PONDICHERRY 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

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

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.

   or

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 (IIIrd to VIth 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

COURSE	SUBJECT CODE	Paper	CREDITS		HOURS		
			Theory	Prac.	L	T	P
MIL	LTAM/LHIN/LTEL LMAL111	Language-I	3	-	4	1	0
ENG	ENGL112	English-I	3	-	4	1	0
DSC-1	CSCS113	Introduction to Problem Solving using C	3	-	4	1	0
DSC-2	CSCS114	Digital Electronics	3	-	4	1	0
AECC-1	PADM115	Public Administration	2	-	2	0	0
DSC-1(lab)	CSCS116	C lab	-	2	0	0	4
DSC-2 (lab)	CSCS117	Digital lab		2	0	0	4
		TOTAL	**18**		**30**		

SECOND SEMESTER

COURSE	SUBJECT CODE	Paper	CREDITS		HOURS		
			Theory	Prac.	L	T	P
MI4L	LTAM/LHIN/LTEL LMAL121	Language-II	3	-	4	1	0
ENG	ENGL122	English-II	3	-	4	1	0
DSC - 3	CSCS123	PYTHON Programming	3	-	3	0	0
DSC - 4	CSCS124	Data Structures and Algorithms	3	-	3	1	0
GE – 1 (1 out of 2)	CSCS125	Mathematics for Computer Science	3	-	3	0	0
	CSCS126	Numerical Methods					
AECC-2	ENVS127	EVS	2	-	2	0	0
DSC-3 (lab)	CSCS128	PYTHON lab	-	2	0	0	4
DSC-4(lab)	CSCS129	Data Structure & Algorithm lab	-	2	0	0	4
		TOTAL	**21**		**30**		

THIRD SEMESTER

COURSE	SUBJECT CODE	Paper	CREDITS		HOURS		
			Theory	Prac.	L	T	P
DSC – 5	CSCS231	Database Management System	3	-	3	1	0
DSC – 6	CSCS232	Visual Programming using C#	3	-	3	1	0
DSC – 7	CSCS233	Computer Networks	3	-	3	1	0
DSC – 8	CSCS234	Software Engineering	3	-	3	0	0
GE-**2 (1 out of 2)**	CSCS235	Probability and Statistics	3	-	3	0	0
	CSCS236	AutomataTheory & Computations					
DSC-6(lab)	CSCS237	Visual Programming & RDBMS lab	-	2	0	0	4
DSC-7(lab)	CSCS238	Network lab	-	2	0	0	4
1 out of 2	CSCS201	**SEC-1** Soft Skills	-	2		1	3
	CSCS202	**SEC-2** Office Automation					
		TOTAL	**21**		**30**		

FOURTH SEMESTER

COURSE	SUBJECT CODE	Paper	CREDITS		HOURS		
			Theory	Prac.	L	T	P
DSC – 9	CSCS241	Operating Systems	3	-	3	0	0
DSC – 10	CSCS242	Object Oriented Programming using Java	3	-	3	1	0
DSE – 1 DSE – 2 (2 out of 5 stream s)	CSCS243	Client/Server Computing	3	1	3	1	2
	CSCS244	Data Warehousing	-	-	0	0	0
	CSCS245	Object Oriented System Design	-	-	0	0	0
	CSCS246	Principles of Information Security	3	1	3	1	2
	CSCS247	Principles of Programming Languages	-	-	0	0	0
OE-2 **(1 out of 2)**	CSCS248	Distributed System	3	-	3		0
	CSCS249	Computer Graphics	-	-	0	0	0
DSC-10 (lab)	CSCS250	Object Oriented Programming lab	-	2	0	0	4
1 out of 2	CSCS301	**SEC-3** Programming with C++	1	2	0	1	3
	CSCS302	**SEC-4** Programming with PHP					
		TOTAL	**21**		**30**		

FIFTH SEMESTER

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

SIXTH SEMESTER

COURSE	SUBJECT CODE	Paper	CREDITS		HOURS		
			Theory	Prac.	L	T	P
DSC – 12	CSCS361	Microprocessors & Microcontrollers	3	-	3	0	0
DSC – 13	CSCS362	PROJECT	-	6	0	1	10
DSE - 5 DSE – 6 **(2 out of 5)**	CSCA363	Cloud Computing	3 3	1 1	3 3	1 1	2 2
	CSCA364	Foundations of Data Analytics					
	CSCA365	Software Quality Management					
	CSCA366	Ethical Hacking					
	CSCA367	Principles of Compiler Design					
DSC-12 (lab)	CSCS368	Microprocessor lab	-	2	0	0	4
		TOTAL	**19**		**30**		

Bachelor of Science (*COMPUTER SCIENCE)*

under *CHOICE-BASED CREDIT SYSTEM(CBCS)*

(Effective from the academic year 2017- 2018)

Paper Code: CSCS113

INTRODUCTION TO PROBLEM SOLVING USING C

L	T	P
4	1	0

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

Introduction to Computers - Characteristics of Computers, Uses of computers, Types and generations of Computers – Basic Computer Organization -Modules of a computer – Planning the Computer Program - Debugging, Types of errors - Documentation – Techniques of Problem Solving – Problem solving aspects – Top-Down aspects – Implementation of algorithms – Program verification - Flowcharting, decision table, algorithms, Structured programming concepts, Programming methodologies viz. top-down and bottom-up programming.

MODULE- II

C Programming Language- C Standard Library- C++ and Other C-based Languages- Object Technology- Introduction to C Programming - Memory Concepts-Decision Making - Secure C Programming - Structured Program Development in C- Algorithms-Pseudocode- Control Structures- if Selection Statement- while Repetition Statement - Assignment Operators- Increment and Decrement Operators- C Program Control- for Repetition Statement - switch Multiple-Selection Statement - do…while Repetition Statement - break and continue Statements-Logical Operators

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:

1. P. K. Sinha & Priti Sinha,"Computer Fundamentals", BPB Publications, 2007.
2. R.G. Tromey, "How to solve it by computer", Prentice Hall, 1982.
3. Paul Deital & Harvey Deital, "C How to Program", 7th edition, Pearson Education, 2013.

Paper Code: CSCS116

PROGRAMMING IN C LAB

L	T	P
0	0	4

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

Paper Code: CSCS114

DIGITAL ELECTRONICS & COMPUTER ORGANIZATION

L	T	P
4	1	0

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

Number systems & Conversions – Arithmetic of number systems – binary codes – BCD – The excess – 3code – Gray code – ASCII – EBCDIC - Introduction to Logic Circuits – logic functions & gates – Inversion – truth tables – logic gates – truth table of basics gates – timing diagrams of NOT, AND & OR gates – Boolean algebra – NAND& NOR logic gates - truth table of a logic circuit – de-morgan’s theorem

MODULE – II

Logic families – factors affecting performance of a logic family – register transistor logic – diode transistor logic – DCTL – ECL – TTL logic family – Karnaugh maps – two, three & four-variables K-map – loops in K-map – mapping of K-maps – don’t care condition

MODULE – III

Sequential logic circuits – sequential circuits – SR flip flop – D flip flop – JK flip flop – T flip flop – flip flop triggering – Shift registers – data movements in digital systems – classification of counters – Combinatorial logic circuits – designing procedure– code converters – multiplexers – multiplexer tree – demultiplexers /decoders – half & full adder – half & full subtractor – encoders – BCD adder

MODULE – IV

Basic Structure of Computers - Computer Types, Functional Modules, Basic operational Concepts, Bus Structures, Software, Performance, Multiprocessors and Multi-computers, Historical perspective - Input/Output Organization - Accessing I/O devices, Interrupts, Processor examples, Direct memory access, Buses, Interface circuits, Standard I/O interfaces.

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:

1. Morris Mano M, “Digital Logic and Computer Design”, Pearson Education, 4th edition, 2014.
2. S.S. Bhatti & Ragul Malhotra, ”A Textbook of Digital Electronics”, I.K. International publishing, New Delhi, 2013.
3. Carl Hamacher, Zvonko Vranesic, Safwat Zaky, “Computer Organization”, 5th edition, McGraw Hill, 2002.

Paper Code: CSCS117

DIGITAL LAB

L	T	P
0	0	4

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

L	T	P
3	0	3

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:

Ljubomir Perkovic, "Introduction to Computing Using Python: An Application Development Focus", John Wiley & Sons, 2012

Paper Code: CSCS128

PYTHON LAB

L	T	P
0	0	4

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) = (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*t)
    III. distance wrt velocity (s=(v*v-u*u)/2*a)

Paper Code: CSCS124

DATA STRUCTURES & ALGORITHMS

L	T	P
3	1	0

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

Recursion – Queue - operations - Singly Linked List – Operations - Application - Representation of a Polynomial - Polynomial Addition - Doubly Linked List - Operations.

MODULE-III

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

MODULE-IV

Basic Design and Analysis techniques of Algorithms, Correctness of Algorithm - Algorithm Design Techniques - Iterative techniques - Divide and Conquer - Dynamic Programming, Greedy Algorithms.

MODULE - V

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

Text Books

1. Ellis Horowiz, Sartaj Sahni and Anderson, "Fundamentals of Data Structure in C", University Press, 2nd edition, 2008.
2. T.H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein. "Introduction to Algorithms, PHI, 3rd edition. 2009.

Paper Code: CSCS129

DATA STRUCTURE & ALGORITHM LAB

L	T	P
0	0	4

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.

Paper Code: CSCS231

L	T	P
3	1	0

DATABASE MANEGEMENT 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

Overview of Database Management System - Introduction, file-based system, drawbacks of file-Based System, Data and information, Database, Database management System, Objectives of DBMS, Evaluation of Database management system, classification of Database Management System, DBMS Approach, advantages of DBMS, Anis/spark Data Model, data models, Components and Interfaces of Database Management System - Database Architecture, situations where DBMS is not Necessary - DBMS Vendors and their Products.

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

Relational Model – Introduction - ACID property - CODD Rules, relational data model, concept of key, relational integrity – primary key – foreign key - normalization – 1st normal form, 2nd normal form & 3rd normal form.

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

1. Abraham Silberschatz, Henry Korth, and S. Sudarshan, “Database System Concepts”, 6th edition, McGraw Hill, 2010,
2. Bulusu, “Oracle PL/SQL Programming”, OReilly, 5th edition, 2009.
3. Steve Bobrowski, “Hands-On Oracle Database 10g Express Edition for Windows”, Tata McGraw Hill, 2010.

Paper Code: CSCS232

L	T	P
3	1	0

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
Introduction - C, C++, Objective-C, Java and C# - Extensible Markup Language (XML) - Introduction to Microsoft .NET - The .NET Framework and the Common Language Runtime - Introduction to Object Technology - Introduction to C# Applications - Creating a Simple Application in Visual C# Express - Formatting Text with Console.Write and Console.WriteLine - Another C# Application: Adding Integers – Arithmetic - Decision Making: Equality and Relational Operators - Strings and Characters

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
Graphical User Interfaces with Windows Forms – Introduction - Windows Forms - Control Properties and Layout - Labels, TextBoxes and Buttons - GroupBoxes and Panels - CheckBoxes and RadioButtons - NumericUpDown Control - Mouse-Event Handling- Keyboard-Event Handling – Menus- various controls - Multiple Document Interface (MDI) Windows - Visual Inheritance - User-Defined Controls

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:
Paul Deitel & Harvey Deitel, “C# 2010 for Programmers”, Pearson Education, 4th edition, 2011.

Paper Code: CSCS237

L	T	P
0	0	4

VISUAL PROGRAMMING & DBMS LAB

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

7. Create program using Database Controls
8. 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

Paper Code: CSCS233

L	T	P
3	1	0

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

Data link layer - Design Issues, Error Detection and Correction - Elementary Data Link Protocols - Sliding Window Protocols - Network Layer - Design Issues, Routing Algorithms - Congestion Control Algorithms

MODULE - III

Internetworking - Transport Layer - The Transport Service – Service provided to the Upper Layers, elements of Transport Protocols – Addressing, Connection Establishment, Connection Release, Flow Control & Buffering - TCP - Introduction, TCP Service model, TCP Protocol, TCP Segment Header, TCP connection Establishment, TCP Connection Release, TCP Transmission Policy, TCP Congestion Control

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

The World Wide Web - Architectural Overview, Static Web Documents, Dynamic Web Documents, Hyper Text Transfer Protocol (HTTP) - Introduction to Security.

Text Books:

Andrew S. Tanenbaum, “Computer Networks”, Prentice Hall India, 5th edition, 2010.

Paper Code: CSCS238

L	T	P
0	0	4

NETWORKS LAB

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.

Paper Code: CSCS234

L	T	P
3	0	0

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

Introduction to Software Engineering – evolving role of software – defining software engineering – changing nature of software – software myths – terminologies – role of software development – software life cycle models – build & fix model – waterfall model – incremental model – evolutionary model – unified model – selection of a life cycle model

MODULE – II

Software Requirements: Analysis & Specifications – requirements engineering – type of requirements – feasibility studies – requirements elicitation – requirement analysis - – requirement documentation - – requirement validation - – requirement management – Case studies

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

Software Metrics – Software & Metrics: What & Why – token count – data structure metrics – information flow metrics – object oriented metrics – Use-Case metrics – metrics analysis - software reliability – basic concepts – software reliability models – capability maturity model

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:

1. K.K. Aggarwal & Yogesh Singh, “Software Engineering”, New Age International Publishers, 2012.
2. Roger S. Pressman, “Software Engineering: A Practitioner’s Approach”, McGraw Hill, 7th edition, 2010.

Paper Code: CSCS241

L	T	P
3	0	0

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
Introduction - Mainframe systems – Desktop Systems – Multiprocessor Systems – Distributed Systems – Clustered Systems – Real Time Systems – Handheld Systems - Hardware Protection - System Components – Operating System Services – System Calls – System Programs - Process Concept – Process Scheduling – Operations on Processes – Cooperating Processes – Inter-process Communication.

MODULE - II
Threads – Overview – Threading issues - CPU Scheduling – Basic Concepts – Scheduling Criteria – Scheduling Algorithms – Multiple-Processor Scheduling – Real Time Scheduling - The Critical-Section Problem – Synchronization Hardware – Semaphores – Classic problems of Synchronization – Critical regions – Monitors.

MODULE - III
System Model – Deadlock Characterization – Methods for handling Deadlocks - Deadlock Prevention – Deadlock avoidance – Deadlock detection – Recovery from Deadlocks - Storage Management – Swapping – Contiguous Memory allocation – Paging – Segmentation – Segmentation with Paging.

MODULE - IV
Virtual Memory – Demand Paging – Process creation – Page Replacement – Allocation of frames – Thrashing - File Concept – Access Methods – Directory Structure – File System Mounting – File Sharing – Protection

MODULE - V
File System Structure – File System Implementation – Directory Implementation – Allocation Methods – Free-space Management. Kernel I/O Subsystems - Disk Structure – Disk Scheduling – Disk Management – Swap-Space Management - Case Study: The Linux System & Windows

Text Books:

1. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, “Operating System Concepts”, 6th edition, John Wiley & Sons, 2003.
2. Harvey M. Deitel, “Operating Systems”, 2nd edition, Pearson Education, 2002.

Paper Code: CSCS242

L	T	P
3	1	0

OBJECT ORIENTED PROGRAMMING USING JAVA

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

Exception Handling – GUI components – Introduction – Overview of Swing components – Swing vs AWT – SWING: Displaying Text and Images in a Window - Text Fields and an Introduction to Event Handling with Nested Classes - Common GUI Event Types and Listener Interfaces - How Event Handling Works – various event handling – layout manager

MODULE – IV

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

MODULE – V

Applets & Java Web Start – applet life-cycle – sandbox security model – Java web start & Java Network Launch Protocol (JNLP) – Accessing databases with java database connectivity (JDBC)

Text Books:

Paul Deital & Harvey Deital, ”Java: How to Program”, Pearson Education, 10th edition, 2015.

Paper Code: CSCS250

L	T	P
0	0	4

OBJECT ORIENTED PROGRAMMING USING JAVA LAB

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

Paper Code: CSCS351

L	T	P
3	0	0

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

Web Essentials: Clients, Servers, and Communication - Internet - Basic Internet Protocols - The World Wide Web - World Wide Web - HTTP Request Message - HTTP Response Message - Web Clients - Web Servers

MODULE - II

Markup Languages: XHTML - An Introduction to HTML - HTML's History and Versions - Basic XHTML Syntax and Semantics - Some Fundamental HTML Elements. - Relative URLs - Lists - Tables - Frames - Forms - Defining XHTML's Abstract Syntax: XML - Creating HTML Documents - Style Sheets: CSS-Introduction to Cascading Style Sheets - Cascading Style Sheet Features - CSS Core Syntax - Style Sheets and HTML - Style Rule Cascading and Inheritance - Text Properties - CSS Box Model

MODULE – III

Client-Side Programming: JavaScript Language - History and versions of JavaScript - Introduction to JavaScript - JavaScript in Perspective - Basic Syntax - Variables and Data Types - Statements. - Operators - Literals - Functions - Objects - Arrays - Built-in Objects - Host Objects: Browsers and the DOM - Introduction to the Document Object Model- Intrinsic Event Handling - DOM History and Levels

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:

1. Jeffery C. Jackson, "Web Technologies: A Computer Science Perspective", Pearson Education, 2007.
2. Julie C. Meloni," Sams Teach Yourself; HTML, CSS, and JavaScript All in One", SAMS, 2014.

Paper Code: CSCS239

L	T	P
0	0	4

WEB TECHNOLOGY LAB

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.

Paper Code: CSCS361

L	T	P
3	0	0

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

Intel 8085 Microprocessor: Introduction - Need for Microprocessors – Evolution – Intel 8085 Hardware - Architecture – Pin description - Internal Registers – Arithmetic and Logic Unit – Control Unit – Instruction word size - Addressing modes – Instruction Set – Assembly Language Programming - Stacks and Subroutines - Timing Diagrams. Evolution of Microprocessors – 16-bit and 32-bit microprocessors.

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

Memory & I/O Interfacing: Types of memory – Memory mapping and addressing – Concept of I/O map – types – I/O decode logic – Interfacing key switches and LEDs – 8279 Keyboard/Display Interface - 8255 Programmable Peripheral Interface – Concept of Serial Communication – 8251 USART – RS232C Interface.

MODULE – IV

Intel 8086 Microprocessor: Introduction-Intel 8086 Hardware – Pin description – External memory Addressing – Bus cycles – Interrupt Processing. Addressing modes - Instruction set – Assembler Directives.

MODULE – V

Microcontroller**:** Intel 8051 Microcontroller: Introduction – Architecture – Memory Organization – Special Function Registers – Pins and Signals – Timing and control – Port Operation – Memory and I/O interfacing – Interrupts – Instruction Set and Programming.

Text Books:

Krishna Kant, “Microprocessors and Microcontrollers – Architectures, Programming and System Design 8085, 8086, 8051, 8096”, PHI, 2008.

Paper Code: CSCS368

L	T	P
0	0	4

MICROPROCESSOR LAB

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
6. Code Conversion, Decimal Arithmetic and Matrix Operations.
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

Paper Code: CSCS362

L	T	P
0	1	10

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

L	T	P
3	1	2

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

Architectures of client/server systems – introduction – components – principles behind client/server systems – client components – server components – communication middleware components – architecture for business information system – existing client/server architecture

MODULE – III

Client/Server databases – Introduction – client/server in respect of databases – client/server database architecture – database middleware component – access to multiple databases – distributed client/server database systems – distributed DBMS – web/database system for client/server applications

MODULE – IV

client/server application components – introduction – technologies for client/server application – services of a client/server application – categories of client/server applications – client services – server services – client/server application connectivity – client/server application: Layered Architecture

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:

Subhash Chandra Yadav & Sanjay Kumar Singh,"An Introduction to Client/Server Computing", New Age International Publishers, 2009.

Paper Code: CSCS352

L	T	P
3	1	2

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:

Michael P. Papazoglou,” Web Services: Principles and Technology”, Pearson Education, 2008.

Paper Code: CSCS363

L	T	P
3	1	2

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
Computing Paradigms - Cloud Computing Fundamentals - Motivation for Cloud Computing - Defining Cloud Computing - Principles of Cloud computing - Cloud Ecosystem - Requirements for Cloud Services - Cloud Application - Benefits and Drawbacks - Cloud Computing Architecture and Management Cloud Architecture - Anatomy of the Cloud- Network Connectivity in Cloud Computing - Applications on the Cloud- Managing the Cloud - Migrating Application to Cloud

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

MODULE – III
Technological Drivers for Cloud Computing - SOA and Cloud – Virtualization-Multicore Technology - Memory and Storage Technologies - Networking Technologies - Web 2.0 - Web 3.0 - Software Process Models for Cloud-Programming Models - Operating System - Application Environment

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:
K. Chandrasekaran, “Essentials of Cloud Computing“, CRC Press, Taylor & Francis Group, 2015.

DISCIPLINE SPECIFIC ELECTIVE (DSE) PAPERS

Specialization Stream – II [BUSINESS INTELLIGENCE STREAM]

Paper Code: CSCS244

L	T	P
3	1	2

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
Introduction – Data warehouse delivery method – system process – typical process flow within a data ware house – query management process – process architecture – meta data-data mart.

MODULE - II
Design aspects – Designing dimension tables – Designing star flake schema – Multi dimensional schema – partitioning strategy aggregations – Data mart- Meta data – System Data warehouse process manager.

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
Planning and Development **-** Capacity planning – Estimating the load – Tuning the data warehouse – Assessing performance –Tuning the data load and queries – Testing data warehouse – Development of test plan – Testing the data base and operational environment.

MODULE - V
Case Studies **-** Data Warehousing in the Tamilnadu Government - Data Warehouse for the Ministry of commerce- Data Warehouse for the government of Andhra Pradesh- Data Warehousing in Hewlett –Packard- Data Warehousing in Levi Strauss- Data Warehousing in the World Bank- HARBOR, A Highly available Data Warehouse-A typical Business data Warehouse for a Trading company.

Text Books:

1. Sam Anahory & Dennis Murray, “Data Warehousing in the real world”, Pearson Education.
2. Prabhu C.S.R, “Data Warehousing: Concepts, Techniques, Products and Applications”, PHI Learning, 3rd edition, 2009.

Paper Code: CSCS253

L	T	P
3	1	2

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

MODUEL – 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

Association Pattern Mining – Introduction- Frequent Pattern Mining Model - Association Rule Generation Framework - Frequent Item set Mining Algorithms- Brute Force Algorithms - Apriori Algorithm - Enumeration-Tree Algorithms - Pattern Summarization

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:

Charu C. Aggarwal**,** Data Mining: The Textbook, Springer, 2015.

Paper Code: CSCS264

L	T	P
3	1	2

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

Modeling Methods - Choosing and evaluating models – mapping problems to machine learning, evaluating clustering models, validating models – cluster analysis – K-Means algorithm, Naïve Bayes – Memorization Methods – Linear and logistic regression – unsupervised methods.

MODULE - III

Introduction to R - Language: Reading and getting data into R – ordered and unordered factors – arrays and matrices – lists and data frames – reading data from files – probability distributions – statistical models in R - manipulating objects – data distribution.

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:

1. Nina Zumel, John Mount, “Practical Data Science with R”, Manning Publications, 2014.

2. Jure Leskovec, Anand Rajaraman, Jeffrey D.Ullman, “Mining of Massive Datasets”, Cambridge University Press, 2014.

3. Mark Gardener, “Beginning R - The Statistical Programming Language”, John Wiley & Sons, Inc., 2012.

4.W. N. Venables, D. M. Smith and the R Core Team, “An Introduction to R”, 2013.

DISCIPLINE SPECIFIC ELECTIVE (DSE) PAPERS

Specialization Stream – III [SOFTWARE ENGINEERING STREAM]

Paper Code: CSCS245

L	T	P
3	1	2

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

Introduction to UML - Use case diagram - Requirement Capture with Use case- Building blocks of Use Case diagram - Relationships between use cases - extend, include, generalize - Activity diagram - Elements of Activity Diagram - Object - Control and Object flow, Transition - Guidelines for Creating Activity Diagrams- Activity Diagram - Action Decomposition - Partition - Swim Lane - Static structural view - Relationships among classes- Dependency relationships among classes, notations - Package & interface notation - Object diagram notations and modelling - relations among objects.

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:

1. Ali Bahrami, “Object Oriented Systems Development using the unified modeling language”, 1st edition, TMH, 2008.
2. Grady Booch, James Rumbaugh, Ivar Jacobson, “The Unified Modeling Language User Guide”, 2nd edition, Pearson Education, 2007.

Paper Code: CSCS354

L	T	P
3	1	2

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

Introduction **-** Software Architecture –Architecture Structures and Views – Importance of Software Architecture – Predicting System Quality – Influencing Organizational Structure – Improving Cost and Schedule estimates – Context of Software architecture.

MODULE - II

Quality Attributes **-** Understanding quality attributes – availability – interoperability – modifiability - performance and security – testability - usability – quality attribute modelling and analysis.

MODULE - III

Architecture in the Life Cycle **-** Architecture in the agile projects – Architecture and requirements – Designing and documentation – Implementation and testing – Architecture reconstruction and conformance.

MODULE - IV

Architecture and Business **-** Economic analysis of Architecture – Architecture competence – Architecture and Software product lines – Case Studies.

MODULE - V

Architecture in Advance **-** Architecture in Cloud - Cloud Definition – Service Model – Economic Justification – Base Mechanism – Architecture for the Edge – Edge Document system – SDLC – Metropolis Model.

Text Books:

1. Len Bass, Paul Clements, Rick Kazman, “Software Architecture in Practice”, 3rd edition Pearson, 3rd edition, 2013.
2. Mary Shaw, David Garlan, “Software Architecture: Perspectives on an Emerging Discipline”, PHI, 2009.

Paper Code: 365

L	T	P
3	1	2

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
Introduction - Software Quality Challenge - Software Quality Factors - Components of the Software Quality Assurance System - Pre-Project Software Quality Components - Contract Review - Development and Quality Plans

MODULE - II
SQA Components in The Project Life Cycle - Integrating Quality Activities in the Project Life Cycle – Reviews - Software Testing – Strategies - Software Testing – Implementation - Assuring the Quality of Software Maintenance - Assuring The Quality of External Participants' Parts – Case Tools and their effect on Software Quality.

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
Software Quality Metrics - Objectives of quality measurement- Process metrics- Product metrics - Software Quality Costs - Objectives of cost of software quality metrics- classic model of cost of software quality - Maturity Models - Basic Idea in Software Process - Capability Maturity Mode Capability Maturity Model

Text Books:

1. Daniel Galin - “Software Quality Assurance: From Theory to Implementation” - Pearson Addison-Wesley, 2012.
2. Kshirasagar Naik and Priyadarshi Tripathy, “Software Testing and Quality Asurance”, John Wiley, 2008.
3. Allen Gilles, “Software quality: Theory and management”, 2nd edition, Cengage Learning, 2003.

DISCIPLINE SPECIFIC ELECTIVE (DSE) PAPERS

Specialization Stream – IV [INFORMATION SECURITY STREAM]

Paper Code: 246

L	T	P
3	1	2

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

Introduction – History of Information Security – defining secuirty – CNSS Security Model – Components of an Information Security – Approaches to Information Security Implementation – System Development Life Cycle - Security Systems Development Life Cycle -Security Professionals and the Organization - Information Security: Is it an Art or a Science?

MODULE – II

The Need for Security – Introduction - Business Needs First – Threats – Attacks Secure Software Development

MODULE – III

Legal, Ethical, and Professional Issues in Information Security - Law and Ethics in Information Security - Relevant U.S. Laws - International Laws and Legal Bodies - Ethics and Information Security - Codes of Ethics and Professional Organizations

MODULE – IV

Risk Management - Introduction - An Overview of Risk Management - Risk Identification - Risk Assessment - Risk Control Strategies - Selecting a Risk Control Strategy - Quantatitive Versus Qualitative Risk Control Practices - Risk Management Discussion Points

MODULE – V

Planning for Security – Introduction - Information Security Planning and Governance - Information Security Policy, Standards, and Practices - Security Education, Training, and Awareness Program- Continuity Strategies - Implementing Information Security - Information Security Project Management - Technical Aspects of Implementation - Nontechnical Aspects of Implementation - Information Systems Security Certification and Accreditation

Text Book:

Michael E. Whitman & Herbert J. Mattord, “Principles of Information Security”, Course Technology, Cengage Learning, 4th edition, 2011.

Paper Code: 355

L	T	P
3	1	2

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

Computer Network Fundamentals - Introduction - Computer Network Models- Computer Network Types - Data Communication Media Technology - Network Topology Network Connectivity and Protocol - Network Services - Network Connecting Devices- Network Technologies

MODULE – II

Understanding Network Security - Defining Network Security - Security Services - Security Standards - Elements of Security - Security Threats to Computer Networks- Sources of Security Threats - Security Threat Motives - Security Threat Management - Security Threat Correlation

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

Dealing with Network Security Challenges - Access Rights - Access Control Systems – Authorization - Types of Authorization Systems – Authentication - Multiple Factors and Effectiveness of Authentication - Authentication Elements Types of Authentication - Authentication Methods Developing an Authentication Policy

MODULE – V

Cryptography – Definition - Block Ciphers - Symmetric Encryption - Public Key Encryption - Key Management: Generation, Transportation, and Distribution - Public Key Infrastructure (PKI) - Hash Function - Digital Signatures – Firewalls - Types of Firewalls - Configuration and Implementation of a Firewall - Firewall Forensics - Firewall Services and Limitations - Computer Network Security Protocols and Standards - Application Level Security -Security in the Transport Layer Security in the Network Layer

Text Books:

1. Kizza & Joseph Migga**, “**Computer Network Security”, Springer, 2005.
2. William Stallings, “Cryptography & Network Security”, Pearson Education, 4th edition, 2010.

Paper Code: 366

L	T	P
3	1	2

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

Data Theft in Organizations, Elements of Information Security, Authenticity and Non-Repudiation, Security Challenges, Effects of Hacking, Hacker – Types of Hacker, Ethical Hacker, Hacktivism - Role of Security and Penetration Tester, Penetration Testing Methodology, Networking & Computer Attacks – Malicious Software (Malware), Protection Against Malware, Intruder Attacks on Networks and Computers, Addressing Physical Security – Key Loggers and Back Doors

MODULE II

Web Tools for Foot Printing, Conducting Competitive Intelligence, Google Hacking, Scanning, Enumeration, Trojans & Backdoors, Virus & Worms, Proxy & Packet Filtering, Denial of Service, Sniffer, Social Engineering – shoulder surfing, Dumpster Diving, Piggybacking.

MODULE III

Physical Security – Attacks and Protection, Steganography – Methods, Attacks and Measures, Cryptography – Methods and Types of Attacks, Wireless Hacking, Windows Hacking, Linux Hacking

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:

1. Patrick Engebretson, “The Basics of Hacking and Penetration Testing: Ethical Hacking and Penetration Testing Made Easy”, Syngress Basics Series – Elsevier, 2011.
2. Michael T. Simpson, Kent Backman, James E. Corley, “Hands-On Ethical Hacking and Network Defense”, Second Edition, CENGAGE Learning, 2010.

DISCIPLINE SPECIFIC ELECTIVE (DSE) PAPERS

Specialization Stream – V [SCIENCE OF PROGRAMMING LANGUAGE STREAM]

Paper Code: 247

L	T	P
3	1	2

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-Conditional, 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

Case Studies: Functional programming Languages- Introduction- Mathematical functions- functional programming languages – LISP - Application of functional languages- Logic programming Languages – Introduction to predicate Calculus - An overview of logic programming- Applications of logic programming

Text Books

1. Ellis Horowitz, “Fundamentals of programming languages”, Galgotia Publications, 1998
2. Robert W. Sebesta, “Concepts of programming languages”, Addison-Wesley, 1996.

Paper Code: 356

L	T	P
3	1	2

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:

1. Leland L. Black & D. Manjula, “System Software”, Pearson Education, 3rd edition, 2002.

2. A.V. Aho, R. Semi, J.D. Ullman, “Compilers - Principles, techniques and tools”, 2nd edition, Pearson Education, 2011.

Paper Code: 367

L	T	P
3	1	2

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

Introduction to Compiler – Compliers – Analysis of the source program – Phases of compiler – cousins of the compiler – grouping of phases – compiler construction tools – simple one pass compiler – overview – syntax definition – syntax directed translation – parsing – lexical analysis – incorporating symbol table

MODULE - II

Lexical analysis – role of lexical analyzer – input buffering – specification of tokens – recognition of token – finite automata – from a regular expression to NFA – design of a lexical analyzer generator – optimization of DFA based pattern matches

MODULE - III

Syntax Analysis – role of the parser – top-down parsing – bottom-up parsing – operator precedence parsing – LR parsers – using ambiguous grammer – syntax directed translation – syntax directed definitions – construction of syntax trees

MODULE - IV

Runtime environment – source language issues – storage orgranization – storage allocation stategies – storage allocation in C, parameter parsing – intermediate code generation – intermediate languages – declaration – assignment statements

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:

Alfred Aho, Jeffrey D.Ullman, Ravi Sethi , “Compiler Design-Principles Techniques and Tools”, 2nd edition, Pearson Education, 2011.

OPEN ELECTIVE – I

Paper Code: 248

L	T	P
3	0	0

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
Introduction – Examples of Distributed Systems–Trends in Distributed Systems – Focus on resource sharing – Challenges - Case study: World Wide Web.

MODULE II
System Model – Inter process Communication - the API for internet protocols – External data representation and Multicast communication - Network virtualization: Overlay networks. Case study: MPI

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:
George Coulouris, Jean Dollimore, Tim Kindberg, “Distributed Systems Concepts and Design”, Addison Wesley, 5th edition, 2011.

OPEN ELECTIVE – II

Paper Code: 249

L	T	P
3	0	0

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
Overview of Computer Graphics System: Video Display Devices – Raster Scan Systems – Random – Scan Systems - Graphics Monitors and Workstations – Input Devices – Hardcopy Devices – Graphics Software.

MODULE - II
Output Primitives: Line Drawing Algorithms – Loading the Frame Buffer – LineFunction – Circle – Generating Algorithms - Attributes of Output Primitives: Line Attributes – Curve Attributes – Color and Gray scale levels– Area fill Attributes – Character Attributes – Bundled Attributes – Inquiry Functions.

MODULE - III
2D Geometric Transformations: Basic Transformation – Matrix Representations – Composite Transformations – Window to View port Co-Ordinate Transformations - Clipping: Point Clipping – Line Clipping – Cohen-Sutherland Line Clipping – Liang Barsky Line Clipping – Polygon Clipping – Sutherland – Hodgman Polygon Clipping – Curve Clipping – Text Clipping.

MODULE - IV
Graphical User Interfaces and Interactive Input Methods: The User Dialogue – Input of Graphical Data – Input Functions – Interactive Picture Construction Techniques – Three Dimensional Concepts: 3D-Display Methods – #Three Dimensional Graphics Packages

MODULE - V
3D Geometric and Modeling Transformations: Translation – Scaling – Rotation – Other Transformations.Visible Surface Detection Methods: Classification of Visible Surface Detection Algorithm –Backface Detection – Depth-Buffer Method – A-Buffer Method – Scan-Line Method –Applications of Computer Graphics.

Text Book:
Donald Hearn M. Pauline Baker, Computer Graphics C Version, 2nd edition, Pearson Education, 2014.

OPEN ELECTIVE – III

Paper Code: CSCS357

L	T	P
3	0	0

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 Al & 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

Searching Techniques **-** Searching-Searching for solutions, uniformed search strategies – Breadth first search, depth first Search. Search with partial information (Heuristic search) Greedy best first search- A* search Game Playing- Adversial search, Games, minimax, algorithm, optimal decisions in multiplayer games, Alpha-Beta pruning, Evaluation functions, cutting of search.

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:

1. Rich E, Knight K, “Artificial Intelligence”, 2nd edition, TMH, 2005.
2. Stuart Russel, Peter Norvig “AI – A Modern Approach”, 2nd edition, Pearson Education, 2007.
3. Ivan Bratka, “PROLOG Programming for Artificial Intelligence”, Addison Wesley, 1986.

OPEN ELECTIVE – IV

CSCS358

L	T	P
3	0	0

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

Electronic commerce: Business models - Electronic data interchange conventional trading process – Defining EDI – building blocks of EDI systems: Layered Architecture – Value added networks – benefits of EDI – application of EDI

MODULE – III

Electronic commerce: Architectural framework - Electronic commerce: Information Publishing Technology – Information publishing – web browsers – Hypertext Markup Language – Common Gateway Interface – multimedia content – Other multimedia objects – virtual reality modeling language

MODULE - IV

Electronic commerce: Securing the Business on Internet – Security policies, procedures & practices – site security – protecting the network – firewalls – securing the Web (HTTP) service - Electronic commerce: securing network transaction – transaction security – cryptology - cryptographic algorithms – public key algorithms – authentication protocols – digital signatures – electronic mail security – security protocols for web commerce

MODULE – V

Electronic Payment Systems – introduction to payment systems – online payment systems – pre-paid electronic payment systems – post-paid electronic systems requirement metrics of a payment system - Mobile commerce – Introduction, framework, and models- benefits of m-commerce – impediments in mobile commerce – mobile commerce framework

Text Book:

Bharat Bhasker, "Electronic Commerce: Framework, Technologies and Applications", McGraw Hill Education (India), 4th edition, 2013.

SKILL ENHANCEMENT COURSES (SEC)

Paper Code: CSCS201

L	T	P
0	1	3

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

Job interviews: Pre – interview preparation techniques – Interview questions – Answering strategies – Frequently asked interview questions – Projecting a positive image – Alternative interview formats - Group Discussion: Nature of group discussion – Characteristics of successful group discussions – Selection group discussion – Group discussion strategies – Techniques for individual contribution – Group interaction strategies.

MODULE - III

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

Text Book:

M. Ashraf Rizvi , "Effective Technical Communication", Tata McGraw – Hill Education, 2005

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

Paper Code: CSCS201

L	T	P
0	1	3

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

Word processing - Introduction - Introduction to MS Word - Working with MS Word - Creating a New Document-Different Page Views and layouts - Working with Styles, Text Attributes; Paragraph and Page Formatting - Text Editing using various features - Advanced Features of MS-Word – Inserting – Page Numbers, Pictures, Files, Auto texts, Symbols - Working with Columns, Tabs & Indents - Creation & Working with Tables -Margins & Space management in Document - Mail Merge- MS PowerPoint - Creating a New Presentation-Working with Presentation; Using Wizards- Slides & it's different views; Inserting, Deleting and Copying of Slides - Adding Graphics

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:

1. Dinesh Maidasani , Straight to the Point – MS Office 2010, Laxmi Publications, 2010.
2. Sherry Kinkoph Gunter, Master Visually Microsoft Office 2010, WILEY, 2010.

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.

Paper Code: CSCS301

L	T	P
0	1	3

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:

1. E Balagurusamy , Object Oriented Programming with C++, 5th edition, Tata McGraw, 2011.
2. Deitel and Deitel , “C++: How to Program”, 9th Edition, Pearson, 2013.

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

L	T	P
0	1	3

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
Introduction to PHP – brief history – installing PHP – Language basics – Lexical structure – data types – variables – expressions and Operators – flow-control statements – including code – embedding PHP in web pages

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:
Kevin Tatroe, Peter MacIntyre, and Rasmus Lerdorf, "Programming PHP", O'Reilly, 3rd edition, 2013.

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 accept the number as an argument.
3. WAP to check whether the given number is prime or not.
4. Create a PHP page which accepts string from user. After submission that page displays the reverse of provided string.
5. Write a PHP function that checks if a string is all lower case.
6. Write a PHP script that checks whether a passed string is palindrome or not? (A palindrome is word, phrase, or sequence that reads the same backward as forward, e.g., madam or nurses run)
7. WAP to sort an array.
8. Write a PHP script that removes the whitespaces from a string.
   Sample string : 'The quick " " brown fox'
   Expected Output : 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.

Paper Code: CSCS401

L	T	P
0	1	3

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
Introduction - Android - Android Versions - Features of Android - Architecture of Android - Obtaining the Required Tools - Android SDK - Installing the Android SDK Tools - Configuring the Android SDK Manager – Eclipse - Android Development Tools (ADT) - Creating Android Virtual Devices (AVDs) - Types of Android Application - Anatomy of an Android Application - Activities

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:

1. Wei - Meng Lee, "Beginning Android Application Development", 2nd edition, John Wiley, 2012.
2. Reto Meier, "Android 6 for Programmers: An App-driven Approach", 3rd edition, Pearson Education, 2016.
3. Deital & Deital, "Android for Programmers: An App-Driven Approach", 1st edition, Pearson Education, 2012.

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.

Paper Code: CSCS402

L	T	P
0	1	3

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:

1. Saroj Kaushik, :Logic and PROLOG Programing", New Age International Publishers, 2nd reprint, 2014.
2. Lvan Bratko, Prolog Programming For Artificial Intelligence, Addison Wesley, 1986.

PROGRAMMING IN PROLOG LAB – LIST OF EXCERCISES

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
14. Solve any problem using depth first search.
15. Solve any problem using best first search.
16. Solve 8-puzzle problem using best first search

Paper Code: CSCS403

L	T	P
0	1	3

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
Performance testing – factors governing performance testing – methodology for performance testing – tools for performance testing – process for performance testing – challenges

Text Book:
Srinivasan Desikan & Goplaswamy Ramesh, "Software Testing: Principles & Practices", Dorling Kindersley (India) Pvt. Ltd, 2013.

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

L	T	P
0	1	3

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

GENERIC ELECTIVE - I

Paper Code: CSCS124

L	T	P
3	0	0

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

Connectives, Statement formulae, well-formed formulae-Tautologies - Equivalence of Statement formulae, Duality law-Tautological implications- Functionally complete set of connectives-NAND and NOR connectives.

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:

1. J.P.Tremblay & R.Manohar, “Discrete Mathematical Structures with applications to Computer science”, Tata McGraw-Hill, 2002.
2. Narsingh Deo, “Graph Theory with applications to Engineering and Computer science”, PHI, 2002.

GENERIC ELECTIVE - II

Paper Code: CSCS125

L	T	P
3	0	0

NUMERCIAL 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

Linear interpolation methods (method of false position) – Newton’s method – Statement of Fixed Point Theorem – Fixed point iteration: x=g(x) method – Solution of linear system by Gaussian elimination and Gauss-Jordon methods - Iterative methods: Gauss Jacobi and Gauss-Seidel methods- Inverse of a matrix by Gauss Jordon method – Eigenvalue of a matrix by power method.

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

Single step methods: Taylor series method – Euler and modified Euler methods – Fourth order Runge – Kutta method for solving first and second order equations

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:

1. Gerald, C.F, and Wheatley, P.O, “Applied Numerical Analysis”, 6th edition, Pearson Education Asia, New Delhi, 2002.
2. Kandasamy, P., Thilagavathy, K. and Gunavathy, K., “Numerical Methods”, S. Chand Co. Ltd., New Delhi, 2003
3. Balagurusamy, E., “Numerical Methods”, Tata McGraw-Hill, 1999.
4. Burden, R.L and Faires, T.D., “Numerical Analysis”, 7th edition, Thomson Asia Pvt. Ltd., Singapore, 2002

GENERIC ELECTIVE -III

Paper Code: CSCS235

L	T	P
3	0	0

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

Basic Probability - Random Experiments - Sample Spaces Events - The Concept of Probability -The Axioms of Probability - Some Important Theorems on Probability - Assignment of Probabilities -Conditional Probability -Theorems on Conditional Probability -- Independent Events -Bayes’ Theorem or Rule Combinatorial Analysis - Fundamental Principle of Counting -Tree Diagrams -Permutations

MODULE – II

Random Variables and Probability Distributions - Random Variables - Discrete Probability Distributions -Distribution Functions for Random Variables - Distribution Functions for Discrete Random Variables - Continuous Random Variables – Graphical Interpretations Joint Distributions Independent Random Variables - Change of Variables - Probability Distributions of Functions of Random Variables – Convolutions – Conditional Distributions Applications to Geometric Probability

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:

1. Murray R. Spiegel, John J. Schiller & R. Alu Srinivasan, “Probability and Statistics”, Schaum outlines,
McGraw Hill, 3rd edition, 2009.
2. S. P. Gupta, Statistical Methods, S. Chand and Sons.
3. S. C Gupta and V. K. Kapoor, “Fundamentals of Mathematical Statistics”, 11th edition, S. Chand and Sons.

GENERIC ELECTIVE - IV

Paper Code: CSCS236

L	T	P
3	0	0

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

Automata Theory: Finite State Systems – Definition of an Automaton - Non-Deterministic Finite Automaton - Equivalence of DFA and NFA - Finite automata with output (Mealy and Moore Models) - Minimization of Finite Automata – Regular Expressions.

MODULE - II

Push Down Automata Theory: Context-Free Languages and Derivation Trees – Ambiguity in Context-Free Grammars – Chomsky Normal Form – Greibach Normal Form. Push Down Automata – Definition, Acceptance by Push Down Automata – Push Down Automata and Context Free Languages.

MODULE - III

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

MODULE - IV

Chomsky Hierarchy: Regular Grammars – Unrestricted Grammars – Context Sensitive Languages. Linear Bounded Automata – Definition – Linear Bounded Automata and Context Sensitive Languages – Undecidability - Properties of recursive and recursively enumerable languages – Turing Machine Codes – Universal Turing Machine.

MODULE - V

Computational Complexity Theory: Space Complexity – Time complexity – Non-Deterministic Time and Space Complexity – Complexity Classes - Computability: Basic Concepts – Primitive Recursive Functions – Recursive Functions - Case Studies: Application of Finite Automata – Parsing.

Text Books:

1. Daniel I.A. Cohen, “Introduction to Computation Theory”, John Wiley & sons.
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

L	T	P
3	1	0

Objective:

- To acquire the basic knowledge about computers

Module - I

Introduction to Computers - Generation of Computers - Classification of Digital Computer - Anatomy of Digital Computer.

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

Computers at Home, Education, Entertainment, Science, Medicine and Engineering - Introduction to Computer Security - Computer Viruses, Bombs, Worms.

Text Books:

1. Fundamentals of Information Technology, Alexis Leon and Mathews Leon, Vikas Publishing House Pvt. Ltd., 2009.
2. Faithe Wempen, Computing Fundamentals: Introduction to Computers, WILEY, 2014.

Paper Code: CSCS172

L	T	P
3	1	0

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:

1. E. Balagurusamy, “Programming with ANSI-C”, Fourth Edition, Tata McGraw Hill, 2008,
2. Hanly J R & Koffman E.B, “Problem Solving and Programming design in C”, Pearson Education, 2009.

Paper Code: CSCS173

L	T	P
3	1	0

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

MODU|LE – 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:
Faithe Wempen, “HTML5 Step by Step”, Microsoft Press, 2011.

Paper Code: CSCS174

L	T	P
3	1	0

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

Word processing - Introduction - Introduction to MS Word - Working with MS Word - Creating a New Document-Different Page Views and layouts - Working with Styles, Text Attributes; Paragraph and Page Formatting - Text Editing using various features - Advanced Features of MS-Word – bookmarks - Spell Check and Thesaurus; Find & Replace; Headers & Footers ; Inserting – Page Numbers, Pictures, Files, Auto texts, Symbols -Working with Columns, Tabs & Indents;- Creation & Working with Tables -Margins & Space management in Document - Mail Merge.

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:

1. Rob Tidrow, Master Visually Windows 7, John Wiley, 2010.
2. Dinesh Maidasani , Straight to the Point – MS Office 2010, Laxmi Publications, 2010.
3. Sherry Kinkoph Gunter, Master Visually Microsoft Office 2010, WILEY, 2010.
4. Faithe Wempen, Computing Fundamentals: Introduction to Computers, WILEY, 2014.