in team spirit?
- "Dot.com" companies - Is there room for everyone?
- Artificial Intelligence - Will man be ever replaced by machines?

3. INTERVIEW SKILLS
- How to make a good impression
- Basic Interview Questions
- Behavioural Interview Questions
INTRODUCTION TO OFFICE AUTOMATION

**Prerequisite:** Knowledge of computers

**Objectives:**
- To understand how to use office automation software packages in day to day activities

**MODULE – I**

**MODULE – II**
MS Excel - Introduction and area of use -Working with MS Excel - concepts of Workbook & Worksheets - Working with Data & Ranges - Different Views of Worksheets - Column Freezing, Labels, Hiding, Splitting etc.;-Using different features with Data and Text - Use of Formulas, Calculations & Functions-Cell Formatting including Borders & Shading; Working with Different Chart Types - Printing of Workbook & Worksheets with various options.

**MODULE – III**
MS Access: DBMS Concept; Creating database, table, fields & its properties; Data types; Adding primary key into table; Relationship; Adding/Editing data; sorting; indexing; designing queries; using forms; Report generation.

**Text Books:**

**LIST OF LAB EXERCISES:**
1. To create a personal letter using MS-WORD
2. To create company letter head using MS-WORD
3. To create a memo using MS-WORD
4. To create a greeting card using MS-WORD
5. To create a cover page of a project report.
6. To create letter using mail merge.
7. To create a slide show regarding our college and department.
8. To create a spreadsheet for mark statement of students.
9. To create various graphs with respect to students’ academic details.
INTRODUCTION TO C++

Prerequisite: Knowledge of C programming

Objectives:
- To learn the basics of C++ declarations, operators and expressions.
- To work on all the elementary statements and arrays, manipulation of strings, functions and pointers.
- Perform object oriented programming to develop solutions to problems demonstrating usage of control structures and other standard language constructs.
- Demonstrate aptness of object oriented programming in developing solutions to problems demonstrating usage of data abstraction, encapsulation, and inheritance
- Learn syntax, features of the Standard Template Library and exception handling technique.

MODULE - I
Concepts of OOP: Introduction OOP, Procedural Vs Object Oriented Programming, Principles of OOP, Benefits and applications of OOPS

MODULE - II
C++ Basics - Objects and Classes: Basics of object and class in C++, Private and public members, static data and function members, constructors and their types, destructors, operator overloading, type conversion

MODULE - III
Inheritance: Concept of Inheritance, types of inheritance: single, multiple, multilevel, hierarchical, hybrid, protected members, overriding, virtual base class- Polymorphism - Pointers in C++, Pointes and Objects, this pointer, virtual and pure virtual functions, implementing polymorphism - Introduction to exception, try-catch-throw, multiple catch, catch all, re-throwing exception, implementing user defined exceptions

Text Books:

C++ LAB – LIST OF EXERCISES
1. Program to illustrate class and objects
2. Program to illustrate inline member function
3. Program to illustrate static data and member functions
4. Program to illustrate constructors.
5. Program to illustrate friend functions
6. Program to illustrate operator overloading (Unary and Binary)
7. Program to illustrate function overloading.
8. Program to illustrate inheritance
9. Program to illustrate pointer to objects
10. Program to illustrate virtual functions & exception handling.
Paper Code: CSCS302

PHP PROGRAMMING

Prerequisite: Knowledge of programming

Objectives:
- To learn the fundamentals of PHP language
- To learn how to use PHP language to create websites

MODULE – I

MODULE – II
Functions – Strings – Arrays - Multidimensional Arrays- Extracting Multiple Values - Slicing an Array - Checking Whether an Element Exists - Traversing Arrays – Sorting - Objects – Terminology - Creating an Object - Accessing Properties and Methods - Declaring a Class – Introspection

MODULE – III
Web Techniques - HTTP Basics - Server Information - Processing Forms - Setting Response Headers - Maintaining State - Databases - Using PHP to Access a Database - Relational Databases and SQL MySQLi Object Interface – SQLite

Text Book:

PHP Lab – List of Exercises

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 accepts 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 : 'Thequick"brownfox'
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.
ANDROID PROGRAMMING

Prerequisite: Basic Knowledge of programming

Objectives:
- To study about the android architecture and the tools for developing android applications.
- To create an android application
- To learn about the user interfaces used in android applications
- To learn about how to handle and share android data

MODULE - I

MODULE – II
Linking Activities Using Intents – Resolving Intent Filter Collision - Returning Results from an Intent - Passing Data Using an Intent Object - Adding Fragments Dynamically - Life Cycle of a Fragment - Interactions between Fragments

MODULE – III
Understanding the Intent Object - Using Intent Filters – Adding Categories - Displaying Notifications - Android User Interface - Understanding the Components of a Screen - Adapting to Display Orientation - Managing Changes to Screen Orientation - Utilizing the Action Bar - Creating the User Interface Programmatically - Listening for UI Notifications - Designing Your User Interface with Views - Using Basic Views - Using Picker Views - Using List Views to Display Long Lists

Text Books:

ANDROID PROGRAMMING LAB - LIST OF EXERCISES
1. Develop an application that uses GUI components, Font and Colours
2. Develop an application that uses Layout Managers and event listeners.
3. Develop a native calculator application.
4. Write an application that draws basic graphical primitives on the screen.
5. Develop an application that makes use of database.
6. Develop an application that makes use of RSS Feed.
7. Implement an application that implements Multi-threading
8. Develop a native application that uses GPS location information.
9. Implement an application that writes data to the SD card.
10. Implement an application that creates an alert upon receiving a message.
PROGRAMMING IN PROLOG

Prerequisite: Knowledge of artificial intelligence and programming
Objective:

- To learn how to create programs based on artificial intelligence

MODULE – I
Overview of Prolog - An example program: defining family relations - Extending the example program by rules - A recursive rule definition - How Prolog answers questions - Declarative and procedural meaning of programs - Syntax and Meaning of Prolog Programs - Data objects – Matching - Declarative meaning of Prolog programs - Procedural meaning

MODULE - II
Lists, Operators, Arithmetic - Representation of lists – Some operations on lists - Operator notation - Arithmetic notations - Using Structures- Retrieving structured information from a database - Doing data abstraction - Simulating a non-deterministic automaton - Travel planning - The eight queens problem - Examples using cut - Negation as failure - Problems with cut and negation

Text Book:

PROGRAMMING IN PROLOG LAB – LIST OF EXERCISES

1. Program to add two numbers.
2. Program to categorize animal characteristics.
3. Program to read address of a person using compound variable.
4. Program of fun to show concept of cut operator.
5. Program to count number of elements in a list.
6. Program to reverse the list.
7. Program to append an integer into the list.
8. Program to replace an integer from the list.
9. Program to delete an integer from the list.
10. Program to show concept of list.
11. Program to demonstrate family relationship.
12. Program to show how integer variable is used in PROLOG program.
13. Write a program to solve 8 queens problem
15. Solve any problem using best first search.
16. Solve 8-puzzle problem using best first search
SOFTWARE TESTING

Prerequisite: Knowledge of software engineering and programming

Objectives:
- To make practitioners/students to understand the state-of-practice in testing industry by learning various types of software testing.

MODULE - I
Principles of testing – Software development life cycle models – phases of software project – life cycle models - Types of Testing – white box testing – static testing – structural testing – challenges in white box testing – black box testing – how to do black box testing

MODULE - II
Integration Testing – integration testing as a type of testing – integration testing as phase of testing – scenario testing – defect bash – System & Acceptance Testing - reason for conducting system testing – functional testing and non-functional testing - acceptance testing – summary of testing phases

MODULE - III

Text Book:

Software Testing Lab:
Implement the below mentioned exercises using any Testing Tool

1. Test Principles and Concepts
2. Test Management
3. Build the Test Environment
4. Test Planning Process
5. Test Design
6. Performing Tests
7. Defect Streaming and Correction
8. Acceptance Testing
9. Status of Testing
10. Test Reporting
Paper Code: CSCS404

ONLINE COURSE / IN-PLANT TRAINING (2 WEEKS) / ONE MONTH INTERNSHIP / MINI PROJECT

<table>
<thead>
<tr>
<th></th>
<th>L</th>
<th>T</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0</td>
<td>1</td>
<td>3</td>
</tr>
</tbody>
</table>
GENERIC ELECTIVE - I

Paper Code: CSCS124

MATHEMATICS FOR COMPUTER SCIENCE

Prerequisite: Knowledge of basic mathematics

Objectives:

- To learn rules and techniques to recognize valid logical argument
- To learn about the basic idea of logic with the algebra of proposition and predicate logic
- To learn graphs with all types and trees with all algorithms

MODULE - I


MODULE - II

Principal conjunctive and disjunctive normal forms Inference calculus-validity of conclusion using truth table- Rules of inference - Derivation process - Conditional proof-Indirect method of proof - Derivation of validity of conclusion by these methods

MODULE - III

Predicate calculus: Predicates, the statement function, variables and quantifiers- Predicate formulas-symbolizing the statement - Inference theory of the predicate calculus-Rules of specification and generalization-Derivation of conclusion using the rules of inference theory.

MODULE - IV

Graphs-Applications of graphs- Incident and degree-pendant and isolated vertices- Number of odd vertices in a graph- Isomorphism of graphs-sub graphs -Walks-paths and circuits - Connected graphs -Euler graphs operations on complete graphs- More on Euler graphs – Konigsberg bridge problem.

MODULE - V

Hamilton paths and circuits -Trees-properties of Trees with proof-Pendant vertices in a Tree-Distance and Center in a Tree-rooted and binary trees-spanning trees-Fundamental Circuits-Distance between spanning trees shortest spanning trees-Kruskal’s algorithm.

Text Books:

GENERIC ELECTIVE - II

Paper Code: CSCS125

NUMERICAL METHODS

Prerequisite: Knowledge of basic mathematics

Objectives:
- To learn about linear interpolation methods
- To learn about Lagrangian polynomials
- To learn about numerical integration methods
- To learn about single step methods

MODULE - I

MODULE - II
Lagrangian Polynomials – Divided differences – Interpolating with a cubic spline – Newton’s forward and backward difference formulas.

MODULE - III
Derivatives from difference tables – Divided differences and finite differences – Numerical integration by trapezoidal and Simpson’s 1/3 and 3/8 rules – Romberg’s method – Two and Three point Gaussian quadrature formulas

MODULE – IV

MODULE - V
Finite difference solution of second order ordinary differential equation – Finite difference solution of one dimensional heat equation by explicit and implicit methods – One dimensional wave equation and two dimensional Laplace and Poisson equations.

Text Books:
GENERIC ELECTIVE -III

Paper Code: CSCS235

PROBABILITY AND STATISTICS

Prerequisite: Knowledge in basic mathematics

Objectives:
- To learn how to handle situations involving more than one random variable and functions of random variables.
- To learn the notion of sampling distributions and have acquired knowledge of statistical techniques useful in making rational decision in management problems.
- To learn statistical methods designed to contribute to the process of making scientific judgments in the face of uncertainty and variation.

MODULE - I

MODULE – II

MODULE – III
Mathematical Expectation - Definition of Mathematical Expectation - Functions of Random Variables - Theorems on Expectation - Variance & Standard Deviation - Theorems on Variance - Standardized Random Variables - Special Probability Distributions - Binomial Distribution - Normal Distribution - Poisson Distribution

MODULE – IV
STATISTICS - Sampling Theory - Population and Sample - Statistical Inference - Sampling with and Without Replacement Random Samples - Random Numbers - Population Parameters - Sample Statistics - Sampling Distributions - Sample Mean - Sampling Distribution of Means - Sampling Distribution of Proportions - Sampling Distribution of Differences and Sums - Sample Variance - Sampling Distribution of Variances - Computation of Mean, Variance, and Moments for Grouped Data

MODULE – V
Curve Fitting, Regression, Correlation - Curve Fitting – Regression - The Method of Least Squares - The Least-Squares Line - The Least-Squares Line in Terms of Sample Variances and Covariance - The Least-Squares Parabola - Multiple Regression - Standard Error of Estimate - The Linear Correlation Coefficient - Generalized Correlation Coefficient - Rank Correlation
Text books:
2. S. P. Gupta, Statistical Methods, S. Chand and Sons.
GENERIC ELECTIVE - IV

Paper Code: CSCS236

AUTOMATA THEORY AND COMPUTATIONS
(Only statements and applications of Theorems)

Objectives:
- To understand the foundation of computing
- To realize the theoretical knowledge behind the computation
- To understand the construction of formal languages

MODULE - I

MODULE - II

MODULE - III
Turing Theory: Turing Machines – Computable Language and Functions – Techniques for TM Construction – Modification of TM.

MODULE - IV

MODULE - V

Text Books:
2. John E.Hopcraft and Jeffery D. Ullman, "Introduction to Automata theory, languages and computations", Narosa Publication.
3 "K.L.P. Mishra & N. Chandrasekaran” Theory of Computer Science (Automata, Languages and Computation ), PHI.
Non-Major Elective Course

Paper Code: CSCS171

FUNDAMENTALS OF INFORMATION TECHNOLOGY

<table>
<thead>
<tr>
<th>L</th>
<th>T</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Objective:
- To acquire the basic knowledge about computers

Module - I

Module - II
CPU and Memory - Secondary Story Devices - Input Devices - Output Devices.

Module - III
Introduction to Computer Software - Programming Language – Operating Systems - Introduction to Database Management System.

Module - IV
Computer Networks - WWW and Internet - Email - Web Design

Module - V

Text Books:
FUNDAMENTALS OF ‘C’ LANGUAGE

Prerequisite: Knowledge of computers

Objective:
- To learn how to solve common types of computing problems.
- To learn about data types and control structures of C
- To learn how to map problems to programming features of C.
- To learn how to write good portable C programs.

MODULE - I
Introduction to Programming - How to develop a program, Algorithms, Flow-charts, Types of Programming Languages, Compiler and Linker, Testing and Debugging a program, Documentation. Constants, Variables & Data Types - Character set, C Tokens, Identifiers and Keywords, Constants, Variables, Data types - Operators & Expressions - Managing Input & output operations

MODULE - II
Decision Making – Branching & Looping - Arrays - One dimensional array: Array Manipulation, Different operations on one dimensional arrays, two dimensional array, operations on two dimensional arrays, multi-dimensional array, dynamic arrays - Handling of Character Strings.

MODULE - III
Functions - Top down approach of problem solving, standard library functions, passing values between functions, scope rules of functions, calling convention, return type of functions, call by value and call by reference, recursive functions - Storage Classes - Scope and extent, Storage Classes in a single source file: auto, extern and static, register,

MODULE – IV
Structures and Unions - Defining a structure, Declaring Structure variables, accessing structure members, structure initialization, copying and comparing structure variables, operation on individual members, arrays of structures, arrays within structures, structures and functions, union, size of structure, bit fields.

MODULE - V
File Processing - Defining and Opening a file, closing a file, input/output operations on files, error handling during I/O operations, random access to files, Command Line Arguments.

Text Books:
WEB DESIGNING

Prerequisite: Knowledge of computers and internet

Objectives:
- To acquire the fundamental knowledge about internet & WWW
- To learn how to develop static and dynamic web pages / websites for any organization.
- To learn how to develop animated web pages

MODULE - I
Internet and the World Wide Web - Internet - Introduction to internet and its applications, E-mail, telnet, FTP, e-commerce, video conferencing, e-business. Internet service providers, domain name server, internet address, World Wide Web (WWW) - World Wide Web and its evolution, uniform resource locator (URL), browsers – internet explorer, Netscape navigator, opera, Firefox, chrome, Mozilla. search engine, web saver – apache, IIS, proxy server, HTTP protocol

MODULE – II
HTML5 – Introduction - formatting text by using tags, using lists and backgrounds, creating hyperlinks and anchors - Style sheets, CSS formatting text using style sheets, formatting paragraphs using style sheets.

MODULE – III
Page layout and navigation - Creating navigational aids: planning site organization, creating text based navigation bar, creating graphics based navigation bar, creating graphical navigation bar, creating image map, redirecting to another URL, creating division based layouts.

MODULE – IV
Tables, Forms and Media - Creating tables: creating simple table, specifying the size of the table, specifying the width of the column, merging table cells, using tables for page layout, formatting tables: applying table borders, applying background and foreground fills, changing cell padding, spacing and alignment

MODULE – V
Creating user forms: creating basic form, using check boxes and option buttons, creating lists, additional input types in HTML5, Incorporating sound and video: audio and video in HTML5, HTML multimedia basics, embedding video clips, incorporating audio on web page.

Text Book:
Paper Code: CSCS174

BASICS OF COMPUTERS & OFFICE AUTOMATION

Prerequisite: -- Nil

Objectives:

- To understand how to use software packages in day to day activities

MODULE – I
DOS: Internal & External commands; Wildcard Character; file name; Creating/Editing file; batch file - MS Windows: Windows Basic - Introduction to Windows- Using My Computer; Using Windows Explorer - Printing- Introduction to Accessories and Control Panel

MODULE – II

MODULE – III
MS Excel - Introduction and area of use -Working with MS Excel - concepts of Workbook & Worksheets - Working with Data & Ranges - Different Views of Worksheets - Column Freezing, Labels, Hiding, Splitting etc.;-Using different features with Data and Text - Use of Formulas, Calculations & Functions-Cell Formatting including Borders & Shading; Working with Different Chart Types - Printing of Workbook & Worksheets with various options.

MODULE -IV
MS PowerPoint - Introduction & area of use- Working with MS PowerPoint- Creating a New Presentation-Working with Presentation; Using Wizards- Slides & it’s different views; Inserting, Deleting and Copying of Slides - Working with Notes, Handouts, Columns & Lists- Adding Graphics, Sounds and Movies to a Slide-Working with PowerPoint Objects; Designing & Presentation of a Slide Show

MODULE – V
MS Access: DBMS Concept; Creating database, table, fields & its properties; Data types; Adding primary key into table; Relationship; Adding/Editing data; sorting; indexing; designing queries; using forms; Report generation.

Text Books: